

Messung und Auswertung der räumlichen Strahleigenschaften eines Femtosekundenlasers

von Sommer Daniel, Mat.# 01325734

Diplomarbeit

zur Erlangung des akademischen Grades Diplomingenieur

eingereicht an der Technischen Universität Wien

Betreuer Liedl Gerhard, Ao.Univ.Prof. Dipl.-Ing. Dr.techn.

Mitbetreuer Fürbacher Roland, Projektass. Dipl.-Ing. BSc

Institut für Fertigungstechnik und Photonische Technologien

Fakultät für Maschinenwesen und Betriebswissenschaften

Wien, April 2022

Abstract

Zusammenfassung

Laser sind komplexe Geräte, die in vielen Bereichen der Technik nicht mehr wegzudenken sind. Egal in welchem Bereich sie verwendet werden ist es notwendig, dass die Eigenschaften des Lasers kontrolliert und für die jeweilige Anwendung optimiert werden. Damit das möglich ist, müssen die Strahleigenschaften des Lasers bekannt sein. Es gibt verschiedene Methoden wie diese bestimmt werden können. Eine davon basiert auf der Methode der zweiten Momente, welche in der DIN EN ISO 11146 beschrieben ist. Diese Methode verwendet die mit einer pixelbasierten Kamera aufgenommenen Strahlprofile um und in der Strahltaille eines Laserstrahls und berechnet daraus die Strahleigenschaften des Lasers. Im Zuge dieser Arbeit und auf Basis der Norm wurde ein Messaufbau und eine automatisierte Mess- und Auswertungssoftware entwickelt, mit der die Eigenschaften von stigmatischen und leicht astigmatischen Strahlen bestimmt werden können. Um die Ergebnisse der automatisierten Messung zu verifizieren wurden die aufgenommenen Strahlprofile mit Brandlöchern verglichen. Dabei konnte festgestellt werden, dass das Verhältnis der Strahlabmessungen des Messsystems und der Brandlöcher proportional zum Verhältnis der Brennweiten der verwendeten Linsen ist und bestätigt, dass sowohl der Messaufbau, als auch die automatisierten Messund Auswertungssoftware funktioniert. Für den Messaufbau ist zu beachten, dass die verwendete Kamera, die optischen Elemente und die mechanische Komponenten speziell für einen Ti:Saphir Femtosekundenlaser ausgewählt wurden. Sollte ein anderer Laser das Ziel der Messung sein, muss der Messaufbau mit passenden Komponenten adaptiert werden.

Abstract Englisch

Lasers are complex devices that have become indispensable in many areas of technology. No matter in which field they are used, it is necessary that the properties of the laser are controlled and optimized for the respective application. For this to be possible, the beam properties of the laser must be known. There are several methods how these can be determined. One of them is based on the second moment method, which is described in DIN EN ISO 11146. This method uses the beam profiles around and in the beam waist of a laser beam taken with a pixel-based camera and calculates the beam properties of the laser from them. In the course of this work and based on the standard, a measurement setup and automated measurement and evaluation software were developed to determine the properties of stigmatic and slightly astigmatic beams. To verify the results of the automated measurement, the recorded beam profiles were compared with burn holes. It was found that the ratio of the beam dimensions of the measurement system and the burn holes is proportional to the ratio of the focal lengths of the lenses used, confirming that both the measurement setup and the automated measurement and evaluation software work. For the measurement setup, it should be noted that the camera, optical elements and mechanical components used were specifically selected for a Ti:Sapphire femtosecond laser. If another laser is the target of the measurement, the measurement setup must be adapted with suitable components.

Inhaltsverzeichnis

1	Einl	eitung	5				
	1.1	.1 Motivation					
	1.2	Licht	5				
		1.2.1 Lichtwellen	5				
		1.2.2 Polarisation und Farbe des Lichts	6				
		1.2.3 Absorption von Licht	7				
		1.2.4 Spontane Emission	8				
		1.2.5 Induzierte Emission	9				
	1.3	Aufbau von Lasern	0				
	1.4	Laserparameter	1				
		1.4.1 Wellenlänge λ , Frequenz f	1				
		1.4.2 Leistung P , Pulsenergie E , Pulsdauer τ	1				
		1.4.3 Polarisation $\ldots \ldots \ldots$	2				
		1.4.4 Divergenz θ	2				
		1.4.5 Beugungsmaßzahl M^2	2				
		1.4.6 Longitudinal und Transversalelektromagnetische (TEM) Modenstruktur 1	2				
	1.5	Laserstrahlcharakterisierung	4				
	1.6	Ti:Saphir Femtosekundenlaser	4				
2	Best	timmung der Strahleigenschaften 1	7				
	2.1	Hintergrundkorrekturverfahren	$\overline{7}$				
	2.2	Grobkorrektur (Coarse correction)	$\overline{7}$				
		2.2.1 Coarse correction by background map subtraction (CCBMS) 1	8				
		2.2.2 Coarse correction by average background subtraction (CCBAM) 1	8				
	2.3	Feinkorrektur (Fine correction)	8				
		2.3.1 Fine correction by statistical method (FCBSM)	8				
		2.3.2 Fine correction by approximation method (FCBAM)	1				
	2.4	Anwendung der Hintergrundkorrekturverfahren	1				
	2.5	Bestimmung des Strahldurchmessers aus einer gegebenen Leistungsdichteverteilung . 2	3				
		2.5.1 Stigmatische und leicht astigmatische Strahlen	3				
	2.6	Anwendung auf Einzelaufnahme eines Strahlprofils mit pixelbasierten Sensor 2	5				
	2.7	Bestimmung von Taillenlagen, Divergenzwinkeln und Beugungsmaßzahlen 2	6				
3	Mes	saufbau 2	8				
	3.1	Messaufbau 1	8				
		3.1.1 Vorversuch mit einem He-Ne Laser	9				
		3.1.2 Ergebnisse des Vorversuchs mit Messaufbau 1	0				
	3.2	Messaufbau 2	0				
		3.2.1 Ergebnisse des Vorversuchs mit Messaufbau 2	1				
	3.3	Messaufbau 3	2				
		3.3.1 Ergebnisse aus dem Vorversuch mit Messaufbau 3	3				
	3.4	Verwendete Hardware und Software	3				
		3.4.1 Unbeschichtetes rechtwinkliges Prisma (Thorlabs PS908) [10]	3				
		3.4.2 Neutraldichtefilter ND-Filter (Thorlabs NE40B-B, NE50B-B) [13] 3	3				

Inhaltsverzeichnis

		3.4.3	Schrittmotor (Sanyo Denki 103H5205-5240) [16]	34						
		3.4.4	Steuerboard (Arduino UNO Rev3) [19]	35						
		3.4.5	Treiberboard (Arduino Motor Shield Rev3) [21]	35						
		3.4.6	Kamera (Basler ace GigE acA1300-60gmNIR) [23]	36						
		3.4.7	MATLAB R2020b	36						
4	Messprogramm 37									
	4.1	Klasse	enaufbau	37						
	4.2	Progra	ammabläufe und GUI	40						
		4.2.1	Definition des Messgitters	41						
		4.2.2	Messzyklus	42						
		4.2.3	Auswertung	44						
	4.3	Livem	essung	48						
5	Messergebnisse 49									
•	5.1	Messei	rgebnisse vor dem Pumplasertausch	50						
	5.2	Messe	rgebnisse nach dem Pumplasertausch	53						
	5.3	Vergle	ich theoretische und reale Fokusdurchmesser	57						
6	Zusa	ammen	fassung und Ausblick	59						
7	Anh	ang		64						
	Anh	ang 1: 0	Quellcode	64						
	Anh	ang 2:]	Installationsanleitung	191						
	Anh	ang 3:]	Basler GigE Vision Quick Start Configuration Guide	192						
	Anhang 4: Basler Application Note									
	Anh	ang 5: 1	Bedienungsanleitung	228						
Lit	erati	Jr		229						

1.1 Motivation

Verschleiß ist in jedem Bereich der Technik ein Thema das nicht ignoriert werden kann. Dadurch ist es notwendig jede Maschine oder Einrichtung regelmäßig zu warten, damit die langfristige Funktionalität sichergestellt ist. Auch bei Lasern ist das eine Notwendigkeit. Sowohl das Lasermedium als auch die notwendigen Peripherie im Laseraufbau verschleißen mit der Zeit und müssen ersetzt oder nachjustiert werden. Die Komplexität des Lasers ist proportional zum Justierungsaufwand, da die optischen Elemente alle voneinander abhängen, wodurch bereits kleine Änderungen ungewollte Schwierigkeiten bereiten können. Um die Justierung zu vereinfachen bzw. die derzeitigen Eigenschaften des Lasers zu bestimmen existiert das Verfahren der Laserstrahlcharakterisierung. Es ist ein Verfahren bei dem die Strahleigenschaften in Ausbreitungsrichtung bestimmt werden. Kennt man diese Eigenschaften, ist es möglich eine Aussage über den derzeitigen Zustand des Lasers zu treffen und auch festzustellen ob die Justierung die gewünschten Verbesserungen bzw. Änderungen gebracht hat.

1.2 Licht

Die Eigenschaften von Licht werden meistens mit vereinfachten Modellen beschrieben. Eine der ersteren Vorstellungen war es, dass Lichtquellen wie z.B. die Sonne Lichtteilchen auf geradlinigen Bahnen aussenden. Bekannt ist diese Theorie als Teilchenmodell des Lichts [1]. In einem einfachen Experiment mit einer Lochblende und einer Lichtquelle wurde versucht einen scharf begrenzten Strahl herzustellen um das Teilchenmodell zu bestätigen. Der Versuch ergab, dass es nach der Lochblende bedingt durch Beugungseffekte des Lichts zu einer Aufweitung des Strahls gegenüber dem Lochdurchmessers kam. Dieses Verhalten kann durch das Wellenmodell des Lichts erklärt werden. Beide Modelle können mit komplexer Mathematik zu einem einheitlichen Modell zusammengefasst werden. In der Realität reicht für die meisten Beobachtungen jeweils eines der Modelle.

1.2.1 Lichtwellen

Im Wellenmodell [2] wird das Licht als transversale elektromagnetische Welle beschrieben, in der die elektrische Feldstärke **E** und die magnetische Feldstärke **H** mit gleicher Frequenz f schwingen. Die beiden Vektoren stehen stets senkrecht aufeinander siehe Abb. 1.1. Die Wellenlänge λ des Lichts ist dabei der Abstand zweier benachbarter Phasenflächen. Es gilt mit der Lichtgeschwindigkeit im Vakuum $c = 2.998 \times 10^8 \,\mathrm{m\,s^{-1}}$ der Zusammenhang

$$c = \lambda \cdot f. \tag{1.1}$$

In einem transparenten Medium breitet sich das Licht mit der reduzierten Lichtgeschwindigkeit c'aus. Diese reduzierte Lichtgeschwindigkeit hängt vom Brechungsindex n ab und kann mit

$$c' = \frac{c}{n} \tag{1.2}$$

bestimmt werden. Der Brechungsinde
xnkann mit der Dielektrizitätszahl ϵ und der Perme
abilität μ über

$$n = \frac{1}{\sqrt{\epsilon\mu}} \tag{1.3}$$



Abbildung 1.1: Verlauf der elektrischen Feldstärke \mathbf{E} und der magnetischen Feldstärke \mathbf{H} in Ausbreitungsrichtung einer Lichtwelle [2]

bestimmt werden. Trifft das Licht auf eine Grenzfläche siehe Abb. 1.2 zwischen zwei optischen Medien mit Brechungszahlen n_1 und n_2 gilt das Brechungsgesetz nach Snell

$$n_1 \sin \alpha = n_2 \sin \beta. \tag{1.4}$$

Der Winkel α ist der Eintrittswinkel des Primärstrahls bezogen auf die Grenzflächennormale, α'



Abbildung 1.2: Brechungsgesetz

der Winkel der reflektierten Strahls wobei gilt

$$\alpha' = -\alpha \tag{1.5}$$

und β der Winkel des transmittierten Strahls. Sowohl der Winkel α , als auch der Winkel β sind bezogen auf die Grenzflächennormale. Beide Winkel werden ausgehend von dieser Normale gemessen. Schließt z.B. α ausgehend von der Normale einen Winkel gegen den Uhrzeigersinn ein, so schließt auch β einen Winkel gegen den Uhrzeigersinn ein.

1.2.2 Polarisation und Farbe des Lichts

Grundsätzlich beschreibt die Polarisation des Lichts die Schwingungsrichtung einer sich transversal ausbreitenden Welle. Man bezieht sich dafür immer auf den Vektor der elektrischen Feldstärke. Ist die Richtung der elektrischen Feldstärke **E** konstant, so spricht man von linear polarisierten Licht. Die meisten Lichtquellen bestehen aus einem Gemisch von Wellen unterschiedlicher Polarisation und werden daher als unpolarisiert bezeichnet. Die Farbe des Lichts ist abhängig von der Frequenz bzw.

der Wellenlänge. Das menschliche Auge reagiert unterschiedlich empfindlich auf die verschiedenen Wellenlängen des Lichts siehe Abb. 1.3. Ist das Auge auf eine helle Umgebung eingestellt V'(I) reagiert es auf Licht mit einer Wellenlänge von 550 nm am empfindlichsten. Ist es auf dunkle Umgebung eingestellt verschiebt sich die Empfindlichkeit weiter nach links zu 500 nm. Der für das



Abbildung 1.3: relative Lichtempfindlichkeit des menschlichen Auges [2]

menschliche Auge sichtbare Bereich liegt zwischen 380 nm und 780 nm.

1.2.3 Absorption von Licht

Beschrieben wird Absorption mithilfe einer ebenen Lichtwelle der Intensität I_0 die auf eine absorbierende Stoffschicht mit Dicke *d* trifft [2]. Am hinteren Ende der der Stoffschicht beträgt die aus der Schicht austretende Intensität *I*. Die verminderte Intensität *I* ist proportional zu I_0 , hängt von der Schichtdicke ab und kann mit

$$I = I_0 e^{-\alpha d} \tag{1.6}$$

auch bekannt als Lambert-Beer'sches Gesetz beschrieben werden. Führt man eine Koordinate x ein und nimmt an, dass die Intensität an einer Stelle I(x) um dI abnimmt ergibt sich für die Ableitung des Lambert-Beer'schen Gesetzes

$$dI = -\alpha I x dx. \tag{1.7}$$

Der Faktor α ist ein materialabhängiger Absorptionskoeffizient und beträgt z.B. für Glasfaser $\alpha \approx 1$ bis 10 km oder für Metalle $\alpha \approx 1$ nm.

Der Vorgang der Absorption kann mithilfe diskreten Energiezuständen $(E_1, E_2, E_3, ..., E_n)$ dargestellt werden (siehe Abb .1.4). Im nicht angeregten oder auch ungestörten Zustand nehmen alle Atome den Zustand der niedrigsten Energie E_1 ein. Dieser Zustand wird auch als Grundzustand bezeichnet. Trifft Licht mit einer Frequenz f_{12} auf ein Atom, ist es möglich das das Atom in einen höheren Energiezustand E_2 übergeht, solange die Bohrsche Frequenzbedingung

$$E_2 - E_1 = h f_{12} \tag{1.8}$$

erfüllt wird, wobei $h = 6.626^{-34}$ Js das Plancksche Wirkungsquantum beschreibt. Im Gegenzug bedeutet das, dass dem Licht die Energie hf_{12} entzogen wurde und somit auch die Intensität I



Abbildung 1.4: Schematische Darstellung von Absorption, spontaner Emission und induzierter Emission [2]

vermindert. Aus der Anzahl der Übergänge von Zustand E_1 nach E_2 lässt sich der Absorptionskoeffizient berechnen. Sie kann pro Volumen- und Zeiteinheit definiert werden mit

$$\left|\frac{dN_1}{dt}\right|_a = \sigma_{12}N_1\Psi,\tag{1.9}$$

wobei N_1 die Dichte der Atome im Grundzustand, Ψ die Stromdichte (Photonen/Zeit-Fläche) und σ_{12} die Wirkfläche der Absorption beschreibt. Der Index *a* symbolisiert das die Übergänge durch Absorption entstehen. Sie ist gleich der Änderung der Photonendichte (Photonen/Volumen) $d\Phi/dt$. Mit der Stromdichte Ψ und der Änderung der Schichtdicke über der Zeit kann die Photonendichte ausgedrückt werden mit

$$\Phi = \Psi \frac{dt}{dx}.$$
(1.10)

Daraus ergibt sich der Zusammenhang

$$\left|\frac{dN_1}{dt}\right|_a = -\frac{d\Psi}{dx}.$$
(1.11)

Da die Photonendichte abnimmt ist es mit einem negativen Vorzeichen versehen. Mit der Definition der Intensität

$$I = h f_{12} \cdot \Psi, \tag{1.12}$$

und Gl. 1.11 ergibt sich für die Abnahme der Intensität durch Absorption

$$\left. \frac{dI}{dx} \right|_a = -\sigma_{12} N_1 I. \tag{1.13}$$

Durch Vergleich mit Gl. 1.7 kann der Absorptionskoeffizient gefunden werden mit

$$\alpha = \sigma_{12} N_1. \tag{1.14}$$

Aus dieser Beziehung lässt sich erkennen, dass der Absorptionskoeffizient proportional zur Dichte der absorbierten Atome ist.

1.2.4 Spontane Emission

Angeregte Atome gehen nach einer gewissen Zeit wieder in ihren Zustand der niederen Energie über [2]. Dabei kann Energie in Form eines Lichtquants in eine beliebige Richtung abgestrahlt werden. Dieser Prozess ist allgemein bekannt als spontane Emission. Durch spontane Emission

nimmt die Atom
dichte der Atome im höheren Energiezustand ab. Mithilfe der Lebensdauer für spontane Emission
 τ und der Dichte der im höheren Energiezustand vorhandenen Atom
e N_2 kann diese Änderung beschrieben werden mit

$$\left|\frac{dN_2}{dt}\right|_{sp} = \frac{N_2}{\tau}.$$
(1.15)

1.2.5 Induzierte Emission

Der dritte wichtige Mechanismus für Laser ist die induzierte Emission [2]. Bei dieser Art der Emission erfolgt der Übergang der Atome vom höheren zum niedrigeren Energiezustand nicht spontan, sondern mithilfe einer Lichtwelle, welche die Bohrsche Frequenzbedingung erfüllt. Die Beziehungen dieses Prozesses sind analog zu denen der Absorption und ergeben sich zu

$$\left|\frac{dN_2}{dt}\right|_i = \sigma_{21}N_2\Psi \tag{1.16}$$

und

$$\left|\frac{dI}{dx}\right|_{i} = \sigma_{21} N_2 I. \tag{1.17}$$

In diesen Beziehungen ist N_2 wie bei der spontanen Emission die Dichte der Atome im höheren Energiezustand und σ_{21} der Wirkungsquerschnitt der induzierten Emission. Im Gegensatz zur spontanen Emission, bei der ein Lichtquant in eine beliebige Richtung abgestrahlt wird, wird bei der induzierten Emission das Lichtquant in Ausbreitungsrichtung der Lichtwelle abgestrahlt. Die Phase und Frequenz dieser induzierten Welle ist identisch zur einfallenden Welle und verstärkt damit die einfallende Welle. Aus thermodynamischen und quantenmechanischen Überlegungen kann gezeigt werden, dass der Wirkungsquerschnitt der Absorption σ_{12} gleich groß ist wie der der induzierten Emission σ_{21} , sofern die Energieniveaus gleiche statistische Wahrscheinlichkeit haben.

$$\sigma_{12} = \sigma_{21} = \sigma \tag{1.18}$$

Gibt es in den Energieniveaus Unterzustände gilt die Beziehung

$$g_1 \sigma_{12} = g_2 \sigma_{21} = \sigma, \tag{1.19}$$

wobei g_1 und g_2 die Anzahl der Unterzustände des jeweiligen Energieniveaus beschreiben. Die Verstärkung der eingestrahlten Lichtwelle durch die induzierte Emission wirkt der Absorption entgegen. Dadurch ergibt sich die Gesamtänderung der Intensität der Lichtwelle zu

$$dI = -\left|dI\right|_{a} + \left|dI\right|_{i} \tag{1.20}$$

Für Energieniveaus die keine Unterzustände besitzen $(g_1 = g_2 = g = 1)$ gilt daher

$$\frac{dN_2}{dt} = -\sigma N_1 I + \sigma N_2 I = -\sigma (N_1 - N_2) I$$
(1.21)

Integriert man diese Beziehung erhält man das verallgemeinerte Lambert-Beer'sche Gesetz

$$\frac{I}{I_0} = e^{\sigma(N_1 - N_2)d} = G.$$
(1.22)

Das Verhältnis von I/I_0 wird auch Verstärkungsfaktor G genannt. Mit einer He-Ne Gaslaserentladung einer Länge von 1 m lässt sich zum Beispiel im kontinuierlichen Betrieb ein Verstärkungsfaktor von G = 1.1 erreichen.

Damit eine Verstärkung auftreten kann, muss Besetzungsinversion $(N_2 > N_1)$ der Energieniveaus gegeben sein. Bewerkstelligt wird das durch das sogenannte Pumpen. Pumpen beschreibt den Prozess die Atome des Lasermediums z.B He-Ne Gas auf ein höheres Energieniveau zu heben. Der Pumpmechanismus der verwendet wird hängt im allgemeinen vom Lasertyp ab. Die wichtigsten Laserobertypen sind Gaslaser, Festkörperlaser, Farbstofflaser und Halbleiterlaser.

1.3 Aufbau von Lasern

Ein Laser ist im Grunde ein Oszillator (Verstärker) für Signale mit optischen Frequenzen [3]. Der Frequenzbereich streckt sich dabei vom Infrarotbereich bis zum ultravioletten bzw. dem Bereich der Röntgenstrahlung. Der Grundaufbau eines Lasers beinhaltet dabei immer einen Verstärker (Oszillator). Dieser ist von der Funktionsweise her vergleichbar mit einem elektrischen Verstärker. Eine typische Verstärkerschaltung ist in Abb. 1.5 dargestellt. Die Eingangsspannung V_i wird mit der Rückkopplung βV_o summiert und anschließend über das Verstärkerbauteil mit Verstärkung A_0 verstärkt.



Abbildung 1.5: Schaltdiagramm eines Verstärkers mit positiver Rückkopplung [3]

Die Verstärkung A_0 ohne Rückkopplung ist dabei gegeben durch

$$A_0 = \frac{V_o}{V_i} \tag{1.23}$$

dem Verhältnis der Ausgangsspannung V_o zur Eingangsspannung V_i . Um die Gesamtverstärkung der Schaltung zu bestimmen muss die Rückkopplung berücksichtigt werden. Der Rückkopplungsfaktor β ist dabei im allgemeinen eine komplexe Zahl und kann als komplexer Vektor der Form

$$\beta = |\beta| e^{j\varphi} \tag{1.24}$$

dargestellt werden. Er besitzt einer Amplitude $|\beta| \leq 1$ und eine Phase φ . Die Ausgangsspannung V_o ergibt sich mit der Rückkopplung zu

$$V_o = A_o(V_i + \beta V_o) \tag{1.25}$$

und die Gesamtverstärkung A zu

$$A = \frac{A_o V_i}{1 + \beta A_0}.\tag{1.26}$$

Theoretisch ist damit eine unendlich große Verstärkung des Eingangssignals möglich wenn βA_0 genau +1 erreicht. Das würde bedeuten, dass ohne ein Eingangssignal ein finites Ausgangssignal

generiert wird. In der Praxis gibt es in jeder elektrischen Schaltung einen gewissen Rauschanteil, wodurch das Eingangssignal nie wirklich 0 sein kann. Der Faktor βA_0 ist frequenzabhängig und im allgemeinen nur bei einer bestimmten Frequenz genau +1. Bei dieser Frequenz schwingt die Schaltung indem das Rauschsignal verstärkt wird. Theoretisch sollte der Ausgang unendlich groß werden. Das passiert in der Realität allerdings nicht, denn umso größer das Signal wird, umso kleiner wird die Verstärkung A_0 . Dieser Effekt ist bekannt als Sättigung und stellt ein fundamentales Phänomen für alle Verstärker dar. Der schematische Aufbau eines optischen Verstärkers ist in Abb. 1.6 abgebildet. Er besteht aus zwei gegenüberliegend angeordneten Spiegeln mit Abstand L und einem optischen Frequenzverstärker in der Mitte zwischen beiden Spiegeln.



Abbildung 1.6: Schematische Darstellung eines optischen Verstärkers [3]

1.4 Laserparameter

Die Laserparameter [2] können in Laserkenndaten und Laserstrahlparameter unterteilt werden. Zu den Laserkenndaten gehören die Wellenlänge, die Frequenz, die Leistung, die Energie und der Betriebsart. Zu den Laserstrahlparametern gehören das Strahlprofil, die transversale Modenstruktur, die Strahldivergenz, die Fokussierbarkeit, die Kohärenz und die Polarisation.

1.4.1 Wellenlänge λ , Frequenz f

Die Wellenlänge λ beschreibt den kleinsten Abstand zweier Punkte gleicher Phase einer periodischen Welle. Sie kann mithilfe der Lichtgeschwindigkeit c und der Frequenz f der Welle mit

$$\lambda = \frac{c}{f} \tag{1.27}$$

bestimmt werden.

1.4.2 Leistung P, Pulsenergie E, Pulsdauer τ

und Abstand zweier Pulse T Die Leistung P [2] eines Lasers beschreibt die Ausgangsleistung eines Dauerstrichlasers in Watt. Für gepulste Laser wird die Ausgangsleistung charakterisiert über die Pulsenergie E in Joule, die Pulsdauer τ in Sekunden und der zeitliche Abstand zweier Pulse T in Sekunden. Die Pulsspitzenleistung P_{max} ist gegeben durch die Pulsenergie über der Pulsdauer τ mit

$$P_{max} = \frac{E}{\tau} \tag{1.28}$$

und die mittlere Leistung P über

$$P = \frac{E}{T} \tag{1.29}$$

1.4.3 Polarisation

Die Polarisation [2] definiert die Orientierung des elektrischen Feldes einer Lichtwelle. Laserlicht ist meistens linear polarisiert. Linear polarisiert bedeutet, dass der Vektor der elektrischen Feldstärke immer in einer definierten Ebene schwingt.

1.4.4 Divergenz θ

Die Divergenz [4] beschreibt die Verbreiterung eines Strahls mit zunehmender Entfernung im Bezug auf die Strahltaille des Lasers durch Beugung. Sie wird definiert über den Halbwinkel des Lasers und ist bei Gaußstrahlen definiert als

$$\theta = \frac{\lambda}{\pi w_0},\tag{1.30}$$

wobei w_0 der Strahltaillenradius und λ die Wellenlänge des Laserstrahl beschreibt.

1.4.5 Beugungsmaßzahl M^2

Das Strahlprofil eines Laserstrahls beschreibt die Intensitätsverteilung des Laserstrahlquerschnitts. Die gebräuchlichsten Strahlprofile haben entweder eine gaußsche oder eine Flat-Top Verteilung. Ein gaußsches Profil besitzt die geringste Divergenz und ist daher für viele Anwendungen ideal. Die Beugungsmaßzahl M^2 [4] beschreibt die Abweichung des Divergenzwinkels eines realen Strahls von einem idealen Gaußstrahl und ist somit definiert als

$$M^2 = \frac{\varphi_{real}}{\varphi_{gauss}} \tag{1.31}$$

1.4.6 Longitudinal und Transversalelektromagnetische (TEM) Modenstruktur

Jede Welle [5] besteht aus unterschiedlichen Schwingungsformen (Moden). Die Moden in Ausbreitungsrichtung der Strahlung werden Longitudinal Moden und senkrecht zur Ausbreitungsrichtung als Transversal Moden bezeichnet. In longitudinaler Richtung können grundsätzlich alle Wellenlängen, die in den Resonator passen und einen Schwingungsknoten auf den Resonatorspiegeln aufweisen, anschwingen. Besteht es nur aus rein einer optischen Frequenz spricht man von monochromatischen Licht ansonsten von polychromatischem Licht. Welche Moden in transversaler Richtung schwingen wird vom Aufbau des Lasers beziehungsweise vom Resonator bestimmt. Der TEM Grundmode entspricht einem idealen Gaußprofil und wird auch als TEM_{00} bezeichnet. Bei den TEM höherer Ordnung kann zwischen radialsymmetrischen Moden (siehe Abb. 1.7a) und rechteckig symmetrischen Moden (siehe Abb. 1.7b) unterschieden werden. Welcher Mode sich ausbildet kann unter anderen durch Blenden beeinflusst werden. Ist im Resonator keine Blende verbaut, können viele Moden gleichzeitig schwingen. Wird jedoch eine Blende verbaut, ist es möglich die Anzahl der schwingenden Moden zu reduzieren, indem der Durchmesser der Blende verkleinert wird. Möchte man das Strahlprofil nur auf den TEM_{00} Grundmode beschränken, muss der Durchmesser der Blende nahe dem Durchmesser des TEM_{00} Profils gewählt werden. Dadurch werden für alle anderen Moden hohe Verluste an der Blende generiert, wodurch diese nicht schwingen können. Erhöht man den Durchmesser der Blende können auch Moden höherer Ordnung die "durch die Blende passen" schwingen. Niedrige Moden können auch über die Spiegelform und der Resonatorgeometrie, d.h. dem Verhältnis Spiegeldurchmesser und Resonatorlänge selektiert werden.

Dasselbe Prinzip gilt für die Erzeugung anderer Moden. Möchte man einen speziellen Mode generieren müssen für die anderen Moden Verluste verursacht werden. Eine Möglichkeit das zu Bewerkstelligen ist es zwei gekreuzte Drähte im Resonator zu platzieren (siehe Abb. 1.8). Die Drähte verursachen entlang ihrer Länge Verluste. Dadurch werden Moden bevorzugt die Symmetrien um



Abbildung 1.7: TEM Moden mit unterschiedlicher Symmetrie

(a) Berechnete TEM Moden mit radialer Symmetrie[5]



(b) Berechnete TEM Moden mit rechteckiger Symmetrie [5]



Abbildung 1.8: Aufbau zur Beobachtung von individuellen TEM Moden [5]

die Achsen der Drähte besitzen, da diese weniger beeinflusst werden. Wendet man das Prinzip auf einen He-Ne Laser an, können rechteckig symmetrische Moden unterschiedlicher Ordnung erzeugt werden. Abb. 1.9 zeigt TEM Moden die mithilfe so eines Aufbaus erzeugt werden können. Die Drähte wurden an die Stellen verschoben an der der jeweilige Mode seine Symmetrie besitzt und wird dadurch bevorzugt. Der Winkel der Drähte beeinflusst dabei nicht welcher Mode generell schwingt, sondern nur um welche Achse sich der Mode symmetrisch ausbildet.



Abbildung 1.9: Aufgenommene Strahlprofile für verschiedene TEM Moden eines He-Ne Lasers [5]

1.5 Laserstrahlcharakterisierung

Laserstrahlcharakterisierung ist eine Methode um die Strahleigenschaften von Lasern zu ermitteln. Die wichtigsten Parameter sind dabei die Beugungsmaßzahl M^2 , der Ort der Strahltaille d_{z_0} , der Strahldurchmesser d_{σ} und der Divergenzwinkel Θ . Kennt man diese Parameter, ist es möglich den Laserstrahl für jede Anwendungen optimal zu nutzen, bzw. während dem Justieren des Lasers die Einflüsse des Justierens selbst auf den resultierenden Laserstrahl zu beobachten. Besonders wichtig ist die Kenntnis dieser Parameter bei Lasern mit komplexem, internen Aufbau, da bei solchen Lasern regelmäßige Justagearbeiten notwendig sind, um die maximale Arbeitsleistung zu erreichen.

1.6 Ti:Saphir Femtosekundenlaser

Es ist nicht Inhalt dieser Arbeit auf alle Einzelheiten, der physikalischen Effekte für die Erzeugung von Femtosekundenpulsen einzugehen [6]. Ein Überblick über den Aufbau und die Funktionsweise ist aber insofern wichtig, um die Notwendigkeit einer regelmäßigen Überprüfung des Strahlprofils zu erklären. Das FEMTOPOWER COMPACT PRO Lasersystem mit dem Pulse einer Pulsdauer von < 30fs erzeugt werden können besteht aus zwei separaten Lasern (siehe Abb. 1.10): Einem Ti:Saphir Oszillator-laser gepumpt von einem CW Diodenlaser, einer Verstärkerstufe ebenfalls mit einem Ti:Saphir Kristall und einem Nd:YLF Pumplaser. Der Oszillatorlaser liefert Pulse mit einer Wiederholfrequenz von 74-78MHz, einer Pulsdauer von < 12fs und einer Pulsenergie von > 2 nJ (siehe Tab. 1.1). Diese spektral breitbandigen Pulse werden über den Effekt der sogenannten Group-

Output	amplifier	oscillator				
Pulse duration (FHWM)	$< 30 \ fs$	$< 12 \; fs$				
Spectral width (FHWM) @800nm	$> 40 \ nm$	$> 100 \ nm$				
Outtput energy	$> 800 \ \mu J$	> 2 nJ				
Pulse repetition rate	1 kHz	74 - 78 MHz				
Peak power	$> 30 \ GW$	$> 500 \ kW$				
Beam diameter $(1/e^2)$	15 mm (nominal)	>2 mm				
B-integral of amplified pulses	< 1	-				
Spatial mode	$TEM_{00}(M^2 < 2)$	$TEM_{00}(M^2 < 1.3)$				
Contrast ratio	$> 10^7 : 1/ > 10^4 : 1$	-				
Polarizations	linear, horizontal	linear, horizontal				
Pulse-to-pulse energy stability	$< 1.5 \% \ rms$	-				
Beam divergence	< 3 mrad	< 2 mrad				

 Tabelle 1.1: Datenblatt FEMTOPOWER COMPACT PRO



Abbildung 1.10: Vereinfachte Schemaskizze eines Ti:Saphir Femtosekunden Lasers

Delay-Dispersion (GDD) mit einem positiven chirp zeitlich um einen Faktor von ca. 10⁵ aufgeweitet um eine weitere Verstärkung zu ermöglichen. Würden die Oszillatorpulse nicht aufgeweitet werden, würde aufgrund der Selbstfokussierung im Ti:Saphir Kristall, die Intensität der Strahlung den Schadengrenzwert des Kristalls überschreiten und ihn beschädigen. Um eine höhere Verstärkung zu erreichen wird der Oszillator Laserstrahl mehrmals durch den Verstärkerkristall geführt. Durch das mehrmalige "durchfädeln" des Oszillatorlaserstrahls durch den Verstärker Ti:Saphir Kristall wird die Energie der Laserpulse immer weiter erhöht. Nach der Verstärkung wird der aufgeweitete Puls über einen Prismenkompressor bestehend aus 4 Prismen komprimiert um den zuvor eingeführten

positiven chirp wieder auszugleichen. Die einzelnen Prismen des Prismenkompressors sind verschiebbzw. verdrehbar und das Einstellen dieser Prismen ist kein simples Unterfangen. Ändert sich die mittlere Wellenlänge des Laserpulses [7], müssen alle 4 Prismen um denselben Winkel verdreht werden um diese Veränderung auszugleichen. Zusätzlich muss die GDD durch verschieben der Prismen eingestellt werden, damit am Ende ein möglichst kurzer Puls zur Verfügung steht. Der Abstand zwischen dem ersten und zweiten sowie dem dritten und vierten Prisma ermöglicht eine Grobkorrektur der GDD. Die Feinkorrektur der GDD kann durch verschieben des zweiten sowie dritten Prismas weiter in oder aus dem Laserstrahl eingestellt werden. Der resultierende gepulste Laserstrahl hat eine Pulsfrequenz von 1 kHz, mit einer Pulsdauer von < 30 fs und einer Pulsenergie von < 800 μJ .

2 Bestimmung der Strahleigenschaften

Die Bestimmung der Strahleigenschaften von Lasern ist mit der DIN EN ISO 11146 [8] genormt. Aufgeteilt ist die Norm in drei Teile. Der erste Teil, die DIN EN ISO 11146-1 beschreibt die Prüfverfahren und Bestimmung der Strahlparameter für stigmatisch und leicht astigmatische Strahlen, die DIN EN ISO 11146-2 die Prüfverfahren und Bestimmung der Strahlparameter für astigmatische Strahlen und die ISO/TR 11146-3 [9] dient als unterstützende Norm, in der unter anderem Hintergrundkorrekturverfahren beschrieben sind. Die Normen dienen als Grundlage für die in diesem Kapitel beschriebenen Auswertung.

2.1 Hintergrundkorrekturverfahren

Die mit einem pixelbasierten Sensor aufgenommene Leistungsdichteverteilung $E_{meas}(x, y)$ kann in zwei Teile aufgeteilt werden. In die "wahre"Leistungsdichteverteilung E(x, y) und in eine Hintergrundleistungsdichteverteilung E_H die von anderen Lichtquellen (z.B. Raumbeleuchtung) verursacht wird:

$$E_{meas}(x,y) = E(x,y) + E_H(x,y)$$
 (2.1)

Diese Hintergrundleistungsdichteverteilung $E_H(x, y)$ kann weiter aufgeteilt werden in einen homogenen Anteil $E_{H,offset}(x, y)$ der den für die gesamte Verteilung konstanten Versatz beschreibt, einen inhomogenen Anteil $E_{H,inh}(x, y)$ der die Neigung des Versatzes beschreibt und einem hochfrequenten Rauschanteil $E_{H,noise}(x, y)$.

$$E_{H}(x,y) = E_{H,offset}(x,y) + E_{H,inh}(x,y) + E_{H,noise}(x,y).$$
(2.2)

Der hochfrequente Rauschanteil $E_{H,noise}(x,y)$ kann und muss normalerweise nicht korrigiert werden, da Rauschkorrekturen das Bild glätten. Glätten bedeutet, dass jeder einzelnen Pixel mithilfe der umgebenden Pixeln einen neuen Wert zugewiesen bekommt. Dadurch kann Rauschen zwar entfernt werden, aber die Intensitätsverteilung wird ebenfalls als gesamtes beeinflusst. Vor allem die Ränder der Intensitätsverteilung würden auf diese Weise nach außen verschoben werden, wodurch die Intensitätsverteilung künstlich vergrößert wird. Dafür reichen im Randbereich bereits kleine Änderungen aus. Der homogene und inhomogene Anteil der durch zusätzliche Lichtquellen verursacht wird kann allerdings nicht ignoriert werden, da Strahldurchmesser und Strahlmittelpunkt aus Integralen der Leistungsdichteverteilung bestimmt werden. Kompensiert man diese Anteile nicht, führt es dazu, dass die Strahldurchmesser größer berechnet werden als sie in Wirklichkeit sind. Es gibt mehrere Methoden die Hintergrundkorrektur durchzuführen. Im Zuge dieser Arbeit wurden 4 in der Norm beschriebene Methoden miteinander verglichen. Die 4 Verfahren lassen sich in Grobkorrektur und Feinkorrektur unterteilen. Zu den Grobkorrekturverfahren gehören "Coarse correction by background map subtraction" (CCBMS) bzw. "Coarse correction by average background subtraction"(CCBAM) und zu den Feinkorrekturverfahren "Fine correction by statistical method"(FCBSM) bzw. "Fine correction by approximation method" (FCBAM).

2.2 Grobkorrektur (Coarse correction)

Um eine Grobkorrektur des Hintergrunds durchzuführen sind Aufnahmen der Intensitätsverteilung aufgrund von externen Lichtquellen notwendig. Diese Aufnahmen werden auch "schwarze Bilder" genannt. Beide Grobkorrekturverfahren (CCBMS & CCBAM) bestimmen aus diesen "schwarze Bildern" den notwendigen Korrekturwert der auf jedes Pixel angewendet wird. Bestimmt wird dieser Korrekturwert grundsätzlich bei beiden Verfahren durch Mittlung von Pixelwerten. Aus diesem Grund ist ein Einzelbild des Hintergrunds nicht ausreichend. Es sollten laut Norm zumindest n > 10 Bilder an jedem Messpunkt aufgenommen werden. Die Messung des Hintergrunds sollte möglichst zeitnah vor oder nach der Lasermessung durchgeführt werden, damit Änderungen des Hintergrundverhältnisse auf ein Minimum zu reduzieren. Sind die Hintergrundverhältnisse konstant ist es möglich erst den Hintergrund zu messen und danach den Laserstrahl.

2.2.1 Coarse correction by background map subtraction (CCBMS)

Bei CCBMS wird jeder einzelne Pixel der "schwarzen Bilder" gemittelt und daraus eine sogenannte Hintergrundkorrekturkarte(background map) erstellt. Diese Hintergrundkarte ist im Grunde nichts anderes als ein Bild bestehend aus den gemittelten Pixel der "schwarzen Bilder". Bestimmt werden diese gemittelten Pixel mit

$$\overline{E_H(x,y)} = \frac{1}{n} \sum_{i=1}^n E_H(x,y)_i.$$
(2.3)

Mit dieser Hintergrundkarte bestehend aus den gemittelten Pixeln kann anschließend die korrigierte Leistungsdichteverteilung mit

$$E(x,y) = E_{meas}(x,y) - \overline{E_H(x,y)}$$
(2.4)

berechnet werden. Da jeder Pixel einzeln gemittelt wird ist es mit dieser Grobkorrektur möglich Einflüsse die nur Teile des Sensors betreffen zu kompensieren. Ein Beispiel dafür ist eine Strahlungsquelle die den Sensor nur in der unteren Hälfte bestrahlt.

2.2.2 Coarse correction by average background subtraction (CCBAM)

Ist der Einfluss der Umgebung über den gesamten Sensor konstant, kann CCBAM verwendet werden. Im Gegensatz zur CCBMS werden bei diesem Verfahren alle Pixel über alle "schwarzen Bilder" gemittelt und daraus ein konstanter Versatzwert mit

$$\overline{E_{H,Offset}(x,y)} = \frac{1}{n \cdot m} \sum_{i=1}^{n} \sum_{x,y=1}^{m} E_H(x,y)_i$$
(2.5)

bestimmt. Dabei ist m die Gesamtanzahl an (x, y) Datenpunkten auf dem Sensor und n die Anzahl der Einzelmessungen. Mit diesem Versatzwert, wird die korrigierte Leistungsdichteverteilung mit

$$E(x,y) = E_{meas}(x,y) - \overline{E_{H,Offset}(x,y)}$$
(2.6)

bestimmt.

2.3 Feinkorrektur (Fine correction)

Feinkorrektur stellt die zweite Stufe der Hintergrundkorrektur dar. Sie ist erforderlich wenn die Grobkorrektur alleine nicht ausreicht Hintergrundeinflüsse zu kompensieren.

2.3.1 Fine correction by statistical method (FCBSM)

Bei diesem Verfahren wird die Hintergrundverteilung $E_{H,Offset}(x, y)$ durch einen durchschnittlichen Versatz $E_{H,Offset}$ und einer Standardabweichung $E_{H,\sigma}$ beschrieben. Im ersten Schritt ist es bei diesem Verfahren notwendig den Versatz und die Standardabweichung zu schätzen. Eine Methode

2 Bestimmung der Strahleigenschaften

um diese Schätzwerte zu generieren ist es sie aus "schwarzen Bildern" zu bestimmen. Für die n
 "schwarzen Bilder" kann mit Gl. 2.3 $E_{H,Offset}$ abgeschätzt werden. Die Standardabweichung kann aus den n Bildern mit

$$E_{H,\sigma} = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (E_{H,Offset,i} - E_{H,Offset})^2}$$
(2.7)

bestimmt werden. Hat nun ein Pixel einen höheren Graustufenwert als

$$E_{i,j} > E_{H,Offset} + n_T E_{B\sigma} \tag{2.8}$$

wird dieser als beleuchtet angesehen und für die Strahldurchmesser und Strahlmittelpunktbestimmung herangezogen. Der Faktor n_T sollte dabei im Bereich $2 < n_T > 4$ liegen. Er wird zum Feinjustieren des Kriteriums für beleuchtete Pixel 2.8 verwendet. Jeder Pixel der Gl. 2.8 nicht erfüllt wird zur Bestimmung des Versatzes herangezogen und ergibt sich aus dem Mittelwert der "dunklen"Pixel. Angewendet wird dieses Verfahren allerdings nicht direkt auf das gemessene Bild sondern auf eine verschmierte Version davon. Eine einfache Methode ein Bild zu verschmieren ist die 2D-Faltung. Bei der 2D-Faltung wird ein zentraler Pixel mit seinen umgebenden Pixel summiert, wobei jeder Pixel eine eigene Gewichtung besitzt. Die Gewichtung jedes einzelnen Pixel liegt zwischen 0 - 1 und hängt davon ab welcher Effekt mit der Faltung erreicht werden soll. Festgelegt wird diese mit der sogenannten Kernelmatrix. Das sind $n \times m$ Matrizen in der die Gewichtungen jedes einzelnen Pixels festgelegt wird. Sie haben die Form

$$K = \begin{pmatrix} a_{1,1} & a_{1,1} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,m}. \end{pmatrix}$$

Um nun ein Bild zu verschmieren benötigt man eine Kernelmatrix in der jeder Pixel dieselbe Gewichtung besitzt. Die Koeffizienten ergeben sich dann zu

$$a_{n,m} = a = \frac{1}{n \cdot m} \tag{2.9}$$

und die Kernelmatrix zu

$$K = \frac{1}{n \cdot m} \begin{pmatrix} 1 & 1 & \cdots & 1\\ 1 & 1 & \cdots & 1\\ \vdots & \vdots & \ddots & \vdots\\ 1 & 1 & \cdots & 1. \end{pmatrix}$$
(2.10)

Diese Form der Kernelmatrix wird auch als Verwisch-filter bezeichnet. Jeder einzelne Pixelwert des verschmierten Bildes lässt sich mathematisch beschreiben mit

$$\tilde{E_{i,j}} = \frac{1}{(n+1)\cdot(m+1)} \sum_{k=i-\frac{n}{2}}^{k=i+\frac{n}{2}} \sum_{l=j-\frac{m}{2}}^{l=j+\frac{m}{2}} E_{k,l}.$$
(2.11)

Wie stark verschmiert wird hängt von der Größe der Kernelmatrix ab. Verwendet man eine 25×25 Kernelmatrix wird aus dem weißen Kreis in Abb. 2.1a ein leicht verwischter. Durch Erhöhung der Anzahl der Elemente die gemittelt werden, kann das Bild weiter verwischt werden (siehe Abb. 2.1c). Um den Einfluss der Kernelmatrix auf das verschmierte Bild zu zeigen wurde das Bild mit einer 25×25 Kernelmatrix, einer 100×100 Kernelmatrix und einer 200×200 Kernelmatrix verschmiert. Der Durchmesser des Kreises wird wächst mit der Größe der Kernelmatrix, wobei die Intensität jedes Abbildung 2.1: Einfluss der Größe einer Kernelmatrix auf die Bildverschmierung



(a) Bild vor Verschmierung



(c) Bild nach Verschmierung mit einer 100×100 Kernelmatrix



(b) Bild nach Verschmierung mit einer 25×25 Kernelmatrix



(d) Bild nach Verschmierung mit einer 200×200 Kernelmatrix

Pixel gleichzeitig abnimmt. Die Größe der gewählten Kernelmatrix ist daher ein wichtiger Faktor. In der Norm ist als Richtwert für die Größe der Kernelmatrix 2% - 5% der Sensorabmessungen angegeben. Auf das verschmierte Bild kann nun zur Bestimmung der unbeleuchteten Pixel die Bedingung

$$\tilde{E}_{i,j} < E_{H,Offset} + n_T \frac{E_{B\sigma}}{\sqrt{(n+1)(m+1)}}$$
(2.12)

herangezogen werden. Den tatsächlichen Versatz erhalten wir indem alle Pixel die die Bedingung Gl.2.12 erfüllen(die unbeleuchteten Pixel) mitteln.

$$\overline{E_{H,Offset,tats}} = \frac{\sum E_{i,j}}{\text{Anzahl der Pixel}}$$
(2.13)

Das korrigierte Bild wird anschließend mit

$$E(x,y) = E_{meas}(x,y) - \overline{E_{H,Offset,tats}(x,y)}$$
(2.14)

bestimmt.

2.3.2 Fine correction by approximation method (FCBAM)

Bei der FCBAM wird der Versatz direkt aus dem gemessenen Bild ermittelt. Eine Voraussetzung für Anwendung dieses Verfahren ist, dass der Durchmesser des Laserstrahls nicht größer ist als das 0.5 fache der Sensordimensionen. Das liegt daran, dass es nicht beleuchtete Bereiche auf dem Sensor geben muss aus denen der Versatz berechnet werden kann. Die Ecken des Sensors eignen sich dafür besonders gut, da diese Bereiche im Normalfall nicht beleuchtet sind. Für die Bestimmung des Versatzes werden N nicht beleuchtete Pixel in einem Bereich $n \times m$ der 4 Ecken gemittelt. Die Größe des Bereichs kann mit 2% - 5% der Sensordimensionen abgeschätzt werden. Die Voraussetzung der Laserstrahlgröße allein reicht aber noch nicht aus um sicher zu stellen, dass diese Methode zulässig ist. Es muss zusätzlich die Bedingung

$$\frac{E_{H,Offset,meas} - E_{H,Offset} | \sqrt{N}}{E_{H\sigma}} < n_T,$$
(2.15)

erfüllt werden. Dabei ist $E_{H,Offset,meas}$ der Versatz bestimmt aus den Ecken, $E_{H,Offset}$ ist der Versatz bestimmt aus den "schwarzen Bilder" mit Gl. 2.3 und n_T wie bei FCBSM zwischen 2 < $n_T > 4$ liegen. Die Funktion ist dieselbe wie in FCBSM.

2.4 Anwendung der Hintergrundkorrekturverfahren

Der Einfluss der Hintergrundkorrekturverfahren auf das gemessene Strahlprofil ist in den folgenden Abbildungen dargestellt. Aufgenommen wurde das Strahlprofil eines He-Ne Lasers. Verwendet wurde dafür eine monochromatische Kamera mit einem Wertebereich von 0-255 in einem verdunkelten Raum. Der Raum konnte allerdings nicht vollständig verdunkelt werden. Die Verwendung des gesamten Wertebereichs für die Skalierung der Pixel führte zu Bildern die auf ersten Blick völlig hintergrundfrei waren. Um die mitgemessenen Hintergrundeinflüsse sichtbar zu machen, war es notwendig den Wertbereich der Graustufenskalierung von 0-255 auf 0-5 zu verringern. Mit dieser neuen Skalierung hoben sich Pixel mit Graustufen > 0 von den schwarzen Pixeln deutlich ab. Abb. 2.2a zeigt ein Messergebnis mit einer Graustufenskalierung von 0-255 was dem gesamten möglichen Wertebereich entspricht und Abb. 2.2b dasselbe Messergebnis allerdings mit einer reduzierten Graustufenskalierung von 0-5. Der Unterschied zwischen den beiden Skalierungen war klar zu erkennen. In Abb. 2.2a war es eine scheinbar hintergrundfreie Aufnahme eines Laserstrahls mit TEM_{00} Profil. Mit dem verkleinerten Wertebereich in Abb. 2.2b wurde der Hintergrund sichtbar und das Strahlprofil war nur mehr schwer zu erkennen. Abb. 2.2c und Abb. 2.2d zeigen dasselbe Profil, nachdem CCBMS auf das Bild angewendet wurde. Eine Gegenüberstellung der beiden verschiedenen Graustufenskalierungen lieferte in der 0-255 Graustufenskalierung keinen sichtbaren Unterschied zum unkorrigierten Profil in Abb. 2.2a. In Abb. 2.2d wird der Effekt den die Grobkorrektur auf das unkorrigierte Bild hatte sehr klar, da nun das Strahlprofil eindeutig erkennbar war. Betrachtet man Abb. 2.2d im Detail fällt auf, dass die Grobkorrektur nicht alles kompensieren konnte. Aus diesem Grund wurde nun noch eine Feinkorrektur nach der Grobkorrektur durchgeführt. Angewendet wurde die Feinkorrektur anhand einer statistischen Methode (FCBSM). Dargestellt ist das Ergebnis in Abb. 2.2e und Abb. 2.2f wieder mit beiden Graustufenskalierungen. Auch im fein korrigierten Bild mit der Farbskalierung 0-255 ist erwartungsgemäß kein Unterschied zum unkorrigierten Bild Abb.2.2a sichtbar. Mit der reduzierten Graustufenskalierung ist eine klare Verbesserung zur Grobkorrektur in Abb. 2.2d erkennbar. Im Idealfall erhält man nach Grob- und Feinkorrektur ein völlig hintergrundfreies Bild auf dem nur das Laserstrahlprofil zu sehen ist. Realistisch bleibt aber ein gewisses Restrauschen übrig, das aufgrund seiner Zufälligkeit nicht gut korrigiert werden kann. Es gibt wie bereits in Kap. 2.1 erwähnt verschiedene Möglichkeiten diese Korrektur durchzuführen. CCBAM als Grobkorrektur kompensierte einen Großteil des Hintergrundes, allerdings nicht so gut wie CCBMS. Aus den Ergebnissen geht hervor, dass die Grobkorrektur mit CCBMS das Rauschen

Abbildung 2.2: Einfluss der Graustufenskalierung und der Korrekturverfahren



(a) Keine Korrektur Graustufenskalierung 0-255



(c) Nach Background Map Substraction Graustufenskalierung 0-255



(e) Nach Statistical Feinkorrektur Graustufenskalierung 0-255



(b) Keine Korrektur Graustufenskalierung0-5



(d) Nach Background Map Substraction Graustufenskalierung 0-5



(f) Nach Statistical Feinkorrektur Graustufenskalierung 0-5

besser filtert als CCBAM. Der Grund dafür ist in Abb.2.2g zu sehen. Das Bild zeigt eine inhomogene Verteilung des Rauschens. Ein häufiger Grund dafür ist, dass die Hintergrundstrahlung mit einem Eintrittswinkel (Angle of incidence AOI) $\neq 0$ auf den Kamerasensor trifft wodurch der Sensor inhomogen beleuchtet wird. CCBAM eignet sich wie bereits beschrieben nur für die Kompensation

2 Bestimmung der Strahleigenschaften



(g) Nach Average Value Substraction Graustufenskalierung 0-5



(h) Nach Statistical Feinkorrektur Graustufenskalierung 0-5

von jenen Einflüssen, die den gesamten Kamerasensor betreffen.

2.5 Bestimmung des Strahldurchmessers aus einer gegebenen Leistungsdichteverteilung

2.5.1 Stigmatische und leicht astigmatische Strahlen

Als Grundlage für die Bestimmung der Laserstrahleigenschaften von stigmatischen bzw. leicht astigmatischen Strahlen dient die DIN EN ISO 11146-1. In ihr sind die Berechnungsverfahren beschrieben um aus einer gegebenen Leistungsdichteverteilung eines Laserstrahls die Laserstrahleigenschaften zu bestimmen. Verwendet wird dafür die Momente erster- bzw- zweiter- Ordnung.

Bestimmung der Momente erster Ordnung einer Leistungsdichteverteilung $\overline{x}, \overline{y}$

Die Momente erster Ordnung einer Leistungsdichteverteilung beschreiben die Position des Strahlzentrums. In den allgemeinen Gleichungen gehen die Grenzen von $-\infty$ bis ∞ , wir ersetzen diese Grenzen mit unseren finiten Integrationsbereich beschränkt durch die Größe des Kamerasensors. Die Momente erster Ordnung können bestimmt werden mit

$$\overline{x}(z) = \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} E(x, y, z) x \, dx \, dy}{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} E(x, y, z) \, dx \, dy}$$
(2.16)

und

$$\overline{y}(z) = \frac{\int \int \int E(x, y, z)y \, dx \, dy}{\int \int \int \int E(x, y, z) \, dx \, dy}$$
(2.17)

 \overline{x} entspricht dem Abstand des Strahlzentrums von der x-Achse und \overline{y} dem Abstand des Strahlzentrums von der y-Achse des gewählten Integrationsbereichs. E(x, y, z) ist dabei der Intensitätswert eines Pixels und x bzw. y der Abstand des Pixels zum Koordinatensystem des gewählten Integrationsbereichs.

2 Bestimmung der Strahleigenschaften

Bestimmung der Momente zweiter Ordnung einer Leistungsdichteverteilung σ_x^2 , σ_y^2 , σ_{xx}^2

Die Momente zweiter Ordnung einer Leistungsdichteverteilung beschreiben die geometrische Ausdehnung eines Strahls und sind gegeben durch

$$\sigma_x^2(z) = \left\langle x^2 \right\rangle = \frac{\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} E(x, y, z)(x - \overline{x})^2 \, dx \, dy}{\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} E(x, y, z) \, dx \, dy}$$
(2.18)

für das Moment zweiter Ordnung in x,

$$\sigma_y^2(z) = \left\langle y^2 \right\rangle = \frac{\int \limits_{-\infty}^{\infty} \int \limits_{-\infty}^{\infty} E(x, y, z)(y - \overline{y})^2 \, dx \, dy}{\int \limits_{-\infty}^{\infty} \int \limits_{-\infty}^{\infty} E(x, y, z) \, dx \, dy}$$
(2.19)

für das Moment zweiter Ordnung in y und

$$\sigma_{xy}^{2}(z) = \left\langle xy \right\rangle = \frac{\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} E(x, y, z)(x - \overline{x})(y - \overline{y}) \, dx \, dy}{\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} E(x, y, z) \, dx \, dy}$$
(2.20)

für das Mischmoment zweiter Ordnung.

Bestimmung des Azimutwinkels φ

Der Azimutwinkel φ beschreibt den Winkel zwischen der x-Achse der Kamera und der Hauptachse des Strahlprofils die am nächsten dazu liegt. Der Winkel der Kamera zum Laser wird durch die Montage der Kamera festgelegt. Bei einfach astigmatischen Strahlen kann der Azimutwinkel bestimmt werden mit

$$\varphi(z) = \frac{1}{2} \arctan\left(\frac{2\sigma_{xy}^2}{\sigma_x^2 - \sigma_y^2}\right)$$
(2.21)

für Leistungsdichteverteilungen in denen $\sigma_x^2 \neq \sigma_y^2$ ist. Begrenzt ist der Azimutwinkel in diesem Fall auf $-\frac{\pi}{4} < \varphi < \frac{\pi}{4}$. Für den Fall das $\sigma_x^2 = \sigma_y^2$ ist, kann der Azimutwinkel bestimmt werden mit

$$\varphi = \operatorname{sgn}(\sigma_{xy}^2) \frac{\pi}{4} \tag{2.22}$$

und $\operatorname{sgn}(\sigma_{xy}^2)$ mit

$$\operatorname{sgn}(\sigma_{xy}^2) = \frac{\sigma_{xy}^2}{|\sigma_{xy}^2|}.$$
(2.23)

Der Azimutwinkel ist in diesem Fall entweder $-\frac{\pi}{4}$ oder $\frac{\pi}{4}$.

Bestimmung der Strahlabmessungen in Hauptachsenrichtung $d_{\sigma x}, d_{\sigma y}$

Die Bestimmung der Strahlabmessungen erfolgt in Richtung der Hauptachsenrichtung. Für Leistungsdichteverteilungen in denen $\sigma_x^2\neq\sigma_y^2$ können die Laserstrahlabmessungen bestimmt werden mit

$$d_{\sigma x}(z) = 2\sqrt{2} \left\{ \left(\sigma_x^2 + \sigma_y^2 \right) + \gamma \left[\left(\sigma_x^2 - \sigma_y^2 \right)^2 + 4 \left(\sigma_{xy}^2 \right)^2 \right) \right]^{\frac{1}{2}} \right\}^{\frac{1}{2}}$$
(2.24)

und

$$d_{\sigma y}(z) = 2\sqrt{2} \left\{ \left(\sigma_x^2 + \sigma_y^2 \right) - \gamma \left[\left(\sigma_x^2 - \sigma_y^2 \right)^2 + 4 \left(\sigma_{xy}^2 \right)^2 \right]^{\frac{1}{2}} \right\}^{\frac{1}{2}}.$$
 (2.25)

Dabei ist

$$\gamma = \operatorname{sgn}(\sigma_x^2 - \sigma_y^2) = \frac{\sigma_x^2 - \sigma_y^2}{|\sigma_x^2 - \sigma_y^2|}$$
(2.26)

Ist $\sigma_x^2=\sigma_y^2$ vereinfachen sich die Gleichungen zu

$$d_{\sigma x}(z) = 2\sqrt{2} \left(\sigma_x^2 + \sigma_y^2 + |2\sigma_{xy}^2| \right)^{\frac{1}{2}}$$
(2.27)

und

$$d_{\sigma y}(z) = 2\sqrt{2} \left(\sigma_x^2 + \sigma_y^2 - |2\sigma_{xy}^2| \right)^{\frac{1}{2}}.$$
 (2.28)

Ist die Elliptizität ε

$$\varepsilon = \frac{d_{\sigma,max}}{d_{\sigma,min}} \tag{2.29}$$

größer als 0.87 darf der Laserstrahlquerschnitt als symmetrisch betrachtet werden und kann mit

$$d_{\sigma}(z) = 2\sqrt{2} \left(\sigma_x^2 + \sigma_y^2\right)^{\frac{1}{2}}$$
(2.30)

berechnet werden.

2.6 Anwendung auf Einzelaufnahme eines Strahlprofils mit pixelbasierten Sensor

Um nun aus einer Einzelaufnahme eines Strahlprofils alle bisher beschriebenen Parameter zu berechnen, ist ein geeigneter Integrationsbereich notwendig. Festgelegt ist dafür in der Norm, dass der Integrationsbereich dreimal so groß sein sollte wie die Strahlabmessungen. Aus diesem Grund ist auch die Hintergrundkorrektur essentiell, da alle Pixel mit Werten $\neq 0$ in die Momentenberechnung mit einfließen. Gleichzeitig ist diese Anforderung an den Integrationsbereich eine Limitierung des mit einer bestimmten Kamera messbaren maximalen Strahlabmessungen. Wird zum Beispiel eine Kamera mit einem $6 \,\mathrm{mm} \times 6 \,\mathrm{mm}$ Sensor verwendet darf der Strahldurchmesser an keiner Stelle $> 2 \,\mathrm{mm}$ sein. Mit einer Aufnahme eines Strahlprofils die die Bedingung erfüllt siehe Abb. 2.3a muss noch ein passender Integrationsbereich gewählt werden. Dafür wird im ersten Schritt der Integrationsbereich händisch geschätzt. Die mit diesem Integrationsbereich berechneten Strahlabmessungen können aber, wenn man nicht zufällig den idealen Integrationsbereich auswählt, ungenau sein. Die Lösung dafür ist die Strahlabmessungen iterativ zu bestimmen. Wir legen einen ersten Schätzwert für die Position des Strahls und dessen Strahlabmessungen fest siehe (Abb. 2.3b). Mit diesem geschätzten Integrationsbereich werden die Strahlabmessungen berechnet, anschließend werden aufgrund dieser Strahlabmessungen ein neuer Integrationsbereich festgelegt und wieder die Strahlabmessungen berechnet. Dieser Schritt wird wiederholt bis das Konvergenzkriterium

$$\left. \begin{array}{c} d_{\sigma,x,alt} - d_{\sigma,x,neu} \\ d_{\sigma,y,alt} - d_{\sigma,y,neu} \\ |\overline{x}_{alt} - \overline{x}_{neu}| \\ |\overline{y}_{alt} - \overline{y}_{neu}| \end{array} \right\} \qquad \leq k_{convergence}$$

$$(2.31)$$

erfüllt ist.

2 Bestimmung der Strahleigenschaften

Abbildung 2.3: Auswahl des geschätzten Integrationsbereichs



(a) Strahlprofil aufgenommen mit einem pixelbasierten Sensor



(b) Strahlprofil mit geschätzten Integrationsbereich

2.7 Bestimmung von Taillenlagen, Divergenzwinkeln und Beugungsmaßzahlen

Für die Bestimmung von Taillenlagen, Divergenzwinkeln und Beugungsmaßzahlen sind mehrere Messpunkte entlang der Ausbreitungsrichtung (z-Achse) des Strahls notwendig. Vorgegeben wird in der Norm, dass die Hälfte der Messpunkte innerhalb einer Rayleighlänge der Strahltaille und die andere Hälfte jenseits von zwei Rayleighlängen liegen sollten. Die Anzahl der Messpunkte sollte ≥ 10 sein. Bestimmt werden die Parameter indem eine hyperbolische Kurve an die quadrierten Strahlabmessungen der Form

$$d_{\sigma x.y}^2(z) = a + bz + cz^2.$$
(2.32)

angepasst wird siehe Abb. 2.4. Aus den Koeffizienten a, b und c (bzw. $a_x, a_y, b_x, b_y, c_x, c_y$) der



Abbildung 2.4: An die Strahldurchmesser in x-Richtung angepasste hyperbolische Kurve

2 Bestimmung der Strahleigenschaften

angepassten Hyperbel kann die Taillenlage mit

$$z_{0x,0y} = \frac{-b_{x,y}}{2c_{x,y}},\tag{2.33}$$

die Strahlabmessungen der Strahltaille mit

$$d_{\sigma 0x,\sigma 0y} = \frac{1}{2\sqrt{c_{x,y}}}\sqrt{4a_{x,y}c_{x,y} - b_{x,y}^2},$$
(2.34)

die Divergenzwinkel mit

$$\Theta_{\sigma x,\sigma y} = \sqrt{c_{x,y}},\tag{2.35}$$

die Rayleigh Länge mit

$$z_{Rx,Ry} = \frac{1}{2c_{x,y}}\sqrt{4a_{x,y}c_{x,y} - b_{x,y}^2}$$
(2.36)

und die Beugungsmaßzahl mit

$$M_{x,y}^2 = \frac{\pi}{8\lambda} \sqrt{4a_{x,y}c_{x,y} - b_{x,y}^2}$$
(2.37)

bestimmt werden.

Die verfügbare Fläche vor dem FSL ist begrenzt durch opto-mechanische und mechanische Komponenten, die für Oberflächenstrukturierung von Metallen verwendet werden (siehe Abb. 3.1). Aus



Abbildung 3.1: Anordnung opto-mechanischer Elemente vor dem FSL

diesem Grund war es notwendig den Messaufbau so kompakt wie möglich zu gestalten. Entworfen wurde der Messaufbau mithilfe von CAD, wodurch sehr schnell und effizient Änderungen vorgenommen werden konnten. Der entworfene Messaufbau wurde in einem Vorversuch mit einem He-Ne Laser getestet und mit der gewonnenen Erfahrung überarbeitet. Auf diese Weise wurde der Messaufbau zweimal überarbeitet.

3.1 Messaufbau 1

Messaufbau 1 besteht aus einer Aluminium Grundplatte (10) die auf einem vertikalen Steher aus Stahl montiert wurde, einem silberbeschichteten Einkopplungsspiegel (1), 4 rechtwinkligen Prismen (2) - (5), einer Fokussierlinse mit Brennweite 500 mm (6), einer Kamera mit CMOS Sensor (8) und einem Linearschlitten mit maximalen Verfahrweg 23 mm (9) angetrieben mithilfe einea Schrittmotors (7). Der Laserstrahl durchläuft den Messaufbau beginnend mit dem silberbeschichteten Spiegel (1). Er übernimmt das Einkoppeln des Laserstrahls in das Messsystem und wurde auf einem justierbaren Spiegelhalter montiert. Nach dem Einkopplungsspiegel wurde der Laserstrahl durch Reflexion, an den Prismen (2) - (5), umgelenkt und gleichzeitig aufgrund der Reflexionsbzw. Transmissionseigenschaften der Prismen abgeschwächt. Bestimmt werden die Reflexions- und

Abbildung 3.2: Messaufbau Version 1



(a) Übersicht Messaufbau Version 1



Transmissionskurven vom Werkstoff der Prismen. Für die Bandbreite des FSL von 40 nm mit einer zentralen Wellenlänge von 800 nm und dem He-Ne Laser mit einer Wellenlänge von 632.8 nm sind Prismen aus BK7 geeignet, da sie in diesem Bereich nahezu konstante Eigenschaften haben (siehe Abb . 3.9). Jede Prisma reflektiert $\sim 10\%$ der auftreffenden Strahlung. Dadurch ergibt sich bei serieller Reflexion an vier Prismen ein Abschwächungsfaktor von 10^{-4} . Die Prismen wurden in eigens dafür konstruierten Prismahaltern montiert. Sie bestehen aus zwei Platten die über zwei Schrauben miteinander verbunden sind siehe Abb. 3.2b. Das Prisma befindet sich zwischen den beiden Platten und kann über die Schrauben fixiert werden. Auf der Rückseite des Prismahalters sind zusätzlich matt schwarz lackierte Strahlabfangbleche montiert, die transmittierte Strahlung abfangen sollten. Das letzte optische Element bildete eine Fokussierlinse (6) aus BK7 die den Laserstrahl auf die Kamera (7) fokussierte. Die Fokussierung war notwendig, da der Rohstrahl des FSL größer ist als der verbaute Sensor in der Kamera (siehe Kap. 3.4.6. In der Norm wird diese durch die Fokussierlinse erzeugte Strahltaille als künstliche Strahltaille bezeichnet. Die Kamera wurde auf einem Linearschlitten montiert der von einem 24 V Schrittmotor angetrieben wurde. Der Verfahrweg pro Schritt des Schrittmotors ist aus in einem Versuch ermittelt worden. Dabei wurde die Position z_0 des Linearschlittensschlitten zu einer Referenzkante (Hinterkante der Linearschlitten) gemessen. Nach separaten Verfahren des Linearschlittens um 1000 Schritte, wurde der Abstand z_{1000} zur Referenzkante gemessen. Der Verfahrweg pro Schritt ergibt sich zu

$$\Delta z_i = \frac{|z_0 - z_{1000}|}{1000}.\tag{3.1}$$

Die Messung wurde 20 Mal wiederholt und aus den Einzelergebnissen der Mittelwert mit

$$\Delta z = \frac{1}{n} \sum_{i=1}^{n} \Delta z_i \tag{3.2}$$

gebildet. Daraus ergab sich ein Verfahrweg pro Schritt von $1.8\mu m$. Gesteuert wird der Schrittmotor über ein Arduino UNO Rev3 Microcontroller-Board. Als Treiberstufe wird ein Arduino Motor Shield Rev3 verwendet. Die Grundplatte bildet eine unabhängige Messebene, wodurch der Messaufbau ohne großen Aufwand zum Vermessen anderer Laserquellen verwendet werden kann. Die Höhe der Grundplatte bezogen auf die Montageoberfläche des Stehers ist von 375 mm bis 415 mm einstellbar.

3.1.1 Vorversuch mit einem He-Ne Laser

Bevor das Messsystem am FSL getestet wurde, ist ein Vorversuch mit einem He-Ne Laser durchgeführt worden. Der Laser hat eine Zentral-Wellenlänge von $632.8 \,\mathrm{nm}nm$, einen $1/e^2$ Durchmesser

von 1.02 mm, und eine Ausgangleistung von 7 mW. Für den Versuch wurde der Laserstrahl wie in Abb. 3.2a dargestellt über den Einkopplungsspiegel (1) ins Messsystem eingekoppelt und mit der Kamera (7) das Strahlprofil aufgenommen. Die Ziele des Vorversuchs waren es das Messprogramm zu testen, die Funktionalität des Messaufbaus sicher zu stellen und die optischen Elemente so genau wie möglich auszurichten. Die Ausrichtung jedes einzelnen optischen Elements war einfach durchzuführen, aber alle Komponenten im Zusammenspiel miteinander auszurichten war sehr arbeitsintensiv. Ausgerichtet wurden die Komponenten in der Reihenfolge wie der Laserstrahl sie durchläuft (1) - (6). Der Prozess wurde solange wiederholt bis der Laserstrahl den Kamera Sensor traf.

3.1.2 Ergebnisse des Vorversuchs mit Messaufbau 1

Die Ergebnisse des Vorversuchs zeigten, dass es durch die geometrische Anordnung der Prismen und Bleche zu Reflexionen an den Strahlabfangblechen kam. Die Reflexionen überlagerten sich mit dem Messignal, wodurch die Messergebnisse unbrauchbar wurden.

3.2 Messaufbau 2

Messaufbau 2 hatte denselben Grundaufbau wie Messaufbau 1, allerdings wurden die Strahlabfangbleche entfernt und durch Strahlabsorber (11) ersetzt siehe Abb. 3.3a. Strahlabsorber sind Körper

Abbildung 3.3: Messaufbau Version 2 und Strahlfalle

(a) Übersicht Messaufbau Version 2



(b) Dreiviertelschnitt einer Strahfalle

die Laserstrahlen durch Absorption in Wärme umwandeln sollen. Die einfachste Ausführung für Absorber sind passiv oder aktiv gekühlte Platten. Eine weitere Möglichkeit ist es die Strahlabsorber als "Fallen" zu konzipieren in die der Laserstrahl durch eine Öffnung eintreten, aber durch geschickt gewählte Geometrien im inneren der Falle nicht mehr austreten bevor er vollständig absorbiert wird. Wie die Plattenabsorber können auch die Strahlfallen passiv oder aktiv gekühlt ausgeführt werden. Die Strahlabsorber für Messaufbau 2 sind als passiv gekühlte Strahlfallen konzipiert worden (siehe Abb. 3.3b). Sie bestehen aus zwei Teilen. Einem zylindrischen Absorbtionskörper mit der Eintrittsbohrung und einer Abschlussplatte mit Spitzkegel. Der Spitzkegel absorbiert einen Teil der Strahlung und reflektiert den restlichen Anteil in Richtung der Absorptionskörperwände, die wiederum Strahlung absorbieren und reflektieren. Auf diese Art wird der Strahl vielfach im Körper reflektiert bis er vollständig absorbiert ist. Um die Absorptionsfähigkeit der Oberflächen zu steigern wurden diese zusätzlich matt schwarz lackiert.

3.2.1 Ergebnisse des Vorversuchs mit Messaufbau 2

Die Ergebnisse des Vorversuchs mit Messaufbau 2 zeigten, dass die transmittierte Strahlung innerhalb des Prismas reflektiert wurde. Das führte dazu, dass der an der Rückseite des Prismas reflektierte Laserstrahl das Prisma an der Frontfläche als Sekundärstrahl seitlich verschoben aber parallel zum Primärstrahl austrat siehe Abb .3.4. Dieser parallelen Sekundärstrahlen traten bei jedem Prisma auf, wodurch an der Fokussierlinse mehrere örtlich verschobene aber annähernd parallele Strahlen auftrafen. Diese wurden von der Linse auf die Kamera fokussiert, wodurch es zu Interferenzen



Abbildung 3.4: Reflexionen im Prisma

im gemessenen Strahlprofil kam. Die ungewünschten Laserstrahlen konnten im Vorversuch mittels Schablone abgeblockt werden was zu einer Unterdrückung der Interferenzen führte. Zu sehen ist der Einfluss dieser sekundären Reflexionen in Abb. 3.5a. Abb. 3.5b zeigt das Messergebnis mit unterdrückten Sekundärstrahlen. Eine weitere Möglichkeit der Veranschaulichung von Interferenzen ist

Abbildung 3.5: Laserstrahlprofil



die Anwendung eine 2D FastFourierTransformation(FFT) wodurch ein Bild in seine Spektralanteile zerlegt wird siehe Abb. 3.6a und Abb. 3.6b. In den Spektren wurden die Intensitäten logarithmisch



Abbildung 3.6: Ortsfrequenzspektrum eines Laserstrahlprofils

skaliert als Grauwerte im Bereich von 0 – 255 und in x- und y-Richtung die Ortsfrequenzen aufgetragen. Durch die logarithmische Skalierung wurden die Hauptanteile eindeutig sichtbar. Abb. 3.6b der Messung ohne Interferenzen zeigte, dass sich das Strahlprofil hauptsächlich aus niedrigen Ortsfrequenzen zusammensetzt. Verglichen mit Abb. 3.6a dem Strahlprofil mit Interferenzen fiel auf, dass die zentralen Teile beider Spektren beinahe identisch waren. Sie unterschieden sich nur über zusätzliche Anteile mit höhere Ortsfrequenz in x-Richtung. Im Strahlprofil siehe Abb. 3.5a waren diese als Streifen in y-Richtung erkennbar. Damit die Sekundärstrahlen in Messaufbau 2 mit einer Schablone unterdrückt werden konnten mussten die Strahlabmessungen klein sein. Der He-Ne Laser hat einen Strahldurchmesser von 1.02 mm wodurch die Sekundärstrahlen unterdrückt werden konnten. Im Gegensatz dazu hat der FSL laut Datenblatt einen Durchmesser von 15 mm, wodurch der Sekundärstrahl sich bereits vor der Fokussierlinse mit dem Primärstrahl überlagerte und eine Schablone keine Option darstellte.

3.3 Messaufbau 3

Um die Interferenzerscheinungen zu eliminieren wurde Messaufbau 3 neu konzipiert. Das bisher verwendete Konzept bei dem die Abschwächung des Laserstrahls durch Reflexion an Prismen erreicht wurde ist verworfen worden. Ersetzt wurden sie mit Neutraldichte Filter (ND-Filter) (12), die deren



Abbildung 3.7: Messaufbau Version 3

Funktion übernahmen siehe Abb. 3.7. Für die ND-Filter wurden Filter mit einer optischen Dichte von 5.0 bzw. 4.0 verwendet. Der zuvor am Anfang des Laserstrahls platzierte Einkopplungsspiegel wurde ans Ende verschoben und dazu verwendet den Laserstrahl auf die Kamera auszurichten.

3.3.1 Ergebnisse aus dem Vorversuch mit Messaufbau 3

Die Ergebnisse des Vorversuchs mit Messaufbau 3 zeigten ein Interferenzfreies Strahlprofil.

3.4 Verwendete Hardware und Software

3.4.1 Unbeschichtetes rechtwinkliges Prisma (Thorlabs PS908) [10]

Das Prisma PS908 von Thorlabs ist ein unbeschichtetes rechtwinkliges Prisma aus BK7 mit einer Kantenlänge von 20 mm und einer Höhe von 20 mm. Es besitzt im Wellenlängenbereich von 600 nm



Abbildung 3.8: Rechtwinkliges Prisma PS908 [11]

- 1000 nm nahezu konstante Transmissionseigenschaften (siehe Abb .3.9), wodurch es sich sowohl für den Vorversuch mit dem He-Ne Laser, als auch den Versuch am FSL eignet.



Abbildung 3.9: Transmissionskurve von unbeschichtetem BK7 [12]

3.4.2 Neutraldichtefilter ND-Filter (Thorlabs NE40B-B, NE50B-B) [13]

Verwendet wurden OD 4.0 und OD 5.0 Filter vom Typ NE40B-B und NE50B-B der Firma Thorlabs. Es sind Antireflektionsbeschichtete Filter die für einen Wellenlängen Bereich von 650 nm - 1050 nm nahezu konstante Transmissionseigenschaften besitzt. Aufgrund der Beschichtung ist bei der Handhabung und Reinigung besondere Vorsicht geboten um Beschädigungen der Beschichtung



Abbildung 3.10: ND Filter NE40B-B, NE50B-B [14]

zu vermeiden. Die Transmissionskurven für OD 2.0 - OD 6.0 sind in Abb. 3.11 abgebildet. Relevant für diese Arbeit sind die grüne(OD 4.0) und die gelbe (OD 5.0) Kurve. Das OD 4.0 Filter hat bei einer mittleren Wellenlänge von 800 nm eine Transmissionsrate von ca. 0.3% und das OD 5.0 Filter ca. 0.09%. Verwendet wurden die Filter um die Fluenz des Lasers soweit abzuschwächen das die Pixel der Kamera weder zerstört noch gesättigt werden.



Abbildung 3.11: ND-Filter Transmissionskurven für OD 2.0 - OD 6.0 [15]

3.4.3 Schrittmotor (Sanyo Denki 103H5205-5240) [16]

Der zweipolige Schrittmotor vom Typ 103H5205-5240 Sanyo Denki besitzt bei einem Nennstrom von $1 \,\mathrm{A}/Phase$ ein Haltemoment von $0.265\,\mathrm{N\,m}$. Wie bei allen E-Motoren hängt das verfügbare



Abbildung 3.12: Schrittmotor Sanyo Denki 103H5205-5240 [17]

Drehmoment von der Drehzahl ab. Bis zu einer Drehzahl von 100 min^{-1} ist er in der Lage das maximale Drehmoment bereitzustellen siehe Abb. 3.13. Er wurde verwendet um die Linearschlitten anzutreiben. Gesteuert wurde er über ein Arduino UNO Rev3 Microcontroller-Board mit zusätzlichem Arduino motor shield Rev3 Treiberboard direkt aus MATLAB.



Abbildung 3.13: Momentenverlauf Sanyo Denki 103H5205-5240 [18]

3.4.4 Steuerboard (Arduino UNO Rev3) [19]

Das Arduino UNO Rev3 board ist ein sehr einfach zu programmierendes Board das viele Möglichkeiten in seiner Verwendung bietet. Da das zu erstellende Messprogramm in MATLAB geschrieben



Abbildung 3.14: Arduino UNO Rev3 [20]

wurde eignete sich dieses Board in Verbindung mit einem Arduino Motor Shield Rev3 ideal, weil es bereits vorgefertigte Toolboxen gibt um eine Verbindung mit dem Board und Periphere Hardware wie zum Beispiel einen Motor herzustellen. Es wurde auch für die Überwachung eines Endschalters verwendet der sicher stellt, dass der Linearschlitten nicht den Rand erreicht.

3.4.5 Treiberboard (Arduino Motor Shield Rev3) [21]

Mit dem Arduino Motor Shield Rev3 können sowohl Schrittmotoren als auch Gleichstrommotoren angesteuert werden. Es ist direkt kompatibel mit dem Arduino UNO Rev3 und bietet die Möglichkeit direkt aus MATLAB heraus über das Arduino UNO Rev3 board verwendet zu werden. Es wurde verwendet um den Schrittmotor anzutreiben.



Abbildung 3.15: Arduino Motor Shield Rev3 [22]

3.4.6 Kamera (Basler ace GigE acA1300-60gmNIR) [23]

Die Basler ac
e GigE ac A1300-60gmNIR ist eine NIR Kamera mit $1.2\,MP$ und einer maximalen
Bildrate von 60 Bildern pro Sekunde. In ihr verbaut ist ein e2v EV
76C661 mono CMOS Sensor mit einer Auflösung von 1282 x 1026 (BxH Pixel), einer Pixelgröße von
 $5.3\,\mu\rm{m}$ x $5.3\,\mu\rm{m}$ und sowohl global als auch progressive shutter. Global shutter bedeutet das alle Pixel zum selben Zeitpunkt ausgelesen werden und progressive shutter, dass eine Pixelreihe nach der anderen ausgelesen wird. Die Verbindung zur Kamera besteht über Ethernet. Ausgelöst wird die Kamera entweder über einen Software Trigger oder einen externen Hardware Trigger. Bei der Inbetriebnahme sollten immer die vom Hersteller zur Verfügung gestellten Anleitungen verwendet werden um Verbindungsproblemen mit der Kamera vorzubeugen.



Abbildung 3.16: Basler ace GigE acA1300-60gmNIR [24]

3.4.7 MATLAB R2020b

MATLAB ist eine Software zur Analyse und Visualisierung von Daten, entwickeln von Algorithmen, erstellen von Benutzeroberflächen und Steuern und auslesen von Hardware. Es bietet sehr viele Toolboxen mit vorgefertigten Funktionen an die immer weiterentwickelt bzw. um neue Funktionen ergänzt werden. Jeder Entwickler hat zusätzlich die Möglichkeit seine Programme oder Toolboxen mit anderen zu Teilen wodurch die Auswahl an Toolboxen stetig steigt. Mit ihm wurde die gesamte Software entwickelt. Das umfasst das Steuern der Hardware, das Auswerten der Daten, das Visualisieren der Daten, und der Bereitstellung einer Benutzeroberfläche mit der jede Funktion bedient werden kann.
Im Vordergrund der Arbeit stand das Erstellen eines Messprogramms mit dem automatisiert die Strahlparameter auf Basis der DIN EN ISO-11146 gemessen, ausgewertet und angezeigt werden können. Eine weitere Anforderung an das Prüfprogramm war die Implementierung einer Livemessung, um einen möglichen Einfluss der Laserjustage auf das Strahlprofil sichtbar zu machen. Zur Eingabe der benötigten Prüfparameter musste eine anwenderfreundliche Benutzeroberfläche geschaffen werden. Um alle diese Anforderungen umzusetzen wurde dabei die Programmierumgebung MATLAB (MathWorks) genutzt. Es bietet zusätzlich zu seiner Grundfunktionalität prozedural mathematische Berechnungen durchzuführen und Benutzeroberflächen zu erstellen, vorgefertigten Bibliotheken für Bildverarbeitung und steuern bzw. auslesen von externer Hardware.

4.1 Klassenaufbau

Die Benutzeroberfläche und deren Elemente, im englischen auch "Frontend" genannt, wurden vollständig von der Auswertungsfunktionalität die im Hintergrund arbeitet, im englischen "Backend" genannt, getrennt. Eine solche Trennung bietet den Vorteil, dass es möglich ist Änderungen oder zusätzliche Funktionalität nachträglich einzubauen ohne gegenseitige Beeinflussung von Frontend und Backend. MATLAB arbeitet grundsätzlich prozedural, unterstützt aber auch eine eigene Form der objektorientierten Programmierung die ich im Backend dazu nutzte eigene Datenelemente als Klassen zu definieren. Die Klassen sind hierarchisch aufeinander aufgebaut, wobei jede Klasse auch



Abbildung 4.1: Klassendiagramm Measurement



Abbildung 4.2: Klassendiagramm Background

alleinstehend verwendet werden kann. Die oberste Klasse bildet die Klasse Measurement. Sie beinhaltet alle Daten einer Messung und besteht aus vier untergeordneten Klassen (siehe Abb.4.1). Die vier untergeordneten Klassen sind die Klasse MeasurementData in der die Messdaten enthalten sind, die Klasse Background in der die Hintergrunddaten enthalten sind, die Klasse MeasurementSettings in der die Messparameter gespeichert sind und die Klasse Result in der ausgewertete Messdaten enthalten sind.

MeasurementData	
+ measurementDataSet : Array <singlemeasurementpoint></singlemeasurementpoint>	
+ fx : cfit	Measurement
+ z0x : double	1
+ dSigma0x : double	SingleMeasurement
+ thetaSigmaX : double	
+ zRx : double	
+ MsquaredX : double	
+ fy : cfit	
+ z0y : double	
+ dSigma0y : double	
+ thetaSigmaY : double	
+ zRy : double	
+ MsquaredY : double	
+ zPos : Array <double></double>	
+ dWx : Array <double></double>	
+ dWy : Array <double></double>	
+ reset() : void	
+ getMeasurementDataSet() : Array <singlemeasurementpoint></singlemeasurementpoint>	
+ getZ0X() : double	
+ getZ0Y() : double	
+ getDSigma0x() : double	
+ getDSigma0y() : double	
+ getThetaSigmaX() : double	
+ getThetaSigmaY() : double	
+ getZrX() : double	
+ getZrY() : double	
+ getMsquaredX() : double	
+ getMsquaredY() : double	
+ getfxFit() : cfit	
+ getfyFit() : cfit	
+ getzPos() : Array <double></double>	
+ getdWx() : Array <double></double>	
+ getdWy() : Array <double></double>	
+ backgroundCorrectionMeasurementData(coarseMethod : String, fineMethod : String,background, kernelSize : double, ntFactor : double, manualValue : double) : MeasurementData	
+ iso11146Part1MeasurementData(centerGuessX : double, centerGuessY : double integAreaGuessX : double, integAreaGuessY : double, convergeKriterium : double, faktor : double : lambda : double, pixelsize : double) : MeasurementData	
+ addSingleMeasurementPoint(value : SingleMeasurementPoint) : void	

Abbildung 4.3: Klassendiagramm MeasurementData

Die Klasse Background (siehe Abb.4.2) besteht nur aus einer Liste von SingleMeasurementPoint

Objekten, wobei jedes Listenelement einen der Messpunkte darstellt. Auf die Klasse *SingleMeasurementPoint* wird noch im Detail eingegangen. Die Methoden der Klasse *Background*, dienen der Datenbereitstellung für die Hintergrundkorrektur, die ein wichtiger Bestandteil der Auswertung ist.

Die Klasse *MeasurementData* ist das Herzstück der Messung (siehe Abb.4.3). Es enthält, wie bereits die Klasse *Background*, ebenfalls eine Liste von *SingleMeasurementPoint* Objekten, aber zusätzlich dazu noch alle übergreifenden Strahlparameter die für ein Messergebnis bestimmt werden müssen. Die Methoden dieser Klasse beinhalten sowohl die wichtigsten Berechnungsoperationen der Hintergrundkorrektur und der Parameterbestimmung laut DIN EN ISO 11146, als auch die notwendigen GET Funktionen um alle Daten abzurufen.

SingleMeasurementPoint	FgetW_XY_std() : double
+ dataSet : Array <singlemeasurement></singlemeasurement>	+ getW_X_squared_std() : double
+ zPos : double	+ getW_Y_squared_std() : double
+ meanImage : SingleMeasurement	+ getAzimutPhi_mean() : double
+ dWx_mean : double	+ getAzimutPhi_std(): double
+ dWy_mean : double	+ getBeamCenterX_mean() : double
+ azimutPhi_mean : double	+ getBeamCenterY_mean() : double
+ beamCenterX_mean : double	+ showImage(imageIndex : int) : void
+ beamCenterY_mean : double	+ showAllImages() : void
+ W_X_mean : double	+ showImageWithImTool(imageIndex : int) : void
+ W_Y_mean : double	+ calcMeanImage() : void
+ W_XY_mean : double	+ backgroundCorrectionSingleMeasurementPoint(coarseMethod : String, fineMethod : String, background : Background
+ W_X_squared_mean : double	kernelSize : double, ntFactor : double, k : int, manualValue : double) - SindeMeasurementPoint
+ W_Y_squared_mean : double	+ iso11146Part1SingleMeasurementPoint(centerGuessX : double,
+ dWx_std : double	centerGuessY : double, integAreaGuessX : double, integAreaGuessY : double, convergeKriterium : double, faktor : double,
+ dWy_std : double	pixelsize : double) : SingleMeasurementPoint
+ azimutPhi_std : double	- setMeanImage(value : SingleMeasurement) : void
+ beamCenterX_std : double	
+ beamCenterY_std : double	1* 1 MeasurementData
+ W_X_std : double	
+ W_Y_std : double	SingleMeasurement
+ W_XT_std : double	1* 1 Backaround
+ W_A_squared_std : double	
TT_1_0quared_bio : double	-
+ getDataSet() : Array <singlemeasurementpoint></singlemeasurementpoint>	
+ getZPos() : double	
+ getz+os() : double	
+ geuveanimage() : SingleMeasurement	
+ neldWy mean() : double	
+ cetW(X mean() : double	
+ getW_Y_mean() : double	
+ getW XY mean() : double	
+ getW X squared mean(): double	
+ getW_Y_squared_mean(): double	
+ getdWx_std() : double	
+ getdWy_std() : double	
The second s	
+ getW_X_std() : double	

Abbildung 4.4: Klassendiagramm SingleMeasurementPoint

Die Klasse *SingleMeasurementPoint* (siehe Abb.4.4) repräsentiert einen einzelnen Messpunkt und besitzt einen sehr ähnlichen Aufbau wie die Klasse *MeasurementData*. Sie besteht aus einer Liste von *SingleMeasurement* Objekten die eine Einzelmessung beschreiben und den aus den Einzelmessungen bestimmten gemittelten Strahlparametern.

Die Klasse SingleMeasurement (siehe Abb.4.5) ist das unterste Element des hierarchischen Aufbaus und stellt eine Einzelmessung dar. Darin enthalten ist ein aufgenommenes Bild gespeichert in Form einer $n \times m$ Matrix in der die Graustufenwerte jedes einzelnen Pixels hinterlegt sind und den berechnete Strahlparametern einer Einzelmessung.



Abbildung 4.5: Klassendiagramm SingleMeasurement

4.2 Programmabläufe und GUI

Bisher wurden nur die Klassen und ihre Beziehungen zueinander beschrieben. Als nächstes werden jetzt die Abläufe in dem umgesetzten Messprogramm anhand von Ablaufdiagrammen beschrieben. Eine der geforderten Funktionen ist die Automatisierte Messung der Strahlcharakteristik eines Laserstrahls. Der Ablauf einer Messung wurde auf mehrere Schritte aufgeteilt (siehe Abb.4.8).

Der erste Schritt einer Messung ist das Auslesen der Messparameter aus einer grafischen Benutze-

roberfläche (GUI). Die Messparameter, die dafür festgelegt werden müssen, sind das Messgitter, die Anzahl der Messungen pro Messpunkt und die Belichtungszeit (ExposureTime) der Kamerapixel.

4.2.1 Definition des Messgitters

Für die Festlegung des Messgitters in Ausbreitungsrichtung (Z-Achse) des Laserstrahls (siehe Abb. 4.6) wurden zwei Konfigurationsmethoden gefordert. Eine Konfigurationsmethode die das Messgitter symmetrisch um einen zentralen Punkt verteilt und eine zweite die das Messgitter ausgehend vom Nullpunkt des Linearschlittens auf einen einstellbaren Bereich verteilt. Die Methode der Verteilung um einen zentralen Punkt ist am besten geeignet, wenn die Position der Strahltaille (Fokuspunkt) des Laserstrahls bekannt ist. Die Methode der Verteilung ausgehend vom Nullpunkt des Linearschlittens ist am besten dafür geeignet die Position der Strahltaille initial zu bestimmen. Die Methode zur Verteilung des Messgitters um einen zentralen Punkt (siehe Abb. 4.7a) benötigt

Hardware Setup	Automated Measurement	Evaluate Measurement	Live Measurement													
Ce	enter Offsetin μm	3000.0	Measurer	ient Area in	μm	20007.0]								O Config O Config	ure around centerpoint ure from zero
Rayl	leigh Length in μm	3000														
Numbe	er of Rayleigh Lengths	4														
Number	of Measurement Points	14														
Meas	surements per Point	5											Cancel			Start
	Current Action		Γ		55				Measurin	g Points		- 25			1.	z - coordinates
Curre	ent Measuring point			*	×e	sert eree	16 ¹⁰	* 37	Ano .	* saint	* Sa	* and	* ⁸⁵¹⁰	and a start	an and	Darth
Cur																
	rrent Measurement			С	urrent Positi	ion		0								
	rrent Measurement Inactive		- 1	С	urrent Positi	ion		0								
-	rrent Measurement		1	С	urrent Positi	ion		0								

Abbildung 4.6: Messgitterkonfiguration

vier Parameter um das Messgitter festzulegen. Die Position des zentralen Punktes ausgehend vom Nullpunkt des Linearschlittens, die Rayleighlänge des zu vermessenden Laserstrahls, der zusammen mit einem einstellbaren Multiplikator den Messbereich in beide Richtungen vom zentralen Punkt festlegt und die Anzahl der Messpunkte. Damit werden die Messpunkte auf den festgelegten Messbereich verteilt. Sollte es nicht möglich sein ein homogenes Messgitter auf den Messbereich zu verteilen, bei dem jeder Messpunkt genau angefahren werden kann (limitiert durch die Schrittweite des Motors), wird die Anzahl der eingestellten Messpunkte automatisch angepasst, sodass jeder Messpunkt angefahren werden kann.

Bei der zweiten Konfigurationsmethode für die Verteilung der Messpunkte wird das Messgitter ausgehend vom Nullpunkt des Linearschlittens (siehe Abb.4.7b) auf eine einstellbare Länge verteilt. Benötigt werden dazu nur zwei Parameter. Die Länge des Messbereichs und die Anzahl der Messpunkte. Das Messgitter wird gleich zur Methode um einen zentralen Punkt auf den Messbereich aufgeteilt und die Anzahl der Messpunkt angepasst, sollte es notwendig sein.





(b) Konfiguration ausgehend vom Nullpunkt des Linearschlittens

4.2.2 Messzyklus

Nach dem Erstellen des Messgitters beginnt ein Messzyklus. Zuerst wird dafür ein neues Objekt Measurement erstellt in dem die Messdaten abgespeichert werden sollen. Der Messzyklus besteht aus einer Hintergrundmessung und einer Hauptmessung. Für die Hintergrundmessung muss der Laserstrahl abgeblockt werden, damit rein der Hintergrund gemessen werden kann. Ist das sichergestellt läuft die Hintergrundmessung wie in Abb.4.10 dargestellt ab. Zu Beginn wird ein neues Objekt Background erstellt in dem die Hintergrunddaten abgespeichert werden. Danach werden die Messpunkte nacheinander angefahren, wobei für jeden Messpunkt ein Objekt SingleMeasurementPoint erstellt wird in dem die Messdaten eines Messpunktes abgespeichert werden. Anschließend wird für jeden Messpunkt die zuvor eingestellte Anzahl der Messungen durchgeführt. Für jede Einzelmessung wird die Kamera über einen Softwaretrigger ausgelöst, das Bild ausgelesen, und mit dem Bild ein Objekt Single-Measurement erstellt. Die SingleMeasurement Objekte werden immer dem jeweiligen SingleMeasurement-Point hinzugefügt und der SingleMeasurementPoint dem Background. Das Endergebnis ist eine wie im Klassenaufbau beschriebene hierarchische Struktur. Ist die Hintergrundmessung abgeschlossen kann der Laserstrahl wieder freigegeben werden und die Hauptmessung gestartet. Der Ablauf der Hauptmessung siehe Abb.4.9 ist identisch zur Hintergrundmessung, nur wird anstelle eines Objektes *Background* ein Objekt MeasurementData erstellt in dem die Messdaten abgelegt werden.



Abbildung 4.8: Ablaufplan Automatisierte Messung



Abbildung 4.9: Ablaufplan Hauptmessung

Abbildung 4.10: Ablaufplan Hintergrundmessung

4.2.3 Auswertung

Mit dem Messprogramm aufgenommenen Messdaten werden nach einem abgeschlossenen Messzyklus direkt an den Auswertungstab in der GUI übergeben und angezeigt (siehe Abb.4.11). Bevor die

Hardware Setup Automated Measurement Eva	luate Measurement Live Measurement		
Configure Evaluation Evaluation Result			
Load Measurement			Measuring Point No. Image No.
Coarse Background Correction Method	Fine Background Correction Method	Fine Corr Value	Measuring Point 1 / z-Pos: 0.0010 mm 🔻 1 🐨 Guess Beamarea
Background Map Substraction	Statistical Method	0	
Convergence Criterion in μm	Wavelength λ in nm		
0.1	300		
nT standard deviation multiplicator	Kernelsize in %		
2	2		
Integrationrange multiplicator			
1			
Image Resolution W x H in pixel	Pixelsize in μm		
1282 1026	5.3		
Evaluation Norm			
ISO11146-1	Evaluate		

Abbildung 4.11: Konfiguration der Auswertungsparameter

Auswertung gestartet werden kann, müssen zuerst die Auswertungsparameter eingestellt werden. Die notwendigen Parameter können in Hintergrundkorrekturparameter und ISO-11146 Parameter unterteilt werden. Die Hintergrundkorrekturparameter umfassen alle Parameter die für die Hinter-



Abbildung 4.12: Ablaufplan der Auswertung

grundkorrektur notwendig sind (Grobkorrekturmethode, Feinkorrekturmethode, n_T und Kernelgröße) und die ISO 11146 Parameter. Davon all jene, die zur Bestimmung der Laserstrahlparameter

notwendig sind (Wellenlänge, Integrationsbereichsmultiplikator, Konvergenzkriterium und Schätzung des Integrationsbereichs).

Der Ablauf der Auswertung ist in Abb.4.12 dargestellt. Zuerst werden die Auswertungsparameter aus der GUI ausgelesen. Anschließend erfolgt die Hintergrundkorrektur (siehe Abb. 4.15) eines Messdatensatzes in der Form eines *MeasurementData* Objekts. Zu Beginn wird ein neues *MeasurementData* Objekt erstellt, indem die korrigierten Messdaten abgespeichert werden. Danach wird in einer Schleife über die Messpunkte iteriert, wobei für jeden *SingleMeasurementPoint* ein neues Objekt vom selben Typ erstellt wird. In einer zweiten Schleife werden alle SingleMeasurement Objekte des *SingleMeasurementPoint* hintergrundkorrigiert und dem neuen *SingleMeasurementPoint* hinzugefügt. Die korrigierten *SingleMeasurementPoint* Objekte werden dann dem *MeasurementData* Objekt hinzugefügt. Der Einfluss und Ablauf der Hintergrundkorrektur eines *SingleMeasurement* Objekts (Einzelbild) wurde in Kap. 2.1 beschrieben. Sind die Messdaten hintergrundkorrigiert können die Parameter wie in Kap. 2.5 beschrieben bestimmt werden.

In Abb. 4.14 ist der Ablauf der Parameterbestimmung dargestellt. Sehr ähnlich zur Hintergrundkorrektur wird über die *SingleMeasurementPoint* Objekte und die SingleMeasurement Objekte iteriert. Die Parameter werden zuerst in der innersten Schleife für die *SingleMeasurement* Objekte, in der äußeren Schleife für die *SingleMeasurementPoint* Objekte und am Ende für das *MeasurementData* Objekt bestimmt. Die ausgewerteten Daten werden dann in der GUI angezeigt (siehe Abb. 4.13). Gegliedert ist die Anzeige der ausgewerteten Messdaten in drei Bereiche. Einem in dem die Parameter für die gesamte Messung angezeigt werden (links), einem für die gemittelten Messdaten eines Messpunktes (Mitte) und einem in dem die einzelnen Messungen und Parameter angezeigt werden (rechts). Im rechten Bereich wird zusätzlich zu den ausgewerteten Parametern immer auch das Bild des Strahlprofil angezeigt. Das Farbschema und die Limits für die Farbskalierung können darunter angepasst werden. Im linken Bereich befinden sich zusätzlich zu den Parametern die Plotoptionen der Messdaten.

|--|

5

			•							•		250		200	150	001	UU+	2	50	}	•		255
			0.7581	1.218	2.777e+04	2.823e+04	-458.7	669.3	669.3	31.67													Max Value
		-	L	L												6)						lue 0
		nage No.	$\langle x \rangle$ in μm	$\langle y \rangle$ in μm	x^2 in μm^2	y^2 in μm^2	xy in μm^2	iameter d_x in μm	iameter d_y in μm	utangle φ in $^{\circ}$												ſ	- Map Min Va
		Ir			\sim	~		Beamdi	Beamdi	Azim													Switch Color
		0010 •																					
		g Point 1 / z-Pos: 0.	1.469	0.5988	1.148	1.478	2.774e+04	86.19	2.823e+04	79.01	-443.2	7.981	669.2	0.9811	669.2	0.9811	30.53	0.512					
		Measuring		L																			
		ng Point No.	_{nean} in µm	std in µm	_{nean} in µm	std in µm	nean in μm^2	$\lambda_{\rm std}$ in μm^2	nean in μm^2	$\rangle_{ m std}$ in μm^2	mean in μm^2	$_{\rm std}$ in μm^2	eter $d_{x \text{ mean}}$ in μm	neter $d_{x \text{ std}}$ in μm	eter $d_{y \text{ mean}}$ in μm	neter $d_{y \text{ std}}$ in μm	ngle $\phi_{ m mean}$ in $^\circ$	angle $\varphi_{\rm std}$ in °					
t		Measuri	$\langle x \rangle$	$\langle x \rangle$	$\langle y \rangle$	$\langle y \rangle$	$\langle x^2 \rangle$	$\langle x^2 \rangle$	$\langle y^2 \rangle$	$\langle y^2 \rangle$	$\langle \Lambda x \rangle$	(x)	Beamdiam	Beamdiar	Beamdiam	Beamdiar	Azimuta	Azimut					
Live Measuremen				•] [] [] [] [] [] [•			
Measurement				SO11146-1	BMS/MV	0.1	300	e	73	2	-10171.4	369644.2	668.6	0.001809	3.165994	-10171.4	369644.2	668.6	0.001809			Plot Selection	
ent Evaluate I		leasurement	Result No.1																				
omated Measurem	Evaluation Result	Load N	No.	n Norm	rection Method	riterion in µm	h A in nm	ize Multiplicator		ze in %	unt	шп	шп	rad	C1 ×	шп	шт	шп	rad	~~~	Data	MP	e
are Setup Aut	lure Evaluation		Result	Evaluatio	Background Cor	Convergence C	Wavelengt	integration Area S	ľu	Kernelsi	z _{0x} in	z _{Rx} in	d_{0x} in	Θ _x in	M	z _{0y} in	z _{Ry} in	d _{0y} in	Θ _y in	Μ	Hold	ot Radius X of al	Sar
lardware Set	Configure Eve				Backg	Con		Integrat														Plot Radi	

Abbildung 4.13: Darstellung der Ergebnisse mit Farbschema jet

4 Messprogramm





Abbildung 4.14: Ablaufplan

Parameterberechnung

4.3 Livemessung

Der Grundaufbau und die notwendigen Parameter der Livemessung sind identisch zur Auswertung (siehe Abb. 4.17). Im Gegensatz zur Auswertung steht hier aber noch kein Messdatensatz und damit auch keine Hintergrunddaten zur Verfügung. Daher ist es notwendig vor der Livemessung selbst Hintergrunddaten am gewünschten Messpunkt aufzunehmen. Die Anzahl der Bilder die vom Hintergrund gemacht werden ist frei wählbar, aber es sollten nie weniger als 15 Bilder eingestellt werden. Erst wenn Hintergrunddaten vorhanden sind, der initiale Integrationsbereich geschätzt und die Auswertungsparameter gesetzt wurden, kann die Livemessung gestartet werden. Das Ergebnis eines Mess- und Auswertungszyklus wird dann im rechten unteren Bereich (siehe Abb. 4.17) angezeigt. Angezeigt werden sowohl das Strahlprofil, als auch die Durchmesser in den Hauptachsenrichtungen und der dazugehörige Azimutwinkel. Das Ergebnis wird periodisch aktualisiert, wobei ein neues Bild erst aufgenommen wird sobald das aktuelle Bild fertig ausgewertet und der *live_toggle* "1" gesetzt ist.



Abbildung 4.16: Ablaufplan Livemessung



Abbildung 4.17: Livemessung

Zusammen mit Messaufbau 3 und dem erstellten Messprogramm wurden zwei Messungen am FSL zu verschiedenen Zeitpunkten durchgeführt. Zum Zeitpunkt der ersten Messung war der im FSL verbaute Pumplaser schon weit degradiert und nur durch sehr hohe Stromzufuhr betreibbar, weshalb der Pumplaser zeitnah nach der Messung ausgetauscht wurde.

Parameter	Vor Tausch	Nach Tausch						
Temperatur	20.7 °C	20.7 °C						
Luftfeuchtigkeit	47.0 %	47.4%						
Anzahl der Messpunkte	21	21						
Anzahl der Messungen pro Messpunkt	20	20						
Messbereich	$23\mathrm{mm}$	20 mm						
Belichtungszeit	$2\mathrm{ms}$	$2\mathrm{ms}$						

Tabelle 5.1: Messparameter

Tabelle 5.2: Zusammenfassung der Messergebnisse

	Vor Tausch	Nach Tausch
Hintergrundkorrektur grob	Backgroundmap-	Backgroundmap-
	substraction	substraction
Hintergrundkorrektur fein	Manual Value: 2	Manual Value: 2.2
Konvergenzkriterium in μm	0.1	0.1
Wellenlänge	800 nm	800 nm
Integrationsbereichsmultiplikator	3	3
Rayleighlänge z_{Rx}	9.500 mm	14.80 mm
Ort der Strahltaillie z_{0x}	11.217 mm	6.46 mm
Strahldurchmesser d_{0x}	$169.2\mu\mathrm{m}$	$215.2\mu\mathrm{m}$
Divergenzwinkel Θ_x (Vollwinkel)	17.8 mrad	14.5 mrad
Beugungsmaßzahl M_x^2	2.96	3.07
Rayleighlänge z_{Ry}	5.523 mm	4.220 mm
Ort der Strahltaillie z_{0y}	9.724 mm	6.44 mm
Strahldurchmesser d_{0y}	$106.1\mu\mathrm{m}$	$67.6\mu\mathrm{m}$
Divergenzwinkel Θ_y (Vollwinkel)	19.2 mrad	16.0 mrad
Beugungsmaßzahl M_y^2	2	1.06
Azimuthwinkel ϕ	0.67 deg	$-1.09 \deg$
Beugungsmaßzahl M_{eff}^2	2.43	1.80

Die zweite Messung wurde durchgeführt nachdem der Pumplaser getauscht und alle optischen Elemente des FSL neu justiert waren. Für beide Messungen wurden 21 Messpunkte mit jeweils

20 Bildern pro Messpunkt aufgenommen. Die Messpunkte wurden auf den maximal zugänglichen Messbereich des Linearschlitten verteilt. Im Vergleich zu vor dem Pumplasertausch ist der zugängliche Messbereich kleiner geworden, da zwischen den Messungen ein Endschalter montiert wurde. Die Belichtungszeit der Kamera wurde auf 2 ms eingestellt, weil der Laser mit einer Wiederholrate von 1 kHz betrieben wird und damit sichergestellt wurde, dass pro Aufnahme zumindest ein Laserpuls detektiert werden konnte.

5.1 Messergebnisse vor dem Pumplasertausch

Abb. 5.1 und Abb. 5.2 zeigen den Verlauf der Strahlradien in Ausbreitungsrichtung des Laserstrahls. Auf der x-Achse wurde die z-Koordinate des Messpunkts ausgehend vom Nullpunkt des Linearschlittens entgegen der Ausbreitungsrichtung des Laserstrahls und auf der y-Achse der Strahlradius $R_{x,y}$ aufgetragen. Beide Verläufe waren erwartungsgemäß hyperbolisch, wobei der Ort des kleinsten Strahlradius sich in x- und y-Richtung voneinander unterschied. Der Strahlradius R_x hatte sein Minimum von 169.2 μ m bei 11.217 mm und in y-Richtung 106.1 μ m bei 9.724 mm.



Abbildung 5.1: Laserstahlradius R_x in Ausbreitungsrichtung vor dem Tausch



Abbildung 5.2: Laserstahlradius R_y in Ausbreitungsrichtung vor Tausch

Abb .5.3 und Abb. 5.4 zeigen das Strahlprofil des nächstliegenden Messpunktes zur Strahltaille in Z-Richtung. Klar erkennbar ist, dass das Strahlprofil innerhalb eines elliptischen Bereichs eine konstante Intensität ähnlich einem "Top Hat" Strahlprofil hatte. Der FSL hat aber eigentlich ein Gaußverteiltes Strahlprofil. Der Unterschied zwischen dem gemessenen Strahlprofil und dem erwarteten Strahlprofil kommt dadurch zustande, dass der Laserstrahl nicht ausreichend abgeschwächt wurde, wodurch es zur Sättigung der Kamerapixel kam.



3D Intensity Distribution (z = 11.501 mm)

Abbildung 5.3: 3D Intensitätsverteilung vor dem Tausch



Abbildung 5.4: Nahaufnahme des Laserstrahlprofils vor dem Tausch

Durch die Übersättigung der Kamerapixel kam es zu einer lokalen Überbelichtung, dem sogenannten "Blooming" Effekt. Zustande kommt dieser Effekt, wenn ein Pixel mehr Ladung in Form von Photonen ausgesetzt ist als er aufnehmen kann. Das führt in weiterer Folge dazu, dass der Pixel die überschüssige Ladung an umliegende Pixel abgibt, wodurch überbelichtete Bereiche im aufgenommenen Bild entstehen. Das gemessene Strahlprofil wird dadurch künstlich vergrößert.

In Abb. 5.5 und Abb. 5.6 wird die Intensitätsverteilungen entlang der Hauptachsen des Strahlprofils dargestellt. Auf der x-Achse wurde der Strahlradius $R_{x,y}$ und in y-Achse die Intensität als Wert im Bereich von 0 – 255 aufgetragen. Zusätzlich wurden der FHWM Durchmesser (magenta strichlierte Linien), der $D4\sigma$ Durchmesser (schwarz strichlierte Linien) und eine an die Intensitätsverteilung angepasste ideale Gaußkurve (rot strichlierte Linie) eingezeichnet.



Abbildung 5.5: 2D Intensitätsverteilung in x-Richtung vor dem Tausch



Abbildung 5.6: 2D Intensitätsverteilung im Fokuspunkt in y-Richtung vor dem Tausch

Die 2D FFT vom Strahlprofil (siehe Abb. 5.7) zeigt die logarithmisch skalierten Amplituden der Ortsfrequenzen. Diese Darstellung hat den Vorteil, dass nur relevante Amplituden im Spektrum sichtbar werden. Es zeigt den erwarteten niederfrequenten Anteil im Zentrum umgeben von symmetrischen Ringen deren Amplitude mit steigender Frequenz sinkt. Diese Ringe sind ein Produkt der überbelichteten Pixel und der FFT. Die FFT zerlegt das Intensitätsprofil in seine Frequenzanteile. Ist das zu analysierende Signal ein Rechtecksignal, was recht gut dem gemessenen Intensitätsprofil entspricht, siehe Abb. 5.5 oder Abb. 5.6, ergeben sich zusätzliche Frequenzanteile die notwendig sind um das Signal nachzubauen. Die Ortsfrequenzanteile in den Hauptachsenrichtungen sind in Abb. 5.8 dargestellt. Die Interpretation ist dabei identisch zur 2D FFT. Es gibt niederfrequente Anteile und die durch die Pixelsättigung bedingten zusätzlichen Frequenzanteile.



Abbildung 5.7: 2D FFT des Laserstrahlprofils vor dem Tausch



Abbildung 5.8: 1D FFT des Laserstrahlprofils vor dem Tausch in Hauptachsenrichtungen

5.2 Messergebnisse nach dem Pumplasertausch

Um die Pixelsättigung (siehe S. 51) zu verhindern wurde ein zusätzliches Filterelement zur Abschwächung des Laserstrahls verbaut. Nach dem Tausch des Pumplasers sind beide Verläufe ebenfalls hyperbolisch, wobei der Ort des kleinsten Strahlradius in x- und y-Richtung fast identisch war.



Abbildung 5.9: Laserstahlradius in x-Richtung nach dem Tausch

Der Strahlradius R_x hatte sein Minimum von 215.2 $\mu{\rm m}$ bei 6.46 mm und in y-Richtung 67.6 $\mu{\rm m}$ bei 6.44 mm.



Abbildung 5.10: Laserstahlradius in y-Richtung nach dem Tausch

Die Strahlprofile (siehe Abb. 5.11 und Abb. 5.12) sind weiterhin elliptisch, aber nun hat das Strahlprofil im Zentrum ein klares Maximum, dass nach außen hin abnimmt.



Abbildung 5.11: 3D Intensitätsverteilung nach dem Tausch

Mögliche Gründe für das elliptische Strahlprofil:

- Das Zentrum der höchsten Intensität des Strahlprofils ist elliptisch und wurde erst durch Abschwächung sichtbar.
- Da der Laserstrahl des FSL aus einem breiten Spektrum an Wellenlängen besteht, die verwendeten ND-Filter allerdings keine konstante Transmission über diesen Bereich besitzen, könnte durch die inhomogene Abschwächung das Strahlprofil beeinflusst worden sein.
- Die Messebene der Kamera könnte nicht orthogonal auf die Ausbreitungsrichtung des Laserstrahls stehen, wodurch das Strahlprofil eine elliptische Form bekommt.



Abbildung 5.12: Nahaufnahme des Laserstrahlprofils nach dem Tausch

Das unfokussierte Strahlprofil wurde auf eine weiße Oberfläche projiziert. Dabei konnte mit freiem Auge festgestellt werden, dass der Rohstrahl ein gaußsches Profil besitzt. Das würde dafür sprechen, dass durch die signifikante Abschwächung des Laserstrahls nurmehr der Bereich mit der höchsten Intensität messbar bleibt. Ob und welchen Einfluss die inhomogenen Transmissionseigenschaften



Abbildung 5.13: 2D Intensitätsverteilung nach dem Tausch

haben, ließe sich nur durch Kenntnis der im inhomogenen räumlichen Verteilung der Bandbreite im Strahlquerschnitt feststellen. Die Orthogonalität der Messebene der Kamera auf die Ausbreitungsrichtung des Laserstrahls wurde während dem Einrichtprozess des Messsystems geprüft. Bevor aber die Winkelverschiebung der Messebene überprüft werden kann muss zuerst überprüft werden, ob der Laserstrahl koaxial zur Verfahrachse des Schlittens ist. Geprüft wurde es, indem der Laserstrahl zuerst auf das Zentrum des Messensors ausgerichtet wurde und anschließend die Kamera über den Linearschlitten in Ausbreitungsrichtung verfahren wurde. Das gemessene Strahlprofil und dessen Position auf der Kamera wurde währenddessen beobachtet, um zu überprüfen ob der Spot über den Sensor wandert. Stellt man bei verfahren des Linearschlittens fest, dass das gemessene Strahlprofil nach oben oder unten wandert, deutet das darauf hin, dass eine Winkelabweichung zur Horizontalen vorliegt. Wandert er nach links oder rechts deutet das darauf hin, dass der Linearschlitten eine Win-

kelverschiebung zur Ausbreitungsrichtung des Laserstrahl aufweist. Wenn man diese beiden Fälle ausgeschlossen hat, kann mithilfe des verstellbaren Spiegels der Laserstrahl auf das Zentrum der Kamera ausgerichtet werden. Die 2D Intensitätsverteilungen in Hauptachsenrichtungen (siehe Abb.



Abbildung 5.14: 2D Intensitätsverteilung nach dem Tausch

5.13 und Abb. 5.14) decken sich gut mit einer idealen Gaußverteilung. Es gibt ein klares Maximum und die Randbereiche der Verteilung sind nicht abgeschnitten (erkennbar durch den kontinuierlichen Verlauf gegen 0), wodurch Überkompensation durch die Hintergrundkorrektur ausgeschlossen werden kann. In der 2D FFT (siehe Abb. 5.15) sind hauptsächlich niederfrequente Anteile vorhanden.



Abbildung 5.15: 2D FFT des Laserstrahlprofils nach dem Wechsel

Die vor dem Tausch des Pumplasers beobachteten periodischen Ringe, aufgrund der Pixelsättigung um den niederfrequenten zentralen Teil des Spektrums, sind hier nicht mehr vorhanden. Die FFT der 2D Intensitätsverteilungen (siehe Abb. 5.16)zeigt in x-Richtung ein schmaleres Frequenzband als in y-Richtung.



Abbildung 5.16: 1D FFT des Laserstrahlprofils nach dem Tausch in Hauptachsenrichtungen

5.3 Vergleich theoretische und reale Fokusdurchmesser

Jede optische Komponente im Aufbau eines Lasers sind aufgrund von Imperfektionen fehlerbehaftet. Es lässt sich daraus schließen, dass auch die theoretisch erreichbaren Werte von den realen abweichen. Das gilt auch für den Fokusdurchmesser der durch ein fokussierendes optisches Element erreicht werden kann. Der Durchmesser eines fokussierten Strahls [2] kann mit der Brennweite f, der Wellenlänge λ , der Strahlgüte $k = 1/M^2$ und dem Strahltaillenradius w_L am fokussierenden Element über

$$d_f = \frac{2 \cdot f \cdot \lambda}{\pi \cdot k \cdot w_L} \tag{5.1}$$

berechnet werden. Verglichen wurden die theoretischen und gemessenen Werte bei einer Brennweite von 100 mm und 500 mm. Für die 100 mm Brennweite wurden die Strahlabmessungen des fokussierten Strahls visuell anhand eines Einschusses mikroskopisch bestimmt (siehe Abb. 5.17). Für die Brennweite von 500 mm konnten die zuvor über das Messsystem bestimmten Strahlabmes-



Abbildung 5.17: Beispiel eines Einschusslochs bei einer Brennweite von 100 mm

sungen herangezogen werden. In Tab. 5.3 sind die Ergebnisse zusammengefasst. Bestimmt wurden die theoretischen Strahldurchmesser für die aus den Messungen bestimmten effektiven Beugungsmaßzahlen und der vom Hersteller angegebenen Beugungsmaßzahl. Es fällt direkt auf, dass sowohl die vermessenen Brandlöcher als auch die mit dem Messsystem gemessenen Strahlprofile elliptisch

sind. Die Differenz der gemessenen Strahlabmessungen zu den theoretischen Abmessungen beträgt vor dem Tausch bei einer Brennweite von 100 mm in x-Richtung 16.8 µm und nach dem Tausch $30.93 \,\mu$ m. In y-Richtung beträgt die Differenz vor dem Tausch $4.9 \,\mu$ m bzw. nach dem Tausch $2.6 \,\mu$ m. Bei der 500 mm Brennweite beträgt die Differenz vor dem Tausch in x-Richtung 76.7 µm und nach dem Tausch 154.1 µm. In y-Richtung beträgt die Differenz vor dem Tausch 13.6 µm und nach dem Tausch 6.6 µm.

Parameter	Hersteller	Vor Tausch	Nach Tausch
Beugungsmaßzahl M^2	2	2.43	1.80
Strahlradius $1/e^2 w_L$	$7.5\mathrm{mm}$	$7.5\mathrm{mm}$	$7.5\mathrm{mm}$
theoretischer Fokusdurchmesser $1/e^2$ d	$13.6\mu\mathrm{m}$	$16.5\mu{ m m}$	12.2 µm
@100mm			
gemessener Fokusdurchmesser $1/e^2 d_x$	-	$33.3\mu\mathrm{m}$	43.13 μm
@100mm			
Differenz Δd_x @100mm	-	16.8 µm	30.93 µm
gemessener Fokusdurchmesser $1/e^2 d_y$	-	$21.3\mu\mathrm{m}$	14.8 µm
@100mm			
Differenz Δd_y @100mm	-	4.9 μm	2.6 µm
theoretischer Fokusdurchmesser $1/e^2$ d	$67.9\mu{ m m}$	$92.5\mu\mathrm{m}$	61.1 μm
@500mm			
gemessener Fokusdurchmesser $1/e^2 d_x$	-	169.2 μm	215.2 μm
@500mm			
Differenz Δd_x @500mm	-	$76.7\mu{ m m}$	$154.4\mu\mathrm{m}$
gemessener Fokusdurchmesser $1/e^2 d_y$	-	$106.1\mu{ m m}$	67.6 μm
@500mm			
Differenz Δd_y @500mm	-	13.6 µm	6.6 µm
Verhältnis $d_x@500mm/d_x@100mm$	-	5.1	4.99
Verhältnis d_y @500mm/ d_y @100mm	-	4.98	4.35

Tabelle 5.3: Vergleich theoretische und reale Fokusdurchmesser

Die Brennweiten der Messungen stehen in einem Verhältnis von 5 zueinander. Bildet man die Verhältnisse der gemessenen Durchmesser bei 500 mm und 100 mm ergibt sich vor dem Tausch des Pumplasers in x-Richtung ein Verhältnis von 4.98 und in y-Richtung 4.35. Nach dem Pumplasertausch ist das Verhältnis in x-Richtung 4.98 und in y-Richtung 4.35. Das spricht dafür, dass die mit dem Messsystem gemessenen Strahlprofile und damit einhergehend die berechneten Strahleigenschaften den tatsächlichen Eigenschaften des Laserstrahls entsprechen.

6 Zusammenfassung und Ausblick

Der Messaufbau und das Messprogramm die im Zuge dieser Arbeit gebaut und programmiert wurden, sind zwar für den Einsatz am Ti:Sa-Femtosekundenlaser entwickelt, können bei Bedarf aber auch für alternative Laserquellen genutzt werden, solange sie ein stigmatisches oder leicht astigmatisches Strahlprofil besitzen. Für die Messung von alternativen Lasern ist zu beachten, dass die verwendete Hardware an den Leistungs- und Frequenzbereich des zu vermessenden Lasers angepasst werden muss. Ansonsten erlaubt der Aufbau des Messsystems auf einer vom Laser unabhängigen Plattform eine einfache Portabilität für die Montage bei anderen Laserquellen. Für astigmatische Laserstrahlen wäre die ISO 11146-2 anzuwenden. Die Bestimmung der dafür notwendigen Parameter würde zusätzlich zur bestehenden Messung, zwei weitere Messungen voraussetzen, für die der Messaufbau und das Messprogramm umgebaut bzw. erweitert werden müssten. Der bestehende Aufbau liefert acht der zehn Momente zweiter Ordnung die für die Bestimmung der Parameter von astigmatischen Strahlen notwendig sind. Um die letzten zwei Momente zweiter Ordnung zu bestimmen, müsste die bestehende Linse nach einem Messzyklus durch eine Zylinderlinse ersetzt und der Messzyklus insgesamt zweimal wiederholt werden (einmal mit vertikal montierter und einmal mit horizontal montierter Zylinderlinse). In weiterer Folge muss auch das Messprogramm um neue Datenelemente und Funktionen erweitert werden, um die zusätzlichen Anforderungen an den Messzyklus und der Auswertung zu erfüllen. Nach der Bestimmung aller Momente zweiter Ordnung mit dem erweiterten Messaufbau und Messprogramm bestimmt, kann die Strahlmatrix gebildet und damit die Strahleigenschaften berechnet werden.

Im Verlauf dieser Arbeit wurden zwei unterschiedliche Ansätze zur Abschwächung der Laserstrahlen für die anschließende Messung der Strahleigenschaften mittels NIR Kamera untersucht. In Messaufbau Version 1 wurde versucht den Laserstrahl durch Reflexion an Prismen abzuschwächen. Im Zuge der ersten Tests wurde festgestellt, dass "Ghosting" Strahlen resultierend aus sekundären Reflexionen in den Prismen zu Interferenzen mit dem Primärstrahl führten. Die Abschwächung des Laserstrahls mittels Reflexion hat einen Messtechnischen Vorteil gegenüber der Abschwächung mittels Transmission durch Filterelemente. Der Vorteil ist, dass die Reflexionseigenschaften von rechtwinkligen Glasprismen oder Glaswedges sehr konstant über einen breiten Wellenlängenbereich sind. Dadurch kann ein Einfluss der frequenzabhängigen Abschwächung auf das Strahlprofil ausgeschlossen werden. Der große Nachteil dieser Methode ist, dass für große Abschwächungen viele Stufen notwendig sind und zwischen den Stufen genug Abstand vorgesehen werden muss, um die "Ghosting" Strahlen aus dem Messsystem auszukoppeln. Verwendet man wie in Messaufbau Version 3 Filter zur Abschwächung, kann das Messsystem wesentlich kompakter gebaut werden, es muss jedoch die Frequenzabhängigkeit der Transmission durch die Filter berücksichtigt werden. Die Abschwächung des Laserstrahls ist vor allem für die Vermessung von Hochleistungslasern notwendig, da der auf die Kamera fokussierte Laserstrahl ansonsten entweder die Kamerapixel zerstören oder sättigen würde. Gesättigte Pixel führen zu einer künstlichen Vergrößerung des gemessenen Strahlprofil durch den "Blooming" Effekt. Es gibt weitere Anforderungen an das Strahlprofil und die Kamera die beachtet werden müssen. Der Durchmesser des Strahlprofils darf nicht größer sein als ein Drittel der verfügbaren Sensorfläche, da in der Norm festgelegt ist, dass der Auswertungsbereich (Integrationsbereich) dreimal so groß sein sollte wie die zu bestimmenden Strahlabmessungen. Weiters sollte der Strahldurchmesser laut Norm mit mindestens 20 Kamerapixel aufgelöst werden. Mit der in dieser Arbeit verwendeten Kamera betrug die Auflösung des Strahldurchmesser bei einer Brennweite von 500 mm 10 Kamerapixel. Um die Auflösung von 20 Kamerapixel zu erreichen

hätte man entweder eine Kamera mit höherer Auflösung verwenden müssen oder die Brennweite des fokussierenden Elements vergrößern. Aufgrund des beschränkten Platzes war die Brennweite allerdings bereits mit 500 mm am Limit des möglichen und eine andere Kamera stand nicht zur Verfügung. Für die Messung von stigmatischen Strahlen sind 20 Messpunkte im Bereich von zumindest zwei Rayleighlängen in beiden Richtungen ausgehend von der Strahltaille erforderlich um ein normkonformes Messergebnis zu erhalten. Daraus ergibt sich, dass der Linearschlitten einen Verfahrweg von mindestens 5 Rayleighlängen und eine passende Ganggenauigkeit haben muss.

Zusammenfassend gibt es viele Feinheiten die bei der Messung und Auswertung der räumlichen Strahleigenschaften eines Lasers beachtet werden müssen. Jede Komponente im Messaufbau kann das Messergebnis beeinflussen und muss daher sorgfältig ausgewählt werden um ein möglichst großes Spektrum an unterschiedlichen Lasern abdecken zu können. Als nächsten Schritt wäre es wünschenswert einen Messaufbau aufzubauen der den Laserstrahl mittels Reflexion abschwächt. Dann könnten die Vor- und Nachteile der beiden Ansätze zur Abschwächung direkt miteinander verglichen werden.

Abbildungsverzeichnis

1.1	Verlauf der elektrischen Feldstärke \mathbf{E} und der magnetischen Feldstärke \mathbf{H} in Aus-	e
1 0	Dreitungsrichtung einer Lichtweile [2]	0
1.2	relative Lightempfindlighteit des menschlichen Auges [2]	07
1.0	Schematische Dergtellung von Absorption, gnontener Emission und induzierter Emis	1
1.4	sion [2]	0
15	Sion [2]	0
1.0	Schandulagramm eines verstarkers mit positiver Ruckkopplung [5]	10
1.0 1.7	TEM Moden mit unterschiedlicher Summetrie	11
1.1	Aufbau zur Beobachtung von individuellen TEM Moden [5]	10
1.0	Aufgenommene Strahlprofile für verschiedene TEM Moden eines He Ne Lasers [5]	14
1.9	Varainfachta Schemachizza eines Ti:Saphir Femtoschunden Lasers [5].	14
1.10	vereimachte Schemaskizze eines 11.5aphil Feintosekunden Lasers	10
2.1	Einfluss der Größe einer Kernelmatrix auf die Bildverschmierung	20
2.2	Einfluss der Graustufenskalierung und der Korrekturverfahren	22
2.3	Auswahl des geschätzten Integrationsbereichs	26
2.4	An die Strahldurchmesser in x-Richtung angepasste hyperbolische Kurve	26
3.1	Anordnung opto-mechanischer Elemente vor dem FSL	28
3.2	Messaufbau Version 1	29
3.3	Messaufbau Version 2 und Strahlfalle	30
3.4	Reflexionen im Prisma	31
3.5	Laserstrahlprofil	31
3.6	Ortsfrequenzspektrum eines Laserstrahlprofils	32
3.7	Messaufbau Version 3	32
3.8	Rechtwinkliges Prisma PS908 [11]	33
3.9	Transmissionskurve von unbeschichtetem BK7 [12]	33
3.10	ND Filter NE40B-B, NE50B-B [14]	34
3.11	ND-Filter Transmissionskurven für OD 2.0 - OD 6.0 [15]	34
3.12	Schrittmotor Sanyo Denki 103H5205-5240 [17]	34
3.13	Momentenverlauf Sanyo Denki 103H5205-5240 [18]	35
3.14	Arduino UNO Rev3 [20]	35
3.15	Arduino Motor Shield Rev3 [22]	36
3.16	Basler ace GigE acA1300-60gmNIR $[24]$	36
11	Klassondiagramm Moasurement	27
4.1	Klassendiagramm Background	37
4.2	Klassendiagramm MeasurementData	38
4.0	Klassendiagramm SingleMeasurementPoint	30
ч.ч 45	Klassendiagramm SingleMeasurement	- <u>7</u> 9 - 70
н.J Д б	Massentragramm singlemeasurement	40 ∕11
4.0 1/7	Messgitter Konfigurationsmethoden	41 /19
т. 1 8	Ablaufolan Automatisierte Messung	±∠ ⊿9
т.0 Д 0	Ablaufplan Hauptmessung	-∓∠ //२
н.9 Д 10	Ablaufplan Hintergrundmessung	40 42
1. 10		- ± 0

Abbildungsverzeichnis

4.11	Konfiguration der Auswertungsparameter	44
4.12	Ablaufplan der Auswertung	44
4.13	Darstellung der Ergebnisse mit Farbschema jet	46
4.14	Ablaufplan Parameterberechnung	47
4.15	Ablaufplan Hintergrundkorrektur	47
4.16	Ablaufplan Livemessung	48
4.17	Livemessung	48
5.1	Laserstahlradius R_x in Ausbreitungsrichtung vor dem Tausch $\ldots \ldots \ldots \ldots \ldots$	50
5.2	Laserstahlradius R_y in Ausbreitungsrichtung vor Tausch $\ldots \ldots \ldots \ldots \ldots \ldots$	50
5.3	3D Intensitätsverteilung vor dem Tausch	51
5.4	Nahaufnahme des Laserstrahlprofils vor dem Tausch	51
5.5	2D Intensitätsverteilung in x-Richtung vor dem Tausch	52
5.6	2D Intensitätsverteilung im Fokuspunkt in y-Richtung vor dem Tausch	52
5.7	2D FFT des Laserstrahlprofils vor dem Tausch	53
5.8	1D FFT des Laserstrahlprofils vor dem Tausch in Hauptachsenrichtungen \ldots	53
5.9	Laserstahlradius in x-Richtung nach dem Tausch	53
5.10	Laserstahlradius in y-Richtung nach dem Tausch	54
5.11	3D Intensitätsverteilung nach dem Tausch	54
5.12	Nahaufnahme des Laserstrahlprofils nach dem Tausch	55
5.13	2D Intensitätsverteilung nach dem Tausch	55
5.14	2D Intensitätsverteilung nach dem Tausch	56
5.15	2D FFT des Laserstrahlprofils nach dem Wechsel	56
5.16	1D FFT des Laserstrahlprofils nach dem Tausch in Hauptachsenrichtungen \ldots .	57
5.17	Beispiel eines Einschusslochs bei einer Brennweite von 100 mm	57
7.1	Startbildschirm des Messprogramms	191

Tabellenverzeichnis

1.1	Datenblatt FEMTOPOWER COMPACT PRO	15
5.1	Messparameter	49
5.2	Zusammenfassung der Messergebnisse	49
5.3	Vergleich theoretische und reale Fokusdurchmesser	58

7 Anhang

Anhang 1: Quellcode

7.1	Class Measurement			65
7.2	Function addResult			66
7.3	Class MeasurementData			66
7.4	Function addSingleMeasurementPoint			69
7.5	Function backgroundCorrectionMeasurementData			69
7.6	Function iso11146Part1MeasurementData			69
7.7	Class Background			71
7.8	Function addSingleMeasurementPoint			72
7.9	Function getMeanFromCorners			72
7.10	Function getMeanStandardDeviation			73
7.11	Function getMeanSubstractionMap			73
7.12	Function getMeanSubstractionMapDouble			73
7.13	Function getMeanValue			73
7.14	Class MeasurementSettings			74
7.15	Class SingleMeasurementPoint			74
7.16	Function addMeasurement			77
7.17	Function backgroundCorrectionSingleMeasurementPoint			77
7.18	Function backgroundCorrectionSingleMeasurementPoint			78
7.19	Function iso11146Part1SingleMeasurementPoint			78
7.20	Function showAllImages			79
7.21	Function showImage			80
7.22	Function showImageWithCircleDetection			80
7.23	Function showImageWithImTool			80
7.24	Class SingleMeasurement			81
7.25	Function backgroundCorrectionSingleMeasurement			82
7.26	Function iso11146Part1SingleMeasurement			86
7.27	Class Result			88
7.28	Function lbcUI_Launcher			90
7.29	Function CalculatePacketDelay			90
7.30	Class lbcUI			93
7.31	$Function AutoMeasISO111146 Part1MethodRBGroup_SelectionChangedFcn \ .$			146
7.32	Function AutoMeasISO111146Part1StartMeasButton_ButtonPushedFcn			146
7.33	Function CalcNumOfSteps		•	149
7.34	Function ClearGUIEvalResTab			149
7.35	Function ConfEvalBackCorrMethodDropDown_ValueChangedFcn			151
7.36	Function ConfEvalConvergeCriterionSlider_AllEvents		•	151
7.37	'Function ConfEvalDrawRectButton_ButtonPushedFcn			151
7.38	Function ConfEvalEvaluationButton_ButtonPushedFcn			152
7.39	Function ConfEvalLoadMeasButton_ButtonPushedFcn			153
7.40	Function ConfEvalSmpDropDown_ValueChangedFcn			154
7.41	Function ConfEvalSmpImDropDown_ValueChangedFcn		•	155

7.42	Function drawConfEvalRectAllevents	155
7.43	Function drawLiveMeasRectAllevents	156
7.44	Function EvalResImChangeColorMap	156
7.45	Function EvalResImDataChangeColorMapButton_ValueChangedFcn	157
7.46	Function EvalResImDataDropDown_ValueChangedFcn	157
7.47	Function EvalResMeasDataResultDropDown_ValueChangedFcn	157
7.48	Function EvalResMeasPlotButton_ButtonPushedFcn	157
7.49	Function EvalResSaveMeasButton_ButtonPushedFcn	174
7.50	Function EvalResSmpDataDropDown_ValueChangedFcn	175
7.51	Function HSCamConfApplySettingButton_ButtonPushedFcn	175
7.52	Function HSCamConfCamPreviewButton_ValueChangedFcn	175
7.53	Function HSCamConfCamTestButton_ButtonPushedFcn	176
7.54	Function HSCamConfConnectButton_ButtonPushedFcn	176
7.55	Function HSCamConfDisconnectButton_ButtonPushedFcn	177
7.56	Function HSStepmotorConfApplySettingsButton_ButtonPushedFcn	177
7.57	Function HSStepmotorConfComPortRefreshButton_ButtonPushedFcn	178
7.58	Function HSStepmotorConfConnectButton_ButtonPushedFcn	178
7.59	Function HSTestHardwareGoToPositionButton_ButtonPushedFcn	178
7.60	Function HSTestHardwareGoToZeroButton_ButtonPushedFcn	179
7.61	Function HSTestHardwareReferenceStageButton_ButtonPushedFcn	179
7.62	Function LiveMeasBackgroundMeasButton_ButtonPushedFcn	179
7.63	Function LiveMeasDrawRectButton_ButtonPushedFcn	180
7.64	Function LiveMeasStartStopButton_ValueChangedFcn	180
7.65	Function LiveMeasTriggerImageButton_ButtonPushedFcn	182
7.66	Function moveToZpos	183
7.67	Function SetEvalResImDataParams	183
7.68	Function SetEvalResMeasDataParams	184
7.69	Function SetEvalResSmpDataParams	186
7.70	Function TriggerCameraImage	186
7.71	Function UpdateGUIConfEvalTab	187
7.72	Function UpdateGUIEvalResTab	188
7.73	Function UpdatePreviewAxes	189

7.1: Class Measurement

classdef M	leasurement < handle
%MEASU	IREMENT Class used to perform, store and analyze a
%Laser	beammeasurement.
proper	ties(Access = public)
me	asurementData MeasurementData = MeasurementData();
ba	ickgroundData Background
me	asSettings MeasurementSettings;
re	esults Result;
end	
method	ls
fu	<pre>unction obj = Measurement(rayleighLength,timesReyleighLength,noMP,</pre>
	medsPerPoint)
	obj.meassettings = measurementsettings(raytergnLength,

 $\begin{array}{c}
 1 \\
 2 \\
 3 \\
 4 \\
 5 \\
 6 \\
 7 \\
 8 \\
 9
 \end{array}$

```
obj.backgroundData = Background();
                obj.measurementData = MeasurementData();
            end
18
            function reset(obj)
                obj.measurementData.reset();
                obj.backgroundData.reset();
21
            end
            function value = getMeasurementData(obj)
                % Get stored measurment data.
                value = obj.measurementData;
            end
26
            function value = getBackgroundData(obj)
27
                % Get stored background data.
28
                value = obj.backgroundData;
29
            end
            function value = getResults(obj)
                % Get stored measurment data.
                value = obj.results;
            end
            addResult(obj,value);
        end
   end
1
    function addResult(obj,value)
2
   %addResult Adds new Result to the existing Measurement result dataset.
3
   obj.results(end+1) = value;
 4
   end
 1
    classdef MeasurementData < handle</pre>
2
        %MEASUREMENTDATA Class containing the measurement data of all measurementpoints
```

```
properties (Access = public)
```

timesReyleighLength,noMP,measPerPoint);

measurementDataSet SingleMeasurementPoint;

W_X_theta_X double; W_Y_theta_Y double; W_theta_X_squared double; W_theta_Y_squared double; W_theta_X_theta_Y double;

fx cfit; z0x double;

4

6 7

8

9

12

14

7.2: Function addResult

7.3: Class MeasurementData

16	dSigma0x double;
17	thetaSigmaX double;
18	zRx double;
19	MsquaredX double;
20	
21	TY CTIT;
44 02	ZUY double;
20 94	thetaSigmaX double:
$\frac{24}{25}$	$z_{\rm Rv}$ double:
$\frac{20}{26}$	MsquaredY double:
20	
28	zPos double;
29	dWx double;
30	dWxStd double;
31	dWy double;
32	dWyStd double;
33	end
34	
35	methods
30 97	<pre>% TUNCTION delete(obj) % delete(obi measurementDeteCet);</pre>
37 90	% detete(obj.measurementDataSet);
20 20	<pre>% end function reset(obj)</pre>
39 40	obj measurementDataSet = SingleMeasurementPoint empty:
41	end
42	function measurementData(obi)
43	<pre>obj.measurementDataSet = SingleMeasurementPoint.empty;</pre>
44	end
45	<pre>function measurementDataSet = getMeasurementDataSet(obj)</pre>
46	% Get stored measurment dataSet.
47	<pre>measurementDataSet = obj.measurementDataSet;</pre>
48	end
49	<pre>function value = getZ0X(obj)</pre>
50	% Get stored measurment dataSet.
51	<pre>value = obj.z0x;</pre>
52	end
53	<pre>function value = getZ0Y(0bj) </pre>
04 55	% Get stored measurment dataSet.
00 56	value = obj.zov;
57	function value = getDSigmaOx(obi)
58	% Get stored measurment dataSet.
59	value = obj.dSigma0x:
60	end
61	<pre>function value = getDSigma0y(obj)</pre>
62	% Get stored measurment dataSet.
63	<pre>value = obj.dSigma0y;</pre>
64	end

```
Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar vour knowledge hub. The approved original version of this thesis is available in print at TU Wien Bibliothek.
                                                                                                  100
                                                                                             106
                                                                                                108
```

78

80

81

82

84

85

86

87

88

89

90

96

```
function value = getThetaSigmaX(obj)
    % Get stored measurment dataSet.
    value = obj.thetaSigmaX;
end
function value = getThetaSigmaY(obj)
    % Get stored measurment dataSet.
    value = obj.thetaSigmaY;
end
function value = getZrX(obj)
    % Get stored measurment dataSet.
    value = obj.zRx;
end
function value = getZrY(obj)
    % Get stored measurment dataSet.
    value = obj.zRy;
end
function value = getMsquaredX(obj)
    % Get stored measurment dataSet.
    value = obj.MsquaredX;
end
function value = getMsquaredY(obj)
    % Get stored measurment dataSet.
    value = obj.MsquaredY;
end
function value = getfxFit(obj)
    % Get stored measurment dataSet.
    value = obj.fx;
end
function value = getfyFit(obj)
    % Get stored measurment dataSet.
    value = obj.fy;
end
function value = getzPos(obj)
    % Get stored measurment dataSet.
    value = obj.zPos;
end
function value = getdWx(obj)
    % Get stored measurment dataSet.
    value = obj.dWx;
end
function value = getdWxStd(obj)
    % Get stored measurment dataSet.
    value = obj.dWxStd;
end
function value = getdWyStd(obj)
    % Get stored measurment dataSet.
    value = obj.dWyStd;
end
function value = getdWy(obj)
```

114		% Get stored measurment dataSet.
115		<pre>value = obj.dWy;</pre>
116		end
117		addSingleMeasurementPoint(obj,value);
118		<pre>value = backgroundCorrectionMeasurementData(obj,coarseMethod,fineMethod,</pre>
		background,kernelSize,ntFactor,integAreaFactor,k,manualValue);
119		<pre>% value = correction(obj,method,backgroundData,varargin);</pre>
120		<pre>value = iso11146Part1MeasurementData(obj,centerGuessX,centerGuessY,</pre>
		integAreaGuessX,integAreaGuessY,convergeKriterium,faktor,lambda,resWidth
		,resHeight,pixelsize);
121		<pre>value = iso11146Part2MeasurementData(obj,centerGuessX,centerGuessY,</pre>
		integAreaGuessX,integAreaGuessY,convergeKriterium,faktor,lambda);
122	end	
123	end	

7.4: Function addSingleMeasurementPoint

```
function addSingleMeasurementPoint(obj,value)
2
  %addSingleMeasurementPoint Add a element of type SingleMeasurementPoint to
3
  %dataSet.
4
  obj.measurementDataSet(end+1) = value;
  end
```

7.5: Function backgroundCorrectionMeasurementData

```
function value = backgroundCorrectionMeasurementData(obj,coarseMethod,fineMethod,
1
       background,kernelSize,ntFactor,manualValue)
   %backgroundCorrectionMeasurementData Initiates backgroundcorrection for the
   %whole measurementdata.
4
5
       Correct measurment data.
   %
6
   if not(isa( background, 'Background' ))
       error('Invalid Classtype of input parameter background.');
8
   end
9
   measurementDataSet = obj.getMeasurementDataSet();
   [Useless,NoMP] = size(measurementDataSet);
   value = MeasurementData();
12
   for k = 1 : NoMP
       fprintf('Performing background correction for measuring point %i.\n',k);
14
       corrMeasPoint = measurementDataSet(k).backgroundCorrectionSingleMeasurementPoint
           (coarseMethod,fineMethod,background,kernelSize,ntFactor,k,manualValue);
       value.addSingleMeasurementPoint(corrMeasPoint);
15
   end
   end
```

7.6: Function iso11146Part1MeasurementData

```
function value = iso11146Part1MeasurementData(obj,centerGuessX,centerGuessY,
      integAreaGuessX,integAreaGuessY,convergeKriterium,faktor,lambda,pixelsize)
2
  %isoll146Part1MeasurementData calc LB parameters using ISOll146-1 for
  %background corrected Measurementdata
```

1

```
4
           measurementDataSet = obj.getMeasurementDataSet();
        6
           [Useless,NoMP] = size(measurementDataSet);
        7
           value = MeasurementData();
           zPos = [];
        8
        9
           dWx = [];
           dWxStd = [];
           dWy = [];
           dWyStd = [];
           for k = 1 : NoMP
       14
                fprintf('Computing IS011146-1 parameters for measuring point %i.\n',k);
                corrMeasPoint = measurementDataSet(k).isol1146Part1SingleMeasurementPoint(
                     centerGuessX, centerGuessY, integAreaGuessX, integAreaGuessY, convergeKriterium,
TU Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
VIEN vur knowledge hub
The approved original version of this thesis is available in print at TU Wien Bibliothek.
                     faktor,pixelsize);
                value.addSingleMeasurementPoint(corrMeasPoint);
       17
       18
                zPos = [zPos,corrMeasPoint.getZPos()];
                dWx = [dWx,corrMeasPoint.getdWx_mean()];
                dWxStd = [dWxStd,corrMeasPoint.getdWx_std];
                dWy = [dWy,corrMeasPoint.getdWy_mean()];
                dWyStd = [dWyStd,corrMeasPoint.getdWy_std];
       23
       24
           end
           offset_weight = 1;
       26
           weight = 0.1*ones(1,size(dWx,2));
       27
       28
            [V,I] = min(dWx);
       29
           for i=1: size(dWx,2)
                if (i >= I-offset_weight && i <= I+offset_weight) || i <= offset_weight || i>=
                     size(dWx,2)—offset_weight
                     weight(i) = 1;
                end
           end
       34
            fx=fit(zPos',dWx.^2','poly2','Robust','Bisquare','Weight',weight);
           value.fx = fx;
       38
           cx = fx.p1;
           bx = fx.p2;
           ax = fx.p3;
       41
       42
           value.z0x = -bx/(2*cx);
       43
           value.dSigma0x = (1/(2*sqrt(cx)))*sqrt(4*ax*cx-bx^2);
       44
           value.thetaSigmaX = sqrt(cx);
           value.zRx = 1/(2*cx)*sqrt(4*ax*cx-bx^2);
           value.MsquaredX = pi/(8*lambda/le+3)*sqrt(4*ax*cx-bx^2);
       47
          value.fx = fx;
       48
```

```
weight = 0.1*ones(1,size(dWy,2));
    [V,I] = min(dWy);
52
    for i=1: size(dWy,2)
        if (i >= I—offset_weight && i <= I+offset_weight) || i <= offset_weight || i>=
           size(dWy,2)—offset_weight
            weight(i) = 1;
        end
   end
    fy = fit(zPos',dWy.^2','poly2','Robust','Bisquare','Weight',weight);
58
59
   value.fy = fy;
61
   cy = fy.p1;
62
   by = fy.p2;
   ay = fy.p3;
64
   value.z0y = -by/(2*cy);
   value.dSigma0y = (1/(2*sqrt(cy)))*sqrt(4*ay*cy—by^2);
   value.thetaSigmaY = sqrt(cy);
68
   value.zRy = 1/(2*cy)*sqrt(4*ay*cy-by^2);
    value.MsquaredY = pi/(8*lambda/1000)*sqrt(4*ay*cy—by^2);
   value.fy = fy;
   value.zPos = zPos;
   value.dWx = dWx;
74
   value.dWxStd = dWxStd;
   value.dWy = dWy;
   value.dWyStd = dWyStd;
78
   end
```

7.7: Class Background

```
classdef Background < handle</pre>
    %BACKGROUNDMEASUREMENTDATA Background data class containing background
    %Measuring points and functions to evaluate background data.
    properties(Access = public)
        backgroundDataSet SingleMeasurementPoint;
    end
    methods
        %
                  function delete(obj)
        %
                      delete(obj.backgroundDataSet);
        %
                  end
        function reset(obj)
            obj.backgroundDataSet = SingleMeasurementPoint.empty;
        end
        function value = getData(obj,measPoint)
```

1

2

3 4

6

7

8 9

12

```
%METHOD1 Summary of this method goes here
18
                    Detailed explanation goes here
                value = obj.backgroundDataSet(measPoint);
20
            end
            addSingleMeasurementPoint(obj,value);
            value = getMeanValue(obj,measPoint);
23
            value = getMeanSubstractionMap(obj,measPoint);
24
            value = getMeanStandardDeviation(obj,measPoint);
25
            value = getMeanFromCorners(obj,measPoint,n,m);
        end
   end
```

7.8: Function addSingleMeasurementPoint

```
function addSingleMeasurementPoint(obj,value)
  %addSingleMeasurementPoint Adds a new Singlemeasurementpoint to the
  %existing dataset
4
  obj.backgroundDataSet(end+1) = value;
  end
```

7.9: Function getMeanFromCorners

```
function value = getMeanFromCorners(obj,measPoint,n,m)
%GETMEANFROMCORNERS Returns the mean pixel value taken from corner areas.
%The averaged area is defined by n and m.
backgroundDataPoint = obj.getData(measPoint);
[Useless,NoMP] = size(backgroundDataPoint);
for k = 1 : NoMP
    backgroundDataPointSet = backgroundDataPoint(k).getDataSet();
    [Useless,NoI] = size(backgroundDataPointSet);
    for f = 1 : NoI
        backgroundDataPointImageData = backgroundDataPointSet(f).getImageData();
        [rows,cols] = size(backgroundDataPointImageData);
        temp = 0;
        counter = 0;
        for i=1:rows
            for j=1:cols
                if (i <= n && j <= m) || (i <= n && j >= cols-m+1) || (i >= rows-n+1
                     && j <= m) || (i >= rows—n+1 && j >= cols—m+1)
                    temp = temp + double(backgroundDataPointImageData(i,j));
                    counter = counter + 1;
                end
            end
        end
    end
end
value = temp/counter;
end
```

2

1

2

4

7

8

9

12

14

17

18

20

23

24

25

26
7.10: Function getMeanStandardDeviation

```
1
  function value = getMeanStandardDeviation(obj,measPoint)
2
  %GETMEANSTANDARDDEVIATION Returns the Standard Deviation of a Measuring
3
  %Point. All Images belonging to Measuring Point are used.
 backgroundDataPoint = obj.getData(measPoint);
4
  backgroundDataPoint.calcMeanImage();
6
 meanBack=backgroundDataPoint.getMeanImage().getImageData();
7
  value = std(double(reshape(meanBack,[],1)));
8
  end
```

7.11: Function getMeanSubstractionMap

```
function value = getMeanSubstractionMap(obj,measPoint);
1
2
   %GETSUBSTRACTIONMAP Returns the mean Backgroundsubstraction map created
3
   %using all Images belonging to the Measuring Point.
4
   backgroundDataPoint = obj.getData(measPoint);
6
   backgroundDataPointSet = backgroundDataPoint.getDataSet();
7
   [Useless,NoI] = size(backgroundDataPointSet);
8
   [rows,cols] = size(backgroundDataPointSet(1));
9
   temp=zeros(rows,cols,'double');
   for k = 1 : NoI
       temp = temp+double(backgroundDataPointSet(k).getImageData());
12
   end
   % value = cast(ceil(temp./NoI), 'uint8');
14 value = cast(temp./NoI,'uint8');
15 end
```

7.12: Function getMeanSubstractionMapDouble

```
function value = getMeanSubstractionMapDouble(obj,measPoint);
1
2
   %GETSUBSTRACTIONMAP Returns the mean Backgroundsubstraction map created
   %using all Images belonging to the Measuring Point.
4
   backgroundDataPoint = obj.getData(measPoint);
6
   backgroundDataPointSet = backgroundDataPoint.getDataSet();
7
   [Useless,NoI] = size(backgroundDataPointSet);
8
   [rows,cols] = size(backgroundDataPointSet(1));
9
   temp=zeros(rows,cols,'double');
   for k = 1 : NoI
       temp = temp+double(backgroundDataPointSet(k).getImageData());
12
   end
   % value = cast(ceil(temp./NoI), 'uint8');
14
   value = temp./NoI;
   end
```

7.13: Function getMeanValue

```
function value = getMeanValue(obj,measPoint)
 %GETMEANVALUE Returns the mean pixel value of all images belonging to the
3 %Measuring point.
```

```
4
   backgroundDataPoint = obj.getData(measPoint);
6
   backgroundDataPointSet = backgroundDataPoint.getDataSet();
 7
   [Useless,NoI] = size(backgroundDataPointSet);
   [rows,cols] = size(backgroundDataPointSet(end).getImageData());
8
9
   temp=0;
   for k = 1 : NoI
        imageData=backgroundDataPointSet(k).getImageData();
12
        temp = temp + mean( double(imageData) , 'all' );
   end
14
   value = temp/NoI;
   end
```

7.14: Class MeasurementSettings

```
classdef MeasurementSettings
       %MEASUREMENTSETTINGS data class containing information about the
       %measurement settings
       properties
           measurementMethod string;
           rayleighLength double;
           timesReyleighLength double;
9
           noMP double;
           measPerPoint double;
           measArea double;
       end
14
       methods
           function obj = MeasurementSettings(rayleighLength,timesReyleighLength,noMP,
               measPerPoint)
               %MEASUREMENTSETTINGS Construct an instance of this class
                    Detailed explanation goes here
               %
               obj.rayleighLength = rayleighLength;
               obj.timesReyleighLength = timesReyleighLength;
               obj.noMP = noMP;
               obj.measPerPoint = measPerPoint;
           end
       end
   end
```

7.15: Class SingleMeasurementPoint

```
classdef SingleMeasurementPoint < handle</pre>
    SINGLEMEASUREMENTPOINT Class containing the data of a single measurementpoint.
    properties (Access = public)
        dataSet SingleMeasurement;
        zPos double;
        meanImage SingleMeasurement;
```

1

2

4

6

7

8

12

17

18

23

1

2

4

5

8			
9			dWx_mean double;
10			dWy_mean double;
11	azimutPhi_mean double;		
12	beamCenterX_mean double;		
13	beamCenterY_mean double		
14			W_X_mean double;
15			W_Y_mean double;
16			W_XY_mean double;
17			W_X_squared_mean double;
18			W_Y_squared_mean double;
19			
20			dWX_std double;
$\frac{21}{22}$			dwy_std double;
$\frac{22}{23}$			beamCenterX std double:
20			beamCenterY_std double:
25			W X std double:
26			W_Y_std double:
27			W_XY_std double;
28			W_X_squared_std double;
29			W_Y_squared_std double;
30			
31		end	
32			
33		meth	hods
34	%	funct	lon delete(obj)
35 26	<i>б</i>		delete(obj.dataSet);
30 37	6		enu function obi - SingleMeasurementPoint(value)
28			SINCLEMEASUREMENTPOINT Construct an instance of this class
39			% Detailed explanation goes here
40			obi.zPos = value:
41			end
42			<pre>function value = getDataSet(obj)</pre>
43			<pre>% Get stored measurementpoint dataSet.</pre>
44			<pre>value = obj.dataSet;</pre>
45			end
46			<pre>function value = getZPos(obj)</pre>
47			% Get xCoordinate of measurementpoint.
48			value = obj.zPos;
49			end
50 E 1			addMeasurement(obj,value);
91 59			$f_{\text{unction value}} = \text{detMeanImage(obi)}$
53			value = obi meanImage(Obj)
54			end
55			<pre>function value = getdWx_mean(obj)</pre>
56			% Get xCoordinate of measurementpoint.
	I		·

```
Bibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar vour knowedge hub The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

72

78

80

81

82

83 84

85

86

87

88

89

90

96

```
value = obj.dWx_mean;
 end
 function value = getdWy_mean(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.dWy_mean;
 end
 function value = getW_X_mean(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.W_X_mean;
 end
 function value = getW_Y_mean(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.W_Y_mean;
 end
 function value = getW_XY_mean(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.W_XY_mean;
 end
 function value = getW_X_squared_mean(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.W_X_squared_mean;
 end
function value = getW_Y_squared_mean(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.W_Y_squared_mean;
end
         function value = getdWx_std(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.dWx_std;
 end
 function value = getdWy_std(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.dWy_std;
 end
 function value = getW_X_std(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.W_X_std;
 end
 function value = getW_Y_std(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.W_Y_std;
 end
 function value = getW_XY_std(obj)
    % Get xCoordinate of measurementpoint.
    value = obj.W_XY_std;
 end
 function value = getW_X_squared_std(obj)
    % Get xCoordinate of measurementpoint.
```

```
106
                 value = obj.W_X_squared_std;
              end
             function value = getW_Y_squared_std(obj)
                 % Get xCoordinate of measurementpoint.
                 value = obj.W_Y_squared_std;
             end
             function value = getAzimutPhi_mean(obj)
                 % Get xCoordinate of measurementpoint.
                 value = obj.azimutPhi_mean;
             end
             function value = getAzimutPhi_std(obj)
118
                 % Get xCoordinate of measurementpoint.
                 value = obj.azimutPhi_std;
             end
             function value = getBeamCenterX_mean(obj)
                 % Get xCoordinate of measurementpoint.
                 value = obj.beamCenterX_mean;
             end
              function value = getBeamCenterY_mean(obj)
                 % Get xCoordinate of measurementpoint.
                 value = obj.beamCenterY_mean;
             end
             showImage(obj,imageIndex);
             [centers,radii] = showImageWithCircleDetection(obj,imageIndex,dMin,dMax,
                objectPolarity, circleSensitivity, edgeThreshold);
             showAllImages(obj);
             showImageWithImTool(obj,imageIndex);
             calcMeanImage(obj);
             value = backgroundCorrectionSingleMeasurementPoint(obj,coarseMethod,
                fineMethod,background,kernelSize,ntFactor,k,manualValue);
             value = iso11146Part1SingleMeasurementPoint(obj,centerGuessX,centerGuessY,
                integAreaGuessX,integAreaGuessY,convergeKriterium,faktor,pixelsize);
         end
         methods (Access = private)
             function setMeanImage(obj,value)
                 obj.meanImage = value;
             end
         end
    end
```

7.16: Function addMeasurement

```
1 function addMeasurement(obj,value)
2 %addMeasurement Adds a single measurement to the Measuring point dataset
3 obj.dataSet(end+1) = value;
4 end
```

 $7.17: \ Function \ background Correction Single Measurement Point$

```
function value = backgroundCorrectionSingleMeasurementPoint(obj,coarseMethod,
1
       fineMethod,background,kernelSize,ntFactor,k,manualValue)
   %BACKGROUNDCORRECTIONSINGLEMEASUREMENTPOINT perform background correction
   %for a Measuring point
4
   value = SingleMeasurementPoint(obj.getZPos());
6
   measPointDataSet = obj.getDataSet();
   [Useless,NoI] = size(measPointDataSet);
8
   for j = 1 : NoI
       fprintf('Performing background correction for image %i.\n',j);
       corrMeas = measPointDataSet(j).backgroundCorrectionSingleMeasurement(
           coarseMethod,fineMethod,background,kernelSize,ntFactor,k,manualValue);
       value.addMeasurement(corrMeas);
12
   end
   end
```

7.18: Function backgroundCorrectionSingleMeasurementPoint

```
function value = backgroundCorrectionSingleMeasurementPoint(obj,coarseMethod,
1
       fineMethod,background,kernelSize,ntFactor,k,manualValue)
2
   %BACKGROUNDCORRECTIONSINGLEMEASUREMENTPOINT perform background correction
3
   %for a Measuring point
4
   value = SingleMeasurementPoint(obj.getZPos());
   measPointDataSet = obj.getDataSet();
6
7
   [Useless,NoI] = size(measPointDataSet);
8
   for j = 1 : NoI
9
       fprintf('Performing background correction for image %i.\n',j);
       corrMeas = measPointDataSet(j).backgroundCorrectionSingleMeasurement(
           coarseMethod,fineMethod,background,kernelSize,ntFactor,k,manualValue);
       value.addMeasurement(corrMeas);
12
   end
   end
```

7.19: Function iso11146Part1SingleMeasurementPoint

```
function value = isoll146Part1SingleMeasurementPoint(obj,centerGuessX,centerGuessY,
       integAreaGuessX, integAreaGuessY, convergeKriterium, faktor, pixelsize)
   %BACKGROUNDCORRECTIONSINGLEMEASUREMENTPOINT perform ISO11146-1 for a
   %Measuring point.
   value = SingleMeasurementPoint(obj.getZPos());
   measPointDataSet = obj.getDataSet();
   [Useless,NoI] = size(measPointDataSet);
   dWx = [];
9
   dWy = [];
  azimutPhi = [];
11 beamCenterX = [];
12 beamCenterY = [];
13 W_X = [];
```

1

2

3

4

6

7

14	$W_Y = [];$			
15	W_XY = [];			
16	W_X_squared = [];			
17	W_Y_squared = [];			
18	for j = 1 : NoI			
19	fprintf('Computing ISO11146—1 parameters for image %i.\n',j);			
20	<pre>processedMeas = measPointDataSet(j).iso11146Part1SingleMeasurement(centerGuessX,</pre>			
	<pre>centerGuessY,integAreaGuessX,integAreaGuessY,convergeKriterium,faktor,</pre>			
	<pre>pixelsize);</pre>			
21	<pre>dWx = [dWx;processedMeas.getdWx()];</pre>			
22	dWy = [dWy;processedMeas.getdWy()];			
23	<pre>azimutPhi = [azimutPhi;processedMeas.getAzimutPhi()];</pre>			
24	<pre>beamCenterX = [beamCenterX;processedMeas.getbeamCenterX()];</pre>			
25	<pre>beamCenterY = [beamCenterY;processedMeas.getbeamCenterY()];</pre>			
26	$W_X = [W_X; processedMeas.getW_X()];$			
27	$W_Y = [W_Y; processedMeas.getW_Y()];$			
28	W_X_squared = [W_X_squared;processedMeas.getW_X_squared()];			
29	W_Y_squared = [W_Y_squared;processedMeas.getW_Y_squared()];			
30	W_XY = [W_XY;processedMeas.getW_XY()];			
31	<pre>value.addMeasurement(processedMeas);</pre>			
32	end			
33	<pre>value.dWx_mean = mean(dWx,'all');</pre>			
34	<pre>value.dWy_mean = mean(dWy,'all');</pre>			
35	<pre>value.azimutPhi_mean = mean(azimutPhi,'all');</pre>			
36	<pre>value.beamCenterX_mean = mean(beamCenterX,'all');</pre>			
37	<pre>value.beamCenterY_mean = mean(beamCenterY,'all');</pre>			
38	<pre>value.W_X_mean = mean(W_X,'all');</pre>			
39	<pre>value.W_Y_mean = mean(W_Y,'all');</pre>			
40	<pre>value.W_X_squared_mean = mean(W_X_squared,'all');</pre>			
41	<pre>value.W_Y_squared_mean = mean(W_Y_squared,'all');</pre>			
42	<pre>value.W_XY_mean = mean(W_XY,'all');</pre>			
43				
44	<pre>value.dWx_std = std(dWx);</pre>			
45	<pre>value.dWy_std = std(dWy);</pre>			
46	<pre>value.azimutPhi_std = std(azimutPhi);</pre>			
47	<pre>value.beamCenterX_std = std(beamCenterX);</pre>			
48	<pre>value.beamLenterY_std = std(beamLenterY);</pre>			
49	value.W_X_std = std(W_X);			
50	value.w_Y_std = std(w_Y);			
51	<pre>value.w_X_squared_std = std(w_X_squared);</pre>			
0Z	<pre>value.w_r_squared_std = std(w_r_squared); value.w/Y_std = std(w/YY);</pre>			
03 54	value.w_AT_SLU = SLU(W_AT);			
04	enu			

7.20: Function showAllImages

```
function showAllImages(obj)
  %SHOWALLEIMAGES show all Images for a Measuring point.
  [Useless NoI] = size(obj.dataSet);
4
 cols = 5;
```

1

2

```
rows = fix(NoI/cols)+1;
6
   restCols = rem(NoI,cols);
7
8
   temp =[];
9
   figure;
   for j=1 : NoI
        temp = [temp obj.dataSet(j).getImageData()];
12
   end
   montage(temp);
14
   % imageIndex = 1;
   % c = gray;
18
   % figure
19
   %
   %
          for j=1 : NoI
              subplot(rows,cols,j), imshow(obj.dataSet(imageIndex).getImage())
   %
   %
          end
   % end
```

7.21: Function showImage

```
function showImage(obj,imageIndex)
```

```
%SHOWIMAGE show a specified image of the measuring point.
```

- figure;
 - imshow(obj.dataSet(imageIndex).imageData);
- 6 end

7.22: Function showImageWithCircleDetection

```
function [centers, radii] = showImageWithCircleDetection(obj, imageIndex, dMin, dMax,
1
      objectPolarity, circleSensitivity, edgeThreshold)
2
  %SHOWIMAGEWITHCIRCLEDETECTION legacy code show image with matlab circle
3
  %detection. Not usable for our purpose but fun anyway.
4
   imageToShow = obj.dataSet(imageIndex).imageData;
   figure;
6
  imshow(imageToShow);
   [centers,radii] = imfindcircles(imageToShow,[dMin/2 dMax/2],'0bjectPolarity',
7
      objectPolarity, ...
       'Sensitivity', circleSensitivity, 'EdgeThreshold', edgeThreshold);
8
9
  viscircles(centers, radii, 'Color', 'r');
  end
```

 $7.23: Function \ showImageWithImTool$

```
1 function showImageWithImTool(obj,imageIndex)
2 %SHOWIMAGEWITHIMTOOL show a specified image in the imtool.
3 imtool(obj.dataSet(imageIndex).imageData);
4 end
```

classdef SingleMeasurement %MEASURMENTDATA Class containing one single measurement. properties (Access = public) imageData (1026,1282); beamCenterX double; beamCenterY double; W_X double; W_Y double; W_X_squared double; W_Y_squared double; W_XY double; dWx double; dWy double; phi double; end methods % function delete(obj) delete(obj.imageData); % % end function obj = SingleMeasurement(value) %MEASURMENTDATA Construct an instance of this class Detailed explanation goes here % obj.imageData = value; end function value = getImageData(obj) % Get stored measurment image data. value = obj.imageData; end function value = getdWx(obj) % Get stored measurment image data. value = obj.dWx; end function value = getdWy(obj) % Get stored measurment image data. value = obj.dWy; end function value = getAzimutPhi(obj) % Get stored measurment image data. value = obj.phi; end function value = getbeamCenterX(obj) % Get stored measurment image data. value = obj.beamCenterX; end

4

6

7

8

9

18

20

21

24

25

28 29

34

38

41

44

45

47

```
function value = getbeamCenterY(obj)
                                            % Get stored measurment image data.
                                            value = obj.beamCenterY;
                                    end
                                    function value = getW_X(obj)
                                            % Get stored measurment image data.
                                    end
            58
                                    end
W Bibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
Vier Nour knowledge hub
The approved original version of this thesis is available in print at TU Wien Bibliothek.
                                    end
                                    end
                                    end
            74
                    %
                            end
            78
                   end
              1
              2
                    plotBool=false;
              4
              6
              7
                    if plotBool
              8
              9
                   end
            12
```

```
value = obj.W_X;
function value = getW_Y(obj)
    % Get stored measurment image data.
    value = obj.W_Y;
function value = getW_X_squared(obj)
    % Get stored measurment image data.
    value = obj.W_X_squared;
function value = getW_Y_squared(obj)
    % Get stored measurment image data.
    value = obj.W_Y_squared;
function value = getW_XY(obj)
    % Get stored measurment image data.
    value = obj.W_XY;
value = backgroundCorrectionSingleMeasurement(obj,coarseMethod,fineMethod,
   background,kernelSize,ntFactor,k,manualValue);
value = calcValuesIS011146Part1(obj,centerGuessX,centerGuessY,
   integAreaGuessX, integAreaGuessY, convergeKriterium, faktor);
value = calcValuesIS011146Part2(obj,centerGuessX,centerGuessY,
    integAreaGuessX, integAreaGuessY, convergeKriterium, faktor);
```

7.25: Function backgroundCorrectionSingleMeasurement

```
function value = backgroundCorrectionSingleMeasurement(obj,coarseMethod,...
    fineMethod,background,kernelSize,ntFactor,k,manualValue)
singleMeasurementImageData = obj.getImageData();
[rows,cols] = size(singleMeasurementImageData);
% imtool(double(singleMeasurementImageData));
    imtool(singleMeasurementImageData);
    caxis([0 255])
    colorbar('FontSize',20);
switch coarseMethod
    case 'Background Map Substraction'
        backgroundSubstractionMap = background.getMeanSubstractionMapDouble(k);
```

<pre>backgroundsubtractionmap); for i=1:rows for j=1:cols if singleMeasurementImageData(i,j)<0 singleMeasurementImageData(i,j)=0; end end end % intool(double(singleMeasurementImageData)); if plotBool imtool(singleMeasurementImageData); caxis(10 2551) colorbar('FontSize',20); end value = SingleMeasurementImageData); case 'Average Background Substraction' E.b.offset = double(background.getMeanValue(k)); singleMeasurementImageData = imsubtract(double(singleMeasurementImageData), ones(rows,cols)+E.b.offset); % intool(double(singleMeasurementImageData)); for j=1:rows for j=1:cols if singleMeasurementImageData(i,j)<0 end end end end if plotBool intool(singleMeasurementImageData(i,j)=0; end end end if plotBool intool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end Nalue = SingleMeasurementImageData); kernel = [round(height+kernelSize/100) round(width+kernelSize/100)]; switch fineMethod case 'Statistical Method' % calc background getMeanStandradeviation(k); % calc background getMeanStandradeviation(k);</pre>	15	<pre>singleMeasurementImageData = imsubtract(double(singleMeasurementImageData),</pre>			
<pre>10</pre>	1.0	<pre>backgroundSubstractionMap); for i lumous</pre>			
<pre>11 to 'j=1:cols 12 if 'singleMeasurementImageData(i,j)=0 13 end 24 end 25 end 26 intool(double(singleMeasurementImageData)); 27 if plotBool 28 intool(singleMeasurementImageData); 29 cals(0 255]) 20 colorbar('FontSize',20); 20 value = SingleMeasurement(singleMeasurementImageData); 20 case 'Average Background Substraction' 21 E.b.offset = double(background.getMeanValue(k)); 22 singleMeasurementImageData = insubtract(double(singleMeasurementImageData), 23 ones(rows,cols)+E.b.offset); 34 intool(double(singleMeasurementImageData)); 35 for j=1:cows 36 for j=1:cols 37 if singleMeasurementImageData(i,j)=0; 38 end 40 end 41 if plotBool 42 intool(singleMeasurementImageData(i,j)=0; 43 end 44 end 44 if plotBool 45 ingleMeasurementImageData); 46 casis([0 255]) 47 colorbar('FontSize',20); 46 value = SingleMeasurementImageData); 47 end 48 value = SingleMeasurementImageData); 49 kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; 40 switch fineMethod 41 case 'Statistical Method' 42 %% calc background guess for E.b.offset 43 E.b.offsetGuess = background.getMeanValue(k); 44 % calc background guess for standard deviation 45 E.b.SigmaGuess = background.getMeanValue(k); 45 % calc background guess for standard deviation 46 avgilt = fspecial('average',kernel); 47 castedConvImage = imflter(singleMeasurementImageData, avgFilt); 48 cant = 0; 49 counter = 0; 40 counter = 0</pre>	10	TOR 1=1:rows			
<pre>11 singleMeasurementImageData(1,)>>></pre>	1 <i>(</i>	TOR J=1:COLS			
<pre>singleMeasurementImageData(1,))=0; end end % imtool(double(singleMeasurementImageData)); if plotBool imtool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); case 'Average Background Substraction' E.b.offset = double(background.getMeanValue(k)); singleMeasurementImageData = imsubtract(double(singleMeasurementImageData), ones(rows,cols)+E.b.offset); % imtool(double(singleMeasurementImageData)); for i=1:rows for j=1:cols if singleMeasurementImageData(i,j)<0 singleMeasurementImageData(i,j)=0; end end end if plotBool imtool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurementImageData); casis([0 255]) colorbar('FontSize',20); end [height,width] = size(singleMeasurementImageData); kernel = [round(height+kernelSize/100) round(width+kernelSize/100)]; switch fineMethod case 'Statistical Method' % calc background guess for E.b.offset E b.offsetGuess = background.getMeanStandardDeviation(k); % calc background Guess for standard deviation E.b.Sigma_Guess = background.getMeanStandardDeviation(k); % calc background guess for standard deviation avgFilt = fspecial('average', kernel); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E.b.Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0;</pre>	10	<pre>ingleMeasurementImageData(1, j)<0 singleMeasurementImageData(i, j)=0;</pre>			
<pre>end end end end end end end end end end</pre>	19	singlemeasurementimageData(1, j)=0;			
<pre>end end % imtool(double(singleMeasurementImageData)); if plotBool intool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); case 'Average Background Substraction' E.b.offset = double(background.getMeanValue(k)); singleMeasurementImageData = insubtract(double(singleMeasurementImageData), ones(rows,cols)*E.b.offset); % intool(double(singleMeasurementImageData)); for i=1:rows for j=1:cols if singleMeasurementImageData(i,j)<0 if or i=1:rows for j=1:cols end end end if plotBool if plotBool if plotBool if singleMeasurementImageData(i,j)=0; end colorbar('FontSize',20); end kernel = [round(height+kernelSize/100) round(width+kernelSize/100)]; switch fineMethod case 'Statistical Method' % calc background guess for E.b.offset E.b.offsetGuess = background.getMeasurementImageData, avgFilt); E.b.SigmaGuess = infilter(singleMeasurementImageData, avgFilt); E.b.Sigma_Uases = infilter(singleMeasurementImageData, avgFilt); E.b.Si</pre>	20 01	end			
<pre>22 end imtool(double(singleMeasurementImageData)); 23 if plotBool 24 intool(singleMeasurementImageData); 25 caxis([0 255]) 26 colorbar('FontSize',20); 27 end 28 value = SingleMeasurement(singleMeasurementImageData); 29 case 'Average Background Substraction' 29 E.b.offset = double(background.getMeanValue(k)); 20 singleMeasurementImageData = insubtract(double(singleMeasurementImageData), 20 ones(rows,cols)*E.b.offset); 30 % imtool(double(singleMeasurementImageData)); 31 for i=1:rows 32 for j=1:cols 33 for j=1:cols 34 if singleMeasurementImageData(i,j)<0 35 if singleMeasurementImageData(i,j)<0 36 end 40 end 41 if plotBool 42 imtool(singleMeasurementImageData); 43 caxis([0 255]) 44 colorbar('FontSize',20); 45 end 46 value = SingleMeasurementImageData); 47 end 48 value = SingleMeasurementImageData); 49 kerel = [round(height+kernelSize/100) round(width+kernelSize/100)]; 40 switch fineWethod 41 Suitstical Method' 42 Suitstical Method' 43 w calc background guess for ELb-offset 44 E.b.offsetGuess = background.getMeanValue(k); 45 % calc background Guess for standard deviation 46 E.b.SigmaGuess = background.getMeanValue(k); 47 % calc background Guess for standard deviation 48 E.b.offsetGuess = background.getMeanValue(k); 49 % calc background Guess for standard deviation 40 E.b.SigmaTilde = std(double(reshape(castedConvImage,[],1))); 40 temp = 0; 41 counter = 0; 42 counter = 0; 43 counter = 0; 44 counter = 0; 45 counter</pre>	21 99	ena			
<pre>inftot(touble(singleMeasurementImageData); if plotBool imtool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); case 'Average Background Substraction' E_b_offset = double(background.getMeanValue(k)); singleMeasurementImageData = insubtract(double(singleMeasurementImageData), ones(rows,cols)+E_b_offset); intool(double(singleMeasurementImageData)); for i=1:rows for j=1:cols intool(double(singleMeasurementImageData)); for i=1:rows for j=1:cols if singleMeasurementImageData(i,j)=0; end end</pre>	44 02	enu			
<pre>1</pre>	20 94	<pre>% Imtoot(double(singtemeasurementimageData)); if plotDecl</pre>			
<pre>induct(singtemeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); case 'Average Background Substraction' E.b.offset = double(background.getMeanValue(k)); singleMeasurementImageData = insubtract(double(singleMeasurementImageData), ones(rows.cols) = E.b.offset); % imtool(double(singleMeasurementImageData)); for i=1:rows for j=1:cols if singleMeasurementImageData(i,j) = 0; end end end end if plotBool intool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); switch fineMethod case 'Statistical Method' % calc background guess for E.b.offset E.b.offsetGuess = background.getMeanValue(k); & cated ConvImage = imfilter(singleMeasurementImageData, avgFilt); E.b.Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0:</pre>	24 95	II plotbool			
<pre>colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); ccase 'Average Background Substraction' E.b.offset = double(background.getMeanValue(k)); singleMeasurementImageData = insubtract(double(singleMeasurementImageData), ones(rows.cols)*E.b.offset); % intool(double(singleMeasurementImageData)); for i=1:rows for j=1:cols if singleMeasurementImageData(i,j)<0 singleMeasurementImageData(i,j)=0; end end end if plotBool intool(singleMeasurementImageData); ccase(j0 255)) cclorbar('FontSize',20); end value = SingleMeasurement[singleMeasurementImageData); end [height,width] = size(singleMeasurementImageData); end [height,width] = size(singleMeasurementImageData); witch fineMethod ccase 'Statistical Method' % calc background guess for E.b_offset E.b.offsetGuess = background.getMeanValue(k); E.b.SigmaGuess = background.getMeanValue(k); % calc 2-D averaging subarray convolution avgFilt = fspecial('average',kernel); ccastedConvImage = imfilter(singleMeasurementImageData, avgFilt); E.b.Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0:</pre>	20 96	covic(10,2551)			
<pre>control and (for (single (for (for (for (for (for (for (for (for</pre>	$\frac{20}{97}$	calarbar([5,25])			
<pre>value = SingleMeasurement(singleMeasurementImageData); case 'Average Background Substraction' E_b_offset = double(background.getMeanValue(k)); singleMeasurementImageData = insubtract(double(singleMeasurementImageData), ones(rows,cols)*E_b_offset); % imtool(double(singleMeasurementImageData)); for i=1:rows for j=1:cols if singleMeasurementImageData(i,j)<0 singleMeasurementImageData(i,j)=0; end end end if plotBool imtool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurementImageData); kernel = [round(height*kernelSize/100)]; switch fineMethod case 'Statistical Method' % calc background guess for E_b_offset E_b_offsetGuess = background.getMeanValue(k); % calc background Guess for standard deviation E_b_SigmaGuess = background.getMeanValue(k); % % calc 2-D averaging subarray convolution avgFilt = fspecial('average',kernel); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E_b_Sigma_Tide = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0:</pre>	41 28	cotorbar (FontSize , 20),			
<pre>case 'Average Background Substraction' case 'Average Background Substraction' Lb_offset = double(background.getMeanValue(k)); singleMeasurementImageData = insubtract(double(singleMeasurementImageData),</pre>	20 20	value = SingleMeasurement(singleMeasurementImageData);			
<pre>Else intrody but got your about returns E_b_offset = double(background.getMeanValue(k)); singleMeasurementImageData = insubtract(double(singleMeasurementImageData), ones(rows,cols)*E_b_offset); % introl(double(singleMeasurementImageData)); for j=1:rows for j=1:cols if singleMeasurementImageData(i,j)<0 singleMeasurementImageData(i,j)=0; end end end end if plotBool intool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end [height,width] = size(singleMeasurementImageData); kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; switch fineMethod case 'Statistical Method' % calc background guess for E_b_offset E_b_offsetGuess = background.getMeanValue(k); % calc background Guess for standard deviation E_b_SigmaGuess = background.getMeanStandardDeviation(k); % calc 2-D averaging subarray convolution avgFilt = fspecial('average',kernel); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E_b_Sigma_IIde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0:</pre>	29 30	case 'Average Background Substraction'			
<pre>singleMeasurementImageData = insubtract(double(singleMeasurementImageData),</pre>	31	E = h offset = double(background getMeanValue(k));			
<pre>singleHeadIntementImageData = insource(inductedingleHeadIntementImageData); ones(rows.cols)*E.b.offset); if or i=1:rows for j=1:cols if singleMeasurementImageData(i,j)=0; end end end end if plotBool imtool(singleMeasurementImageData(i,j)=0; end end caris([0 255]) colorbar('FontSize',20); end value = SingleMeasurementImageData); end [height,width] = size(singleMeasurementImageData); value = SingleMeasurement(singleMeasurementImageData); end [height,width] = size(singleMeasurementImageData); witch fineMethod case 'Statistical Method' % calc background guess for E.b.offset E.b.offsetGuess = background.getMeanStandardDeviation(k); % calc background Guess for standard deviation E.b.SigmaGuess = background.getMeanStandardDeviation(k); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E.b.Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0; counter = 0; counter = 0;</pre>	30	L_D_0 of the set = doubte(backyround.getheanvalue(k)), singleMeasurementImageData = imsubtract(double(singleMeasurementImageData)			
<pre>bills(function); bills(function); introd(double(singleMeasurementImageData)); for i=1:rows for j=1:cols if singleMeasurementImageData(i,j)<0 singleMeasurementImageData(i,j)=0; end end end end introd(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); end [height,width] = size(singleMeasurementImageData); kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; switch fineMethod case 'Statistical Method' % calc background guess for E_b_offset E_b_offsetGuess = background.getMeanValue(k); % calc background Guess for standard deviation E_b_SigmaGuess = background.getMeanValue(k); % calc 2-D averaging subarray convolution avgFilt = fspecial('average',kernel); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0; counter = 0; counter = 0;</pre>	04	sing teneasurement imagebata = imsubil act (doubte(sing teneasurement imagebata),ones(rows cols) *E b offset):			
<pre>for i=1:rows for j=1:cols for j=1:cols</pre>	33	<pre>imtool(double(singleMeasurementImageData));</pre>			
<pre>int in the interval inter</pre>	34	for i=1:rows			
<pre>if singleMeasurementImageData(i,j)<0 if singleMeasurementImageData(i,j)<0 isingleMeasurementImageData(i,j)=0; end end end if plotBool imtool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); end [height,width] = size(singleMeasurementImageData); kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; switch fineMethod case 'Statistical Method' % calc background guess for E_b_offset E_b_offsetGuess = background.getMeanValue(k); % calc background Guess for standard deviation E_b_SigmaGuess = background.getMeanValue(k); % calc 2-D averaging subarray convolution avgFilt = fspecial('average',kernel); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0:</pre>	35	for i=1:cols			
<pre>singleMeasurementImageData(i,j) d singleMeasurementImageData(i,j) d singleMeasurementImageData(i,j) d singleMeasurementImageData(i,j) d end end if plotBool caris([0 255]) colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); end kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; switch fineMethod case 'Statistical Method' switch fineMethod E_b_SigmaGuess = background.getMeanValue(k); switch fiseMeasures = background.getMeanStandardDeviation(k); swifit = fspecial('average',kernel); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0; </pre>	36	if singleMeasurementImageData(i,i)<0			
<pre>end end end if plotBool imtool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); end (height,width] = size(singleMeasurementImageData); kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; switch fineMethod case 'Statistical Method' % calc background guess for E_b_offset E_b_offsetGuess = background.getMeanValue(k); % calc background Guess for standard deviation E_b_SigmaGuess = background.getMeanStandardDeviation(k); % calc 2-D averaging subarray convolution avgFilt = fspecial('average',kernel); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0:</pre>	37	singleMeasurementImageData(i i)=0:			
<pre>end end if plotBool imtool(singleMeasurementImageData); caxis([0 255]) colorbar('FontSize',20); end value = SingleMeasurement(singleMeasurementImageData); end [height,width] = size(singleMeasurementImageData); kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; switch fineMethod case 'Statistical Method' % calc background guess for E_b_offset E_b_offsetGuess = background.getMeanValue(k); Sw calc background Guess for standard deviation E_b_SigmaGuess = background.getMeanStandardDeviation(k); % calc 2-D averaging subarray convolution avgFilt = fspecial('average',kernel); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0:</pre>	38	end			
<pre>40 end 41 if plotBool 42 imtool(singleMeasurementImageData); 43 caxis([0 255]) 44 colorbar('FontSize',20); 45 end 46 value = SingleMeasurement(singleMeasurementImageData); 47 end 48 [height,width] = size(singleMeasurementImageData); 49 kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; 50 switch fineMethod 51 case 'Statistical Method' 52 %% calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 %% calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 %% calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 59 counter = 0; 50 counter = 0; 50 counter = 0; 51 counter = 0; 52 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 59 counter = 0; 50 counter = 0; 50 counter = 0; 51 counter = 0; 52 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 58</pre>	39	end			
<pre>41 if plotBool 42</pre>	40	end			
<pre>42 imtool(singleMeasurementImageData); 43 caxis([0 255]) 44 colorbar('FontSize',20); 45 end 46 value = SingleMeasurement(singleMeasurementImageData); 47 end 48 [height,width] = size(singleMeasurementImageData); 49 kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; 50 switch fineMethod 51 case 'Statistical Method' 52 % calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 % calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 % calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 60 temp = 0; 61 counter = 0; 61 counter</pre>	41	if plotBool			
<pre>43</pre>	42	<pre>imtool(singleMeasurementImageData);</pre>			
<pre>44 colorbar('FontSize',20); 45 end 46 value = SingleMeasurement(singleMeasurementImageData); 47 end 48 [height,width] = size(singleMeasurementImageData); 49 kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; 50 switch fineMethod 51 case 'Statistical Method' 52 % calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 % calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 % calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0; 61 counter = 0;</pre>	43	caxis([0 255])			
<pre>45 end 46 value = SingleMeasurement(singleMeasurementImageData); 47 end 48 [height,width] = size(singleMeasurementImageData); 49 kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; 50 switch fineMethod 51 case 'Statistical Method' 52 % calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 % calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 % calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0; 51 counter = 0; 52 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 59 counter = 0; 50 counter = 0; 50 counter = 0; 51 counter = 0; 52 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 59 counter = 0; 50 counter = 0; 50 counter = 0; 51 counter = 0; 52 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 59 counter = 0; 50 counter = 0; 50 counter = 0; 51 counter = 0; 52 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 59 counter = 0; 50 counter = 0; 50 counter = 0; 50 counter = 0; 51 counter = 0; 5</pre>	44	<pre>colorbar('FontSize',20);</pre>			
<pre>46 value = SingleMeasurement(singleMeasurementImageData); 47 end 48 [height,width] = size(singleMeasurementImageData); 49 kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; 50 switch fineMethod 51 case 'Statistical Method' 52 % calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 % calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 % calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0;</pre>	45	end			
<pre>47 end 48 [height,width] = size(singleMeasurementImageData); 49 kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; 50 switch fineMethod 51 case 'Statistical Method' 52 % calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 % calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 % calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 51 counter = 0; 52 counter = 0; 53 counter = 0; 54 calc 2-D averaging subarray convolution 55 castedConvImage = o; 56 calc 2-D averaging et mathematical castedConvImage,[],1)); 59 counter = 0; 50 counter = 0; 51 counter = 0; 52 counter = 0; 53 calc 2-D averaging et mathematical castedConvImage,[],1); 54 castedConvImage = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 59 counter = 0; 50 counter = 0; 50 counter = 0; 51 counter = 0; 52 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 59 counter = 0; 50 counter = 0; 50 counter = 0; 51 counter = 0; 52 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 58 counter = 0; 59 counter = 0; 50 counter = 0; 50 counter = 0; 51 counter = 0; 51</pre>	46	<pre>value = SingleMeasurement(singleMeasurementImageData);</pre>			
<pre>48 [height,width] = size(singleMeasurementImageData); 49 kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; 50 switch fineMethod 51 case 'Statistical Method' 52 % calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 % calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 %% calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0;</pre>	47	end			
<pre>49 kernel = [round(height*kernelSize/100) round(width*kernelSize/100)]; 50 switch fineMethod 51 case 'Statistical Method' 52 % calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 % calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 % calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0; 53 </pre>	48	<pre>[height,width] = size(singleMeasurementImageData);</pre>			
<pre>switch fineMethod case 'Statistical Method' %% calc background guess for E_b_offset E_b_offsetGuess = background.getMeanValue(k); %% calc background Guess for standard deviation E_b_SigmaGuess = background.getMeanStandardDeviation(k); %% calc 2-D averaging subarray convolution avgFilt = fspecial('average',kernel); castedConvImage = imfilter(singleMeasurementImageData, avgFilt); E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); temp = 0; counter = 0;</pre>	49	<pre>kernel = [round(height*kernelSize/100) round(width*kernelSize/100)];</pre>			
<pre>51 case 'Statistical Method' 52 %% calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 %% calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 %% calc 2—D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0;</pre>	50	switch fineMethod			
<pre>52 %% calc background guess for E_b_offset 53 E_b_offsetGuess = background.getMeanValue(k); 54 %% calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 %% calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0;</pre>	51	<pre>case 'Statistical Method'</pre>			
<pre>53 E_b_offsetGuess = background.getMeanValue(k); 54 %% calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 %% calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 51 counter = 0; 53 counter = 0; 54 counter = 0; 55 counter = 0; 56 counter = 0; 57 counter = 0; 57 counter = 0;</pre>	52	<pre>%% calc background guess for E_b_offset</pre>			
<pre>54 %% calc background Guess for standard deviation 55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 %% calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0;</pre>	53	<pre>E_b_offsetGuess = background.getMeanValue(k);</pre>			
<pre>55 E_b_SigmaGuess = background.getMeanStandardDeviation(k); 56 % calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0;</pre>	54	<pre>%% calc background Guess for standard deviation</pre>			
<pre>56 %% calc 2-D averaging subarray convolution 57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 50 temp = 0; 61 counter = 0;</pre>	55	<pre>E_b_SigmaGuess = background.getMeanStandardDeviation(k);</pre>			
<pre>57 avgFilt = fspecial('average',kernel); 58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 60 temp = 0; 61 counter = 0;</pre>	56	%% calc 2—D averaging subarray convolution			
<pre>58 castedConvImage = imfilter(singleMeasurementImageData, avgFilt); 59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 60 temp = 0; 61 counter = 0;</pre>	57	<pre>avgFilt = fspecial('average',kernel);</pre>			
<pre>59 E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1))); 60 temp = 0; 61 counter = 0;</pre>	58	castedConvImage = imfilter(singleMeasurementImageData, avgFilt);			
$\begin{array}{c} 60 \\ 61 \\ 61 \\ counter = 0; \\ counter = 0; \\ \end{array}$	59	<pre>E_b_Sigma_Tilde = std(double(reshape(castedConvImage,[],1)));</pre>			
61 counter = 0:	60	temp = 0;			
	61	counter = 0;			

```
62
             for i=1:rows
                 for j=1:cols
                     if double(castedConvImage(i,j)) < double(E_b_offsetGuess)+ntFactor*</pre>
                         E_b_Sigma_Tilde/(sqrt((kernel(1)+1)*(kernel(2)+1)))
                         if singleMeasurementImageData(i,j) >0
                              temp = temp + double(singleMeasurementImageData(i,j));
                              counter = counter+1;
                          end
                     end
                 end
71
             end
             E_b_offset = temp/counter;
             singleMeasurementImageData = imsubtract(singleMeasurementImageData,ones(rows
                 ,cols)*E_b_offset);
             for i=1:rows
74
                 for j=1:cols
                     if singleMeasurementImageData(i,j)<0</pre>
                          singleMeasurementImageData(i,j)=0;
78
                     end
                 end
             end
81
                       imtool(double(singleMeasurementImageData));
             %
82
             if plotBool
83
                 imtool(singleMeasurementImageData);
                 caxis([0 255])
84
85
                 colorbar('FontSize',20);
86
             end
87
             value = SingleMeasurement(singleMeasurementImageData);
88
89
         case 'Approximation Method'
90
             E_b_offsetDark = background.getMeanFromCorners(k,kernel,kernel);
             %% calc background Guess for standard deviation
             E_b_SigmaDark=background.getMeanStandardDeviation(k);
             %% calc E_b_offsetMeas
94
             original=obj.getImageData();
             temp = 0;
96
             counter = 0;
             for i=1:rows
98
                 for j=1:cols
99
                     if (i <= kernel(1) && j <= kernel(2)) ...</pre>
                              || (i <= kernel(1) && j >= cols_kernel(2)+1) ...
                              || (i >= rows-kernel(1)+1 && j <= kernel(2)) ...</pre>
                              || (i >= rows_kernel(1)+1 && j >= cols_kernel(2)+1)
                         temp = temp + double(original(i,j));
                          counter = counter + 1;
                     end
106
                 end
             end
             E_b_offsetMeas = temp/counter;
```

```
Jibliothek, Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
Nour knowledge hub. The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
if abs(double(E_b_offsetMeas) - double(E_b_offsetDark))*sqrt(counter)/double
                 (E_b_SigmaDark) < ntFactor
                 singleMeasurementImageData = imsubtract(singleMeasurementImageData,cast(
                     ones(rows,cols)*E_b_offsetMeas,'uint8'));
                 for i=1:rows
                     for j=1:cols
                         if singleMeasurementImageData(i,j)<0</pre>
                             singleMeasurementImageData(i,j)=0;
                         end
                     end
                 end
118
                               imtool(double(singleMeasurementImageData));
                 %
                 if plotBool
                     imtool(singleMeasurementImageData);
                     caxis([0 255])
                     colorbar('FontSize',20);
                 end
                 value = SingleMeasurement(singleMeasurementImageData);
             else
                 value = SingleMeasurement(zeros(height,width));
128
                 fprintf('\nDifference between dark backgroundimage and measrumentimage
                     is too big.\nLaserbeam might be too big for Sensor\n');
             end
                   case 'Median Filter 3x3'
             %
                       singleMeasurementImageData = medfilt2(singleMeasurementImageData
             %
                 ,[3 3]);
                       value = SingleMeasurement(singleMeasurementImageData);
             %
             %
                   case 'Median Filter 5x5'
                       singleMeasurementImageData = medfilt2(singleMeasurementImageData
             %
                 ,[5 5]);
             %
                       value = SingleMeasurement(singleMeasurementImageData);
             %
         case 'Manual Value'
             E_b_offset = manualValue;
             singleMeasurementImageData = imsubtract(singleMeasurementImageData,ones(rows
                 ,cols)*E_b_offset);
             for i=1:rows
                 for j=1:cols
                     if singleMeasurementImageData(i,j)<0</pre>
                         singleMeasurementImageData(i,j)=0;
                     end
                 end
             end
             if plotBool
148
                 imtool(singleMeasurementImageData);
                 caxis([0 255])
                 colorbar('FontSize',20);
             end
```

```
Bibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar Vour knowledge hub. The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

152		<pre>value = SingleMeasurement(singleMeasurementImageData);</pre>
153	end	
154		
155	end	
	L	

 $7.26: Function\ iso11146 Part 1Single Measurement$

```
function value = iso11146Part1SingleMeasurement(obj,centerGuessX,centerGuessY,
 1
       integAreaGuessX,integAreaGuessY,convergeKriterium,faktor,pixelsize)
   %CALCVALUESIS011146PART1 calc LB parameters using 11146—1 for single
3
   %Measurement
4
   %% set initial Values
6
   oldCenterX = 0;
   oldCenterY = 0;
8
   old_dSigmaX = 0;
9
   old_dSigmaY = 0;
11
   newCenterX = centerGuessX;
12
   newCenterY = centerGuessY;
   new_dWx = integAreaGuessX;
14
   new_dWy = integAreaGuessY;
15
   originalImage = obj.getImageData();
   centerInPixelX = round(newCenterX/pixelsize);
   centerInPixelY = round(newCenterY/pixelsize);
18
   %% start IterationProcess
   while abs(newCenterX—oldCenterX) > convergeKriterium ||...
20
            abs(newCenterY—oldCenterY) > convergeKriterium ||...
21
            abs(new_dWx-old_dSigmaX) > convergeKriterium ||...
            abs(new_dWy-old_dSigmaY) > convergeKriterium
24
        if mod(fix(faktor*new_dWx/pixelsize),2) == 0
            intAreaInPixelX = fix(faktor*new_dWx/pixelsize+1);
        else
27
            intAreaInPixelX = fix(faktor*new_dWx/pixelsize);
28
        end
        if mod(fix(faktor*new_dWy/pixelsize),2) == 0
            intAreaInPixelY = fix(faktor*new_dWy/pixelsize+1);
        else
            intAreaInPixelY = fix(faktor*new_dWy/pixelsize);
        end
        %% create Subimage used for calc
        subImage = originalImage(centerInPixelY-floor(intAreaInPixelY/2):centerInPixelY+
           floor(intAreaInPixelY/2),...
            centerInPixelX-floor(intAreaInPixelX/2):centerInPixelX+floor(intAreaInPixelX
               /2));
        %% create Mesh for integration
```

```
41
        [rows,cols]=size(subImage);
        x = -floor(cols/2)*pixelsize:pixelsize:floor(cols/2)*pixelsize;
43
        y = floor(rows/2)*pixelsize:-pixelsize:-floor(rows/2)*pixelsize;
44
        [X,Y] = meshgrid(x,y);
        %% calc newCenter
        F = double(subImage).*X;
47
        I = trapz(y, trapz(x, F, 2));
48
        I2 = trapz(y,trapz(x,double(subImage),2));
        oldCenterX = newCenterX;
       W_X = I/I2;
        F = double(subImage).*Y;
        I = trapz(y, trapz(x, F, 2));
        I2 = trapz(y,trapz(x,double(subImage),2));
        oldCenterY = newCenterY;
       W_{-}Y = I/I2;
58
        centerInPixelX = centerInPixelX+round(W_X/pixelsize);
        centerInPixelY = centerInPixelY-round(W_Y/pixelsize);
        newCenterY = centerInPixelY*pixelsize;
        newCenterX = centerInPixelX*pixelsize;
        %% calc Sigma_X_squared
        F = double(subImage).*((X-W_X).^2);
        I = trapz(y, trapz(x, F, 2));
        I2 = trapz(y,trapz(x,double(subImage),2));
       W_X_squared = I/I2;
        %% calc Sigma_Y_squared
        F = double(subImage).*((Y-W_Y).^2);
        I = trapz(y, trapz(x, F, 2));
        I2 = trapz(y,trapz(x,double(subImage),2));
72
        W_Y_squared = I/I2;
        %% calc Sigma_XY_squared
74
        F = double(subImage).*((X-W_X).*(Y-W_Y));
        I = trapz(y, trapz(x, F, 2));
        I2 = trapz(y,trapz(x,double(subImage),2));
78
       W_XY = I/I2;
        %% different equations used depending if Sigma_X_squared == Sigma_Y_squared
        if abs(W_X_squared_W_Y_squared) < le8*eps(min(abs(W_X_squared),abs(W_Y_squared))</pre>
80
            )
            %% calc Azimutwinkel
81
82
            if W_XY == 0
83
                azimutSign = 1;
84
            else
85
                azimutSign = W_XY/abs(W_XY);
86
            end
87
            phi = azimutSign*45; % or pi/4
88
            %% calc new_dSigmaX
```

89	<pre>old_dSigmaX = new_dWx;</pre>
90	$new_dWx = 2*sqrt(2)*(W_X_squared+W_Y_squared+2*abs(W_XY))^(1/2);$
91	%% calc new_dSigmaY
92	<pre>old_dSigmaY = new_dWy;</pre>
93	<pre>new_dWy = 2*sqrt(2)*(W_X_squared+W_Y_squared-2*abs(W_XY))^(1/2);</pre>
94	else
95	<pre>phi = 0.5*atand(2*W_XY/(W_X_squared-W_Y_squared));</pre>
96	%% calc gamma is 1 or —1
97	gamma = (W_X_squared-W_Y_squared) / abs(W_X_squared-W_Y_squared);
98	%% calc new_dSigmaX
99	<pre>old_dSigmaX = new_dWx;</pre>
100	<pre>new_dWx=2*sqrt(2)*((W_X_squared+W_Y_squared)+gamma*((W_X_squared-W_Y_squared))</pre>
1.0.1)^2+4*W_XY^2)^(1/2);
101	%% CALC NEW_ASIGMAY
102	old_dSigmay = new_dwy;
109	$\text{Hew}_{\text{dwy}} = 2 \text{sqrt}(2) \text{squareu} + \frac{1}{2} \text{squareu} + \frac{1}{2} \text{squareu} - \frac{1}{2} \text{squareu} + \frac{1}{$
104	w_f_Squareu) 2+4*w_Xf 2) (1/2)) (1/2);
104 105	and
100	end
100 107	epsiton = min([new_uwx,new_uwy])/max([new_uwx,new_uwy]);
107	assumed to be a circle
108	if $ansilon > 0.87$
100	new dWx = $2*sart(2)*(W X squared+W Y squared)^(1/2)$.
110	$new_dW_x = 2*sqrt(2)*(W_X_squared+W_Y_squared)^{(1/2)};$
111	end
112	end
113	obi,beamCenterX = newCenterX:
114	obj.beamCenterY = newCenterY:
115	$obj.W_X = W_X$:
116	$obj.W_Y = W_Y;$
117	obj.W_X_squared = W_X_squared;
118	obj.W_Y_squared = W_Y_squared;
119	$obj.W_XY = W_XY;$
120	obj.dWx = new_dWx;
121	$obj.dWy = new_dWy;$
122	obj.phi = phi;
123	<pre>value = obj;</pre>
124	end

7.27: Class Result

classdef Result
%RESULT Measurement evaluation result containing processed data and
%calced parameters

properties
 backgroundCorrMethod string;
 evalNorm string;

TU **Bibliothek**, Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar WIEN ^{vourknowedge hub} The approved original version of this thesis is available in print at TU Wien Bibliothek.

 $\frac{1}{2}$

4 5 6

7

```
9
            lambda double;
            convCriterion double;
            nT double
12
            integAreaFactor double
            evaluatedMeasurementData MeasurementData = MeasurementData();
14
            kernelSizeInPercent double;
        end
       methods
18
            function obj = Result(coarseBackCorrMethod,fineBackgCorrMethod,evalNorm,
               lambda,convCriterion,nT,integAreaFactor,kernelSizeInPercent,
               evaluatedMeasurementData)
                %RESULT Construct an instance of this class
20
                    Detailed explanation goes here
                %
                switch coarseBackCorrMethod
                    case 'Background Map Substraction'
                        p1 = 'BMS';
                    case 'Average Background Substraction'
                        p1 = 'ABS';
                end
                switch fineBackgCorrMethod
28
                    case 'Statistical Method'
                        p2 = 'SM';
                    case 'Manual Value'
                        p2 = 'MV';
32
                    case 'Off'
                        p2 = '0ff';
34
                end
                obj.backgroundCorrMethod = sprintf('%s/%s',p1,p2);
                obj.lambda = lambda;
                obj.evalNorm = evalNorm;
                obj.convCriterion = convCriterion;
                obj.integAreaFactor = integAreaFactor;
                obj.nT = nT;
                obj.evaluatedMeasurementData = evaluatedMeasurementData;
41
                obj.kernelSizeInPercent = kernelSizeInPercent;
            end
44
            function value = getEvaluatedMeasurementData(obj)
45
                % Get stored measurment data.
                value = obj.evaluatedMeasurementData;
47
            end
48
            function value = getBackgroundCorrMethod(obj)
                % Get stored measurment data.
                value = obj.backgroundCorrMethod;
            end
            function value = getLambda(obj)
                % Get stored measurment data.
                value = obj.lambda;
            end
```

```
Bibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
Nour knowledge hub The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
Jibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
Vour knowedge hub The approved original version of this thesis is available in print at TU Wien Bibliothek.
                                                                    71
                                                                    72
                                                                             2
                                                                             1
                                                                             4
                                                                             6
                                                                             7
                                                                             8
                                                                           9
                                                                    14
                                                                    15
```

19

```
function value = getConvCriterion(obj)
        % Get stored measurment data.
        value = obj.convCriterion;
    end
    function value = getIntegAreaFactor(obj)
        % Get stored measurment data.
        value = obj.integAreaFactor;
    end
    function value = getNt(obj)
        % Get stored measurment data.
        value = obj.nT;
    end
    function value = getEvalNorm(obj)
        % Get stored measurment data.
        value = obj.evalNorm;
    end
    function value = getKernelSizeInPercent(obj)
        % Get stored measurment data.
        value = obj.kernelSizeInPercent;
    end
end
```

end

58

7.28: Function lbcUI Launcher

```
addpath('../backend');
fig = lbcUI;
```

7.29: Function CalculatePacketDelay

```
function [delay] = CalculatePacketDelay(vid, fps)
   \% CalculatePacketDelay Calculates recommended packet delay for a GigE Vision camera.
   %
   %
        DELAY = CalculatePacketDelay(VID, FPS) calculates recommended packet
   %
        delay DELAY for a GigE Vision camera, represented by an image
   %
        acquisition videoinput or gigecam object VID.
   %
        Camera framerate FPS (frames per second) is required and is specific
   %
         to the camera configuration. The companion CalculateFrameRate function
   %
         can be used to measure the camera framerate, if unknown.
   %
   %
        Example with videoinput vid:
           vid = videoinput('gige', 1);
   %
   %
           src = vid.Source;
   %
           src.PacketSize = 9000;
   %
           fps = CalculateFrameRate(vid, 20);
   %
           delay = CalculatePacketDelay(vid, fps);
           src.PacketDelay = delay;
   %
   %
   %
        Example with gigecam g:
20 %
           g = gigecam;
```

```
21
   %
            g.GevSCPSPacketSize = 9000;
   %
            fps = CalculateFrameRate(q, 20);
23
   %
            delay = CalculatePacketDelay(g, fps);
24
            g.GevSCPD = delay;
   %
   %
        Copyright (c) 2013 — 2015 The MathWorks, Inc.
28
    if nargin < 2</pre>
29
        error('CalculatePacketDelay(vid, fps) videoinput/gigecam object and framerate
           are required arguments to run this utility.');
   else
        if ~isnumeric(fps)
            error('CalculatePacketDelay(vid, fps) fps must be a number');
        end
        if ~(isscalar(fps) && isfinite(fps) && (fps > 0))
            error('CalculatePacketDelay(vid, fps) fps must finite and > 0 to find a
                frame rate');
        end
   end
    if ~(isa(vid, 'videoinput') || isa(vid, 'gigecam')) || ~isvalid(vid) || ~isscalar(
       vid)
        error('CalculatePacketDelay(vid, fps) vid must be a valid videoinput or gigecam
           object');
41
   end
42
43
    % get — packetSize (GevSCPSPacketSize)
          — TickFreq (GevTimestampTickFrequency)
44
   %
45
    %
          — video format (PixelFormat)
          — frame height
   %
47
           — frame width
    %
48
    switch class(vid)
        case 'videoinput'
            src = getselectedsource(vid);
            % get packet size (depending on MATLAB release PacketSize is int32 or char)
            if isnumeric(src.PacketSize)
                packetSize = double(src.PacketSize);
            else
                packetSize = str2double(src.PacketSize);
58
            end
            try
                TickFreq = src.TimestampTickFrequency;
62
            catch e
                if strcmp(e.identifier, 'MATLAB:noSuchMethodOrField') || strcmp(e.
                    identifier,'testmeas:getset:invalidProperty')
                     error('Packet delay is not supported on the device');
```

```
TU Bibliothek, Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar

The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
end
             end
             pixelFormat = vid.videoFormat;
68
             roi = vid.ROIPosition;
             height = roi(1,4);
             width = roi(1,3);
         case 'gigecam'
 72
             packetSize = double(vid.GevSCPSPacketSize);
             TickFreq = double(vid.GevTimestampTickFrequency);
 74
             pixelFormat = vid.PixelFormat;
             height = double(vid.Height);
             width = double(vid.Width);
        otherwise
 78
             % unexpected case
             error('CalculatePacketDelay(vid, fps) vid must be a valid videoinput or
                 gigecam object');
80
    end
81
82
    switch (pixelFormat)
83
         case {'Mono8' ,'BayerGR8','BayerRG8','BayerGB8','BayerBG8'}
84
             BytesPerPixel = 1;
         case {'Mono10Packed' ,'YUV411Packed'}
85
86
             BytesPerPixel = 1.5;
87
         case {'Mono10', 'Mono12', 'Mono14' , 'Mono16', 'BayerGR10', 'BayerG10', 'BayerGB10', '
            BayerBG10', 'BayerGR12', 'BayerRG12', 'BayerGB12', 'BayerBG12', 'BayerGR16', '
            BayerRG16', 'BayerGB16', 'BayerBG16', 'YUV422Packed', ''}
88
             BytesPerPixel =2;
         case {'RGB8Packed', 'BGR8Packed', 'YUV444Packed', 'RGB8Planar',}
90
             BytesPerPixel=3;
         case {'RGBA8Packed', 'BGRA8Packed'}
             BytesPerPixel = 4;
         case {'RGB10Packed', 'BGR10Packed', 'RGB12Packed', 'BGR12Packed', 'RGB10Planar', '
            RGB12Planar', 'RGB16Planar'}
             BytesPerPixel = 6;
    end
96
98
    % calculate packet delay
99
    fprintf('Calculating packet delay for:\n frame rate = %d fps,\tPacketSize (
        GevSCPSPacketSize) = %d,\tframe height = %d,\tframe width = %d,\tpixel format =
        %s, \n',...
100
         fps, packetSize, height, width, pixelFormat);
    fprintf(1, 'camera time stamp tick frequency (ticks/s) = %.1f \n', TickFreq);
102
    numOfBytes_EthernetHeader = 14;
104
    numOfBytes_IPHeader = 20;
    numOfBytes_UDPHeader = 8;
106
    numOfBytes_GVSPHeader = 8;
107
    numOfBytes_EthernetFooter = 2;
```

```
TU Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
VIEN vur knowledge hub
The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
108
    numOfBytes_overheadPerPacket = numOfBytes_EthernetHeader+numOfBytes_IPHeader+
        numOfBytes_UDPHeader+numOfBytes_GVSPHeader;
    numOfBytes_GVSP_Leader = numOfBytes_overheadPerPacket+36;
    numOfBytes_GVSP_Trailer = numOfBytes_overheadPerPacket+numOfBytes_EthernetFooter+
        numOfBytes_GVSPHeader; %Ethernet Footer+GVSP Header
112
    actual_packetSize = packetSize-numOfBytes_overheadPerPacket;
114
    numOfBytes_perFrame = height * width * BytesPerPixel;
    numOfPackets_perFrame = ceil(numOfBytes_perFrame/actual_packetSize)+2 ;
118
119
    total_OverHead_Bytes = ceil(numOfPackets_perFrame) * numOfBytes_overheadPerPacket;
121
    full_packets = floor(numOfPackets_perFrame ) * actual_packetSize;
    numOfBytes_perImage = numOfBytes_GVSP_Leader+(floor(numOfPackets_perFrame ) *
        actual_packetSize)+(numOfBytes_perFrame-(full_packets))+total_OverHead_Bytes +
        numOfBytes_GVSP_Trailer;
    numOfBytes_perSec = fps * numOfBytes_perImage;
    bits_perSecond = numOfBytes_perSec * 8;
128
129
    ratio_GigeUtilized= bits_perSecond/10^9;
    ratio_GigeNotUtilized = 1 - ratio_GigeUtilized;
    pauseTime_between_Packets = ratio_GigeNotUtilized/(numOfPackets_perFrame * fps);
    PacketDelay = TickFreq * pauseTime_between_Packets;
    delay = round(0.9 * PacketDelay);
138
    fprintf(1, '\nUsed gigabit bandwitdh: %.1f %%\n', 100 * ratio_GigeUtilized);
    fprintf(1, 'PacketDelay (GevSCPD): %d (ticks)\n', delay);
142
    end
```

7.30: Class lbcUI

1	<pre>classdef lbcUI < matlab.apps.AppE</pre>	Base
2	<pre>% Properties that correspond</pre>	to app components
3	<pre>properties (Access = public)</pre>	
4	UIFigure	matlab.ui.Figure
5	TabGroup	matlab.ui.container.TabGroup
6	HardwareSetupTab	matlab.ui.container.Tab
7	HSGridLayout	matlab.ui.container.GridLayout
8	%% define Hardware Setup	Configuration Properties

9	HSCamConfPanel matlab.ui.container.Panel
10	HSCamConfGridLayout matlab.ui.container.GridLayout
11	HSCamConfLabel matlab.ui.control.Label
12	HSCamConfConnectButton matlab.ui.control.Button
13	HSCamConfDisconnectButton matlab.ui.control.Button
14	HSCamConfCamUIAxes matlab.ui.control.UIAxes
15	HSCamConfCamTestButton matlab.ui.control.Button
16	HSCamConfExposureTimeLabel matlab.ui.control.Label
17	HSCamConfExposureTimeEditField matlab.ui.control.NumericEditField
18	HSCamConfApplySettingButton matlab.ui.control.Button
19	HSCamConfCamPreviewButton matlab.ui.control.StateButton
20	HSCamConfConnectionLabel matlab.ui.control.Label
21	
22	
23	HSStepmotorConfPanel matlab.ui.container.Panel
24	HSStepmotorConfGridLayout matlab.ui.container.GridLayout
25	HSStepmotorConfLabel matlab.ui.control.Label
26	HSStepmotorConfComPortLabel matlab.ui.control.Label
27	HSStepmotorConfComPortRefreshButton matlab.ui.control.Button
28	HSStepmotorConfComPortDropDown matlab.ui.control.DropDown
29	HSStepmotorConfConnectButton matlab.ui.control.Button
30	HSStepmotorConfDisconnectButton matlab.ui.control.Button
31	HSStepmotorConfStepsPerRevolutionLabel matlab.ui.control.Label
32	HSStepmotorConfStepsPerRevolutionEditField matlab.ui.control. NumericEditField
33	HSStepmotorConfRpmLabel matlab.ui.control.Label
34	HSStepmotorConfRpmEditField matlab.ui.control.NumericEditField
35	HSStepmotorConfApplySettingsButton matlab.ui.control.Button
36	HSStepmotorConfConnectionLabel matlab.ui.control.Label
37	
38	HSTestHardwarePanel matlab.ui.container.Panel
39	HSTestHardwareGridLayout matlab.ui.container.GridLayout
40	HSTestHardwareReferenceStageButton matlab.ui.control.Button
41	HSTestHardwareGoToZeroButton matlab.ui.control.Button
42	HSTestHardwareMotorStatusLabel matlab.ui.control.Label
43	HSTestHardwareGoToPositionLabel matlab.ui.control.Label
44	HSTestHardwareGoToPositionEditField matlab.ui.control.NumericEditField
45	HSTestHardwareGoToPositionButton matlab.ui.control.Button
46	HSTestHardwarePositionUIAxes matlab.ui.control.UIAxes
47	HSTestHardwareCurrPosLabel matlab.ui.control.Label
48	HSTestHardwareCurrPosEditField matlab.ui.control.NumericEditField
49	
50	
51	<pre>%% define Automated Measurement Properties</pre>
52	AutoMeasTab matlab.ui.container.Tab
53	AutoMeasTabGroup matlab.ui.container.TabGroup
54	AutoMeasIS0111146Part1Tab matlab.ui.container.Tab
55	AutoMeasISO111146Part1TabGridLayout matlab.ui.container.GridLayout
56	

57	AutoMeasIS0111146DistanceHolderLabel matlab.ui.control.Label				
58	AutoMeasIS0111146Part1MethodRBGroup matlab.ui.container.ButtonGroup				
59	AutoMeasIS0111146Part1FromCenterRB matlab.ui.control.RadioButton				
60	AutoMeasIS0111146Part1FromZeroRB matlab.ui.control.RadioButton				
61	AutoMeasIS0111146Part10ffsetLabel matlab.ui.control.Label				
62	AutoMeasIS0111146Part10ffseEditField matlab.ui.control.NumericEditField				
63	AutoMeasIS0111146Part1RayleighLabel matlab.ui.control.Label				
64	AutoMeasIS0111146Part1RavleighEditField matlab.ui.control.NumericEditField				
65	AutoMeasIS0111146Part1TimesRayleighLabel matlab.ui.control.Label				
66	AutoMeasIS0111146Part1TimesRayleighEditField matlab.ui.control.				
	NumericEditField				
67	AutoMeasIS0111146Part1NoMeasPointsLabel matlab.ui.control.Label				
68	AutoMeasIS0111146Part1NoMeasPointsEditField matlab.ui.control.				
	NumericEditField				
69	AutoMeasIS0111146Part1MeasPerPointLabel matlab.ui.control.Label				
70	AutoMeasIS0111146Part1MeasPerPointEditField matlab.ui.control.				
	NumericEditField				
71	AutoMeasIS0111146Part1MeasAreaLabel matlab.ui.control.Label				
72	AutoMeasISO111146Part1MeasAreaEditField matlab ui control NumericEditField				
73	AutoMeasIS0111146Part1StartMeasButton matlab ui control Button				
74	AutoMeasIS0111146Part1PreviewMeasPointsUIAxes matlab.ui.control.UIAxes				
75	AutoMeasIS0111146Part1ProgressCurrActionLabel matlab.ui.control.Label				
76	AutoMeasIS0111146Part1ProgressCurrActionEditEield_matlab.ui.control.				
	FditField				
77	AutoMeasISO111146Part1ProgressCurrMeasPointLabel matlab.ui.control.				
78	AutoMeasIS0111146Part1ProgressCurrMeasPointEditEield_matlab.ui.control.				
	FditField				
79	AutoMeasIS0111146Part1ProgressCurrMeasPointMeasLabel matlab.ui.control.				
	Label				
80	AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField matlab.ui.control.				
	FditField				
81	AutoMeasIS0111146Part1ProgressSaveMeasButton matlab.ui.control.Button				
82	AutoMeasIS0111146Part1ProgressCurrPositionUTAxes_matlab.ui.control.UTAxes				
83	AutoMeasIS0111146Part1StatusLabel matlab.ui.control.Label				
84	AutoMeasIS0111146Part1CurrPosLabel matlab.ui.control.Label				
85	AutoMeasIS0111146Part1CurrPosEditEield matlab.ui.control.NumericEditEield				
86	AutoMeasCancelButton matlab.ui.control.StateButton				
87					
88	%% define Evaluate Measurement Properties				
89	EvaluateMeasurementTab matlab.ui.container.Tab				
90	EvalTabGroup matlab.ui.container.TabGroup				
91	ConfEvalTab matlab.ui.container.Tab				
92	ConfEvalTabGridLavout matlab.ui.container.GridLavout				
93	ConfEvalloadMeasButton matlab.ui.control.Button				
94	ConfEvalSmpDropDown matlab.ui.control.DropDown				
95	ConfEvalSmplabel matlab.ui.control.label				
96	ConfEvalSmpImDropDown matlab.ui.control.DropDown				
97	ConfEvalSmpImLabel matlab ui control Label				

0.0	Configuration and the second			
90	ConfevalIntegAreaFactorStruer mattab.ui.controt.Struer			
100	ConfevalintegAreaEditField matlab.ul.control.NumericEditField			
100	ConfEvalIntegAreaFactorLabel matlab.ui.control.Label			
101	ConfEvalNtFactorSlider matlab.ui.control.Slider			
102	ConfEvalNtEditField matlab.ui.control.NumericEditField			
103	ConfEvalNtFactorLabel matlab.ui.control.Label			
104	ConfEvalWavelengthSlider matlab.ui.control.Slider			
105	ConfEvalWavelengthEditField matlab.ui.control.NumericEditField			
106	ConfEvalWavelengthLabel matlab.ui.control.Label			
107	ConfEvalConvergeCriterionSlider matlab.ui.control.Slider			
108	ConfEvalConvergeCriterionEditField matlab.ui.control.NumericEditField			
109	ConfEvalConvergeCriterionLabel matlab.ui.control.Label			
110	ConfEvalResLabel matlab.ui.control.Label			
111	ConfEvalResWidthEditField matlab.ui.control.NumericEditField			
112	ConfEvalResHeightEditEield matlab.ui.control.NumericEditEield			
113	ConfEvalPixelsizeLabel matlab ui control Label			
11/	ConfEvalPixelsizeEditField matlab ui control NumericEditField			
115	ConfEvalVarnalSizeClidar matlab ui control Slidar			
116	ConfEvalKernelSizeSilder Mattab.ul.control.Silder			
110	ConfEvalKernelSizeLobel matlab.ul.Control.Numericcultrieu			
110	ConfEvalNerNetSizeLabet Mattab.ul.Control.Labet			
118	ContevalDrawRectangleButton matlab.ul.control.Button			
119	ConfevalMethodDropDown matlab.ul.control.DropDown			
120	ConfEvalMethodLabel matlab.ul.control.Label			
121	ConfEvalBackCorrMethodDropDown matlab.ui.control.DropDown			
122	ConfEvalBackCorrMethodLabel matlab.ui.control.Label			
123	ConfEvalFineBackCorrMethodDropDown matlab.ui.control.DropDown			
124	ConfEvalFineBackCorrMethodLabel matlab.ui.control.Label			
125	ConfEvalEvaluationButton matlab.ui.control.Button			
126	ConfEvalEvaluationStatusLabel matlab.ui.control.Label			
127	ConfEvalPreviewImUIAxes matlab.ui.control.UIAxes			
128	ConfEvalDrawRectButton matlab.ui.control.Button			
129	ConfEvalFineCorrValueEditField matlab.ui.control.NumericEditField			
130	ConfEvalFineCorrValueLabel matlab.ui.control.Label			
131	<pre>%% LiveMeasurementTab</pre>			
132	LiveMeasTab matlab.ui.container.Tab			
133	LiveMeasTabGridLayout matlab.ui.container.GridLayout			
134	LiveMeasIntegAreaEditField matlab.ui.control.NumericEditField			
135	LiveMeasIntegAreaFactorLabel matlab.ui.control.Label			
136	LiveMeasNtEditField matlab.ui.control.NumericEditField			
137	LiveMeasNtFactorLabel matlab.ui.control.Label			
138	LiveMeasConvergeCriterionEditField matlab.ui.control.NumericEditField			
139	LiveMeasConvergeCriterionLabel matlab.ui.control.Label			
140	LiveMeasResLabel matlab.ui.control.label			
141	LiveMeasResWidthEditEield matlab ui control NumericEditEield			
1/19	LiveMeasResHeightEditField matlab ui control NumericEditField			
1/12	liveMeacDivelsizeLahel matlah ui control Lahol			
140	LiveMeasDivelsizeEditEiold matlab.ui.control NumericEditEiold			
1/15	Liveneasrizetsizetutuitetu mattab.ut.control.NumericEditEiold			
140 176	LiveneaskernelSizelahel matlah wi control Lahel			
140				

147	LiveMeasMethodDropDown matlab.ui.control.DropDown
148	LiveMeasMethodLabel matlab.ui.control.Label
149	LiveMeasBackCorrMethodDropDown matlab.ui.control.DropDown
150	LiveMeasBackCorrMethodLabel matlab.ui.control.Label
151	LiveMeasFineBackCorrMethodDropDown matlab.ui.control.DropDown
152	LiveMeasFineBackCorrMethodLabel matlab.ui.control.Label
153	LiveMeasBackgroundMeasButton matlab.ui.control.Button
154	LiveMeasPreviewImUIAxes matlab.ui.control.UIAxes
155	LiveMeasResultImUIAxes matlab.ui.control.UIAxes
156	LiveMeasStartStopButton matlab.ui.control.StateButton
157	LiveMeasStatusLabel matlab.ui.control.Label
158	LiveMeasImData_dWx matlab.ui.control.NumericEditField
159	LiveMeasImData_dWx_Label matlab.ui.control.Label
160	LiveMeasImData_dWy matlab.ui.control.NumericEditField
161	LiveMeasImData_dWy_Label matlab.ui.control.Label
162	LiveMeasImData_phi matlab.ui.control.NumericEditField
163	LiveMeasImData_phi_Label matlab.ui.control.Label
164	LiveMeasTriggerImageButton matlab.ui.control.Button
165	LiveMeasDrawRectButton matlab.ui.control.Button
166	LiveMeasFineCorrValueEditField matlab.ui.control.NumericEditField
167	LiveMeasFineCorrValueLabel matlab.ui.control.Label
168	LiveMeasChangeColorMapButton matlab.ui.control.StateButton
169	LiveMeasColorMinEditField matlab.ui.control.NumericEditField
170	LiveMeasColorMinLabel matlab.ui.control.Label
171	LiveMeasColorMaxEditField matlab.ui.control.NumericEditField
172	LiveMeasColorMaxLabel matlab.ui.control.Label
173	LiveMeasNoOfImEditField matlab.ui.control.NumericEditField
174	LiveMeasNoUfImLabel matlab.ui.control.Label
176	% define Evalutation Result Properties
177	EvalResTah matlah ui container Tah
178	EvalResTabGridLavout matlab ui container GridLavout
179	EvalResMeasDataPanel matlab ui container Panel
180	EvalResMeasDataGridLavout matlab.ui.container.GridLavout
181	EvalResMeasDataParamsGridLavout matlab.ui.container.GridLavout
182	
183	EvalResMeasData_EvalNorm matlab.ui.control.EditField
184	EvalResMeasData_EvalNorm_Label matlab.ui.control.Label
185	EvalResMeasData_BackCorrMethod matlab.ui.control.EditField
186	<pre>EvalResMeasData_BackCorrMethod_Label matlab.ui.control.Label</pre>
187	<pre>EvalResMeasData_Lambda matlab.ui.control.NumericEditField</pre>
188	EvalResMeasData_Lambda_Label matlab.ui.control.Label
189	EvalResMeasData_ConvCrit matlab.ui.control.NumericEditField
190	<pre>EvalResMeasData_ConvCrit_Label matlab.ui.control.Label</pre>
191	EvalResMeasData_IntegAreaFactor matlab.ui.control.NumericEditField
192	EvalResMeasData_IntegAreaFactor_Label matlab.ui.control.Label
193	EvalResMeasData_Nt matlab.ui.control.NumericEditField
194	EvalResMeasData_Nt_Label matlab.ui.control.Label
195	${\sf EvalResMeasData}_{\sf KernelSizeInPercent\ matlab.ui.control.NumericEditField$

106	EvalPacMaacData KarpalSizaInDarcant Labal matlab wi control Labal
190	
108	EvalResMeasData z 0 X matlab ui control NumericEditEield
100	$EvalResNeasData = 7 \oplus X$ hat table in the result of the
200	$EvalResMeasData = d \cap X$ matlab ui control NumericEditEield
200	$EvalResNeasData d \cap X$ label matlab ui control label
201	EvalResMeasData theta X matlah ui control NumericEditEield
202	EvalResMeasData theta X Label matlab ui control Label
200	EvalResMeasData z R X matlab ui control NumericEditEield
205	EvalResMeasData z R X Label matlab.ui.control.label
206	EvalResMeasData M squared X matlab.ui.control.NumericEditField
207	EvalResMeasData M squared X Label matlab.ui.control.Label
208	
209	<pre>EvalResMeasData_z_0_Y matlab.ui.control.NumericEditField</pre>
210	<pre>EvalResMeasData_z_0_Y_Label matlab.ui.control.Label</pre>
211	<pre>EvalResMeasData_d_0_Y matlab.ui.control.NumericEditField</pre>
212	<pre>EvalResMeasData_d_0_Y_Label matlab.ui.control.Label</pre>
213	<pre>EvalResMeasData_theta_Y matlab.ui.control.NumericEditField</pre>
214	EvalResMeasData_theta_Y_Label matlab.ui.control.Label
215	<pre>EvalResMeasData_z_R_Y matlab.ui.control.NumericEditField</pre>
216	<pre>EvalResMeasData_z_R_Y_Label matlab.ui.control.Label</pre>
217	EvalResMeasData_M_squared_Y matlab.ui.control.NumericEditField
218	EvalResMeasData_M_squared_Y_Label matlab.ui.control.Label
219	EvalResMeasData_M_squared_eff matlab.ui.control.NumericEditField
220	EvalResMeasData_M_squared_eff_Label matlab.ui.control.Label
221	
222	EvalResMeasLoadButton matlab.ui.control.Button
223	EvalResMeasDataResultDropDown_Label matlab.ui.control.Label
224	EvalResMeasDataResultDropDown matlab.u1.control.DropDown
225	
220	EvalResSavemedsBullon Mattab.ul.control.Bullon
221	Eval Resexport Measure in mattab.ur.control.button
220	EvalResHoldPlotButton matlabut control StateButton
220	EvalResPlotChooserDronDown matlab ui control DronDown
231	
232	EvalResSmpDataPanel matlab.ui.container.Panel
233	EvalResSmpDataGridLayout matlab.ui.container.GridLayout
234	EvalResSmpDataParamsGridLayout matlab.ui.container.GridLayout
235	EvalResSmpData_W_X matlab.ui.control.NumericEditField
236	EvalResSmpData_W_X_Label matlab.ui.control.Label
237	EvalResSmpData_W_X_std matlab.ui.control.NumericEditField
238	<pre>EvalResSmpData_W_X_std_Label matlab.ui.control.Label</pre>
239	<pre>EvalResSmpData_W_Y matlab.ui.control.NumericEditField</pre>
240	EvalResSmpData_W_Y_Label matlab.ui.control.Label
241	<pre>EvalResSmpData_W_Y_std matlab.ui.control.NumericEditField</pre>
242	EvalResSmpData_W_Y_std_Label matlab.ui.control.Label
243	EvalResSmpData_W_X_squared matlab.ui.control.NumericEditField
244	EvalResSmpData_W_X_squared_Label matlab.ui.control.Label

	247	EvalR
	248	EvalR
	249	EvalR
	250	EvalR
	251	EvalR
	252	EvalR
	253	EvalR
	254	EvalR
	255	EvalR
	256	EvalR
ar	257	EvalR
ügb	258	EvalR
/erf	259	EvalR
ek v	260	EvalR
ioth	261	EvalR
Bibli ek.	262	EvalR
en E oth	263	EvalR
Wid 3ibli	264	EvalR
en E	265	EvalR
der Wi	266	EvalR
an TU	267	EvalR
t ist It at	268	EvalR
bei prin	269	EvalR
mai e in	270	Evalk
iplo able	$211 \\ 979$	Evol B
er D vail	212 273	Eval
iese is a	210	Evalle
on d sis	275	EvalR
the	276	EvalR
alve this	277	EvalR
gina of t	278	EvalR
Ori	279	EvalR
skte vers	280	EvalR
druc	281	EvalR
geo	282	EvalR
erte ed o	283	EvalR
obie	284	EvalR
uppr	285	EvalR
ie a he a	286	EvalR
	287	EvalR
- S	288	EvalR
Je	289	EvalR
e hut	290	EvalR
viedg	291	EvalR
P A	292	EvalR
M §	293	EvalR

EvalResSmpData_W_X_squared_std matlab.ui.control.NumericEditField EvalResSmpData_W_X_squared_std_Label matlab.ui.control.Label esSmpData_W_Y_squared matlab.ui.control.NumericEditField esSmpData_W_Y_squared_Label matlab.ui.control.Label esSmpData_W_Y_squared_std matlab.ui.control.NumericEditField esSmpData_W_Y_squared_std_Label matlab.ui.control.Label esSmpData_W_XY matlab.ui.control.NumericEditField esSmpData_W_XY_Label matlab.ui.control.Label esSmpData_W_XY_std matlab.ui.control.NumericEditField esSmpData_W_XY_std_Label matlab.ui.control.Label esSmpData_dWx matlab.ui.control.NumericEditField esSmpData_dWx_Label matlab.ui.control.Label esSmpData_dWx_std matlab.ui.control.NumericEditField esSmpData_dWx_std_Label matlab.ui.control.Label esSmpData_dWy matlab.ui.control.NumericEditField esSmpData_dWy_Label matlab.ui.control.Label esSmpData_dWy_std matlab.ui.control.NumericEditField esSmpData_dWy_std_Label matlab.ui.control.Label esSmpData_phi matlab.ui.control.NumericEditField esSmpData_phi_Label matlab.ui.control.Label esSmpData_phi_std matlab.ui.control.NumericEditField esSmpData_phi_std_Label matlab.ui.control.Label esSmpDataDropDown_Label matlab.ui.control.Label esSmpDataDropDown matlab.ui.control.DropDown esSmpDataUIAxes matlab.ui.control.UIAxes esSmpInfoTextArea matlab.ui.control.TextArea

esImDataPanel matlab.ui.container.Panel esImDataGridLayout matlab.ui.container.GridLayout esImDataParamsGridLayout matlab.ui.container.GridLayout esImData_W_X matlab.ui.control.NumericEditField esImData_W_X_Label matlab.ui.control.Label esImData_W_Y matlab.ui.control.NumericEditField esImData_W_Y_Label matlab.ui.control.Label esImData_W_X_squared matlab.ui.control.NumericEditField; esImData_W_X_squared_Label matlab.ui.control.Label esImData_W_Y_squared matlab.ui.control.NumericEditField; esImData_W_Y_squared_Label matlab.ui.control.Label esImData_W_XY matlab.ui.control.NumericEditField esImData_W_XY_Label matlab.ui.control.Label esImData_dWx matlab.ui.control.NumericEditField esImData_dWx_Label matlab.ui.control.Label esImData_dWy matlab.ui.control.NumericEditField esImData_dWy_Label matlab.ui.control.Label esImData_phi matlab.ui.control.NumericEditField esImData_phi_Label matlab.ui.control.Label esImDataDropDown_Label matlab.ui.control.Label esImDataDropDown matlab.ui.control.DropDown esImDataUIAxes matlab.ui.control.UIAxes

294	EvalResImDataColorGridLayout matlab.ui.container.GridLayout
295	EvalResImDataChangeColorMapButton matlab.ui.control.StateButton
296	EvalResColorMinEditField matlab.ui.control.NumericEditField
297	EvalResColorMinLabel matlab.ui.control.Label
298	EvalResColorMaxEditField matlab.ui.control.NumericEditField
299	EvalResColorMaxLabel matlab.ui.control.Label
300	<pre>%% define additional necessery Properties and variables needed for GUI</pre>
301	% Rectangle Object
302	ConfEvalRectRoi;
303	% Point Object
304	ConfEvalPointRoi;
305	% Rectangle Object
306	LiveMeasRectRoi;
307	% Point Object
308	LiveMeasPointRoi;
309	LoadedMeas Measurement;
310	
311 210	LiveMeasSelectedArea
012 010	Tontsizel4 double = 14;
010 914	%Arduino Stephiotor
$\frac{014}{215}$	arduinolNOShiold
316	arduinolNOShieldStepmotor:
317	currPos double:
318	currPosBoi:
310	currPosRoiAutoMeas.
320	maxValueStage:
321	StepInkrement double:
322	measurementPoints:
323	previewMarkers;
324	centerMarker;
325	endPointMarkers;
326	evaluationInProgress logical;
327	LiveMeas Measurement;
328	hLinkCurrPos;
329	%Camera class videoinput
330	vid
331	LiveMeasColorMap;
332	EvalResColorMap;
333	end
334	% Component initialization
335	<pre>methods (Access = private)</pre>
336	
337	% Create UIFigure and components
338	<pre>function createComponents(app)</pre>
339	<pre>backGroundColorLabel = [0.75 0.75];</pre>
340	app.LiveMeasColorMap = [0 255];
341	app.EvalKestolorMap = [0 255];
542	%% create ourigure and nide until all components are created

343	<pre>app.UIFigure = uifigure('Visible', 'off');</pre>
344	<pre>%% app.UIFigure.Position = [size(1) size(2) size(3) size(4)];</pre>
345	<pre>app.UIFigure.Name = 'lbc App';</pre>
346	<pre>app.UIFigure.WindowState = 'maximized';</pre>
347	<pre>app.UIFigure.Visible = 'on';</pre>
348	<pre>%% app.UIFigure.Resize = 'off';</pre>
349	<pre>app.UIFigure.AutoResizeChildren = 'on';</pre>
350	<pre>% Setting Stage Variables</pre>
351	<pre>app.StepInkrement = 1.8;</pre>
352	app.maxValueStage = 23400;
353	app.currPos = [0 0];
354	
355	%% Create TabGroup
356	<pre>app.TabGroup = uitabgroup(app.UIFigure);</pre>
357	app.TabGroup.Units = 'normalized';
358	app.labGroup.Position = [0 0 1 1];
359	An Consta HawkiewsCatumTak
30U 261	%% Create Hardwaresetuplab
260	app.HardwareSetupTab = ultab(app.TabGroup);
004 363	app.HardwareSetupTab.fitte = Hardware Setup;
364	app.naruwaresetuprab.scrottabte – on ,
365	%% Create HSGridlayout
366	app.HSGridLavout = uigridlavout(app.HardwareSetupTab):
367	app. HSGridLayout. RowHeight = $\{ 1x \}$:
368	app.HSGridLayout.ColumnWidth = $\{ x', x', x' \}$:
369	
370	%% Create HSCamConfPanel
371	<pre>app.HSCamConfPanel = uipanel(app.HSGridLayout);</pre>
372	<pre>app.HSCamConfPanel.Layout.Row = 1;</pre>
373	<pre>app.HSCamConfPanel.Layout.Column = 1;</pre>
374	
375	<pre>%% Create HSCamConfGridLayout</pre>
376	<pre>app.HSCamConfGridLayout = uigridlayout(app.HSCamConfPanel);</pre>
377	app.HSCamConfGridLayout.RowHeight = { '1x', '1x'
970	· IX', · IX', · IX' ; · IX' };
378	app.HScamconfGridLayout.columnwidth = { ix };
200	eee Croata USCamConflabal
200 281	ann HSCamConflabol - uilabol(ann HSCamConfGridLavout);
385	app.nscanconflabel = uitabet(app.nscanconforidlayout),
383	app HSCamConflabel Layout Column = $\begin{bmatrix} 1 & 3 \end{bmatrix}$:
384	app.HSCamConfLabel.Text = 'Configure Camera':
385	app.HSCamConfLabel.FontSize = app.fontsize14:
386	app.HSCamConfLabel.Interpreter = 'latex':
387	app.HSCamConfLabel.BackgroundColor = backGroundColorLabel:
388	<pre>app.HSCamConfLabel.HorizontalAlignment = 'center';</pre>
389	
390	<pre>%% Create HSCamConfConnectButton</pre>

201	
391	app.HSCamconfconnectButton = ulbutton(app.HScamconfGridLayout);
392	app.HSCamconfconnectButton.lext = 'connect';
393	app.HSCamConfConnectButton.Layout.Row = 2;
394	<pre>app.HSCamConfConnectButton.Layout.Column = 2;</pre>
395	<pre>app.HSCamConfConnectButton.HorizontalAlignment = 'center';</pre>
396	<pre>app.HSCamConfConnectButton.FontSize = app.fontsize14;</pre>
397	<pre>app.HSCamConfConnectButton.ButtonPushedFcn = @(src, event)</pre>
	HSCamConfConnectButton_ButtonPushedFcn(app, src, event);
398	<pre>%% Create HSCamConfDisconnectButton</pre>
399	<pre>app.HSCamConfDisconnectButton = uibutton(app.HSCamConfGridLayout);</pre>
400	<pre>app.HSCamConfDisconnectButton.Text = 'Disconnect';</pre>
401	<pre>app.HSCamConfDisconnectButton.Layout.Row = 2;</pre>
402	<pre>app.HSCamConfDisconnectButton.Layout.Column = 3;</pre>
403	<pre>app.HSCamConfDisconnectButton.HorizontalAlignment = 'center';</pre>
404	app.HSCamConfDisconnectButton.FontSize = app.fontsize14:
405	app.HSCamConfDisconnectButton.Enable = off' :
406	app.HSCamConfDisconnectButton.ButtonPushedFcn = $@(src. event)$
	HSCamConfDisconnectButton ButtonPushedEcn(app. src. event):
407	% Create HSCamConfConnectionLabel
408	app HSCamConfConnectionLabel = μ ilabel(app HSCamConfGridLavout);
409	app HSCamConfConnectionLabel Layout $Row = 2$:
/10	app HSCamConfConnectionLabel Layout Column = 1:
410	app.HSCamConfConnectionLabel.EayOut.cotumn = 1,
411	app. $HSComConfConnectionLabel. Interpreter = app. 101(312014),$
412	app.HSCamConfConnectionLabel PackaroundColor - [1, 0, 07, 0, 2];
410	app. HSCamConfConnectionLabel. HarizentalAlignment = [10.070.2];
414	app. HSCamConficence tionLabel. Horizontal Alignment = Center;
410	app.HSCamConfConnectionLabet.Text = "Disconnected";
410	%% Create HSCAMCONTLAMULAXes
417	app.HSCamconflamUIAXes = ulaxes(app.HScamconfldLayout);
418	app.HSCamConfCamUIAxes.Layout.Row = [5 10];
419	app.HSCamConfCamUIAxes.Layout.Column = [1 3];
420	<pre>app.HSCamConfCamUIAxes.Visible = 'off';</pre>
421	
422	%% Create HSCamContCamTestButton
423	app.HSCamConfCamTestButton = ulbutton(app.HSCamConfGridLayout);
424	<pre>app.HSCamConfCamTestButton.Text = 'Trigger Image';</pre>
425	<pre>app.HSCamConfCamTestButton.HorizontalAlignment = 'center';</pre>
426	<pre>app.HSCamConfCamTestButton.FontSize = app.fontsize14;</pre>
427	<pre>app.HSCamConfCamTestButton.Layout.Row = 4;</pre>
428	<pre>app.HSCamConfCamTestButton.Layout.Column = 1;</pre>
429	<pre>app.HSCamConfCamTestButton.ButtonPushedFcn = @(src, event)</pre>
	HSCamConfCamTestButton_ButtonPushedFcn(app, src, event);
430	<pre>app.HSCamConfCamTestButton.Enable = 'off';</pre>
431	
432	<pre>%% Create HSCamConfExposureTimeLabel</pre>
433	<pre>app.HSCamConfExposureTimeLabel = uilabel(app.HSCamConfGridLayout);</pre>
434	<pre>app.HSCamConfExposureTimeLabel.Layout.Row = 3;</pre>
435	<pre>app.HSCamConfExposureTimeLabel.Layout.Column = 1;</pre>
436	<pre>app.HSCamConfExposureTimeLabel.Text = '\$\textrm{Exposure Time in } \mu s</pre>
I	

	\$';
437	app.HSCamConfExposureTimeLabel.FontSize = app.fontsize14:
438	app.HSCamConfExposureTimeLabel.Interpreter = 'latex':
439	app.HSCamConfExposureTimeLabel.BackgroundColor = backGroundColorLabel:
440	app.HSCamConfExposureTimeLabel.HorizontalAlignment = 'center':
441	
442	%% Create HSCamConfExposureTimeEditField
443	app.HSCamConfExposureTimeEditField = uieditfield(app.HSCamConfGridLavout
	.'numeric'):
444	app.HSCamConfExposureTimeEditField.Lavout.Row = 3:
445	app.HSCamConfExposureTimeEditField.Lavout.Column = 2:
446	app.HSCamConfExposureTimeEditField.HorizontalAlignment = 'center':
447	app.HSCamConfExposureTimeEditField.FontSize = app.fontsize14;
448	<pre>app.HSCamConfExposureTimeEditField.Enable = 'off':</pre>
449	%% Create HSCamConfApplvSettingButton
450	app.HSCamConfApplySettingButton = uibutton(app.HSCamConfGridLavout):
451	app.HSCamConfApplySettingButton.Text = 'Apply';
452	app.HSCamConfApplySettingButton.HorizontalAlignment = 'center':
453	app.HSCamConfApplySettingButton.FontSize = app.fontsize14:
454	app.HSCamConfApplySettingButton.Lavout.Row = 3:
455	app.HSCamConfApplySettingButton.Layout.Column = 3;
456	app.HSCamConfApplySettingButton.ButtonPushedFcn = $@(src, event)$
	HSCamConfApplySettingButton_ButtonPushedFcn(app, src, event);
457	app.HSCamConfApplySettingButton.Enable = 'off';
458	%% Create HSCamConfCamPreviewButton
459	app.HSCamConfCamPreviewButton = uibutton(app.HSCamConfGridLayout,'state'
);
460	<pre>app.HSCamConfCamPreviewButton.Text = 'Toggle Preview';</pre>
461	<pre>app.HSCamConfCamPreviewButton.HorizontalAlignment = 'center';</pre>
462	<pre>app.HSCamConfCamPreviewButton.FontSize = app.fontsize14;</pre>
463	<pre>app.HSCamConfCamPreviewButton.Layout.Row = 4;</pre>
464	<pre>app.HSCamConfCamPreviewButton.Layout.Column = 2;</pre>
465	<pre>app.HSCamConfCamPreviewButton.ValueChangedFcn = @(src, event)</pre>
	<pre>HSCamConfCamPreviewButton_ValueChangedFcn(app, src, event);</pre>
466	<pre>app.HSCamConfCamPreviewButton.Enable = 'off';</pre>
467	
468	<pre>%% Create HSStepmotorConfPanel</pre>
469	<pre>app.HSStepmotorConfPanel = uipanel(app.HSGridLayout);</pre>
470	<pre>app.HSStepmotorConfPanel.Layout.Row = 1;</pre>
471	<pre>app.HSStepmotorConfPanel.Layout.Column = 2;</pre>
472	<pre>%% Create HSStepmotorConfGridLayout</pre>
473	<pre>app.HSStepmotorConfGridLayout = uigridlayout(app.HSStepmotorConfPanel);</pre>
474	<pre>app.HSStepmotorConfGridLayout.RowHeight = {'1x','1x','1x','1x','1x','1x'</pre>
	, '1x', '1x', '1x', '1x'};
475	<pre>app.HSStepmotorConfGridLayout.ColumnWidth = {'1x','1x','1x'};</pre>
476	<pre>%% Create HSStepmotorConfLabel</pre>
477	<pre>app.HSStepmotorConfLabel = uilabel(app.HSStepmotorConfGridLayout);</pre>
478	<pre>app.HSStepmotorConfLabel.Layout.Row = 1;</pre>
479	<pre>app.HSStepmotorConfLabel.Layout.Column = [1 3];</pre>

480	<pre>app.HSStepmotorConfLabel.Text = 'Configure Stepmotor';</pre>
481	<pre>app.HSStepmotorConfLabel.FontSize = app.fontsize14;</pre>
482	<pre>app.HSStepmotorConfLabel.Interpreter = 'latex';</pre>
483	<pre>app.HSStepmotorConfLabel.BackgroundColor = backGroundColorLabel:</pre>
484	<pre>app.HSStepmotorConfLabel.HorizontalAlignment = 'center';</pre>
485	%% Create HSStepmotorConfComPortLabel
486	app.HSStepmotorConfComPortLabel = uilabel(app.HSStepmotorConfGridLayout)
	;
487	<pre>app.HSStepmotorConfComPortLabel.Layout.Row = 2;</pre>
488	<pre>app.HSStepmotorConfComPortLabel.Layout.Column = 1;</pre>
489	<pre>app.HSStepmotorConfComPortLabel.Text = 'Select COM_Port';</pre>
490	<pre>app.HSStepmotorConfComPortLabel.FontSize = app.fontsize14;</pre>
491	<pre>app.HSStepmotorConfComPortLabel.Interpreter = 'latex';</pre>
492	<pre>app.HSStepmotorConfComPortLabel.BackgroundColor = backGroundColorLabel;</pre>
493	<pre>app.HSStepmotorConfComPortLabel.HorizontalAlignment = 'center';</pre>
494	<pre>%% Create HSStepmotorConfComPortRefreshButton</pre>
495	<pre>app.HSStepmotorConfComPortRefreshButton = uibutton(app.</pre>
	HSStepmotorConfGridLayout);
496	<pre>app.HSStepmotorConfComPortRefreshButton.Text = 'Refresh';</pre>
497	<pre>app.HSStepmotorConfComPortRefreshButton.HorizontalAlignment = 'center';</pre>
498	<pre>app.HSStepmotorConfComPortRefreshButton.FontSize = app.fontsize14;</pre>
499	<pre>app.HSStepmotorConfComPortRefreshButton.Layout.Row = 2;</pre>
500	<pre>app.HSStepmotorConfComPortRefreshButton.Layout.Column = 2;</pre>
501	<pre>app.HSStepmotorConfComPortRefreshButton.ButtonPushedFcn = @(src, event)</pre>
	HSStepmotorConfComPortRefreshButton_ButtonPushedFcn(app, src, event)
	;
502	<pre>%% Create HSStepmotorConfComPortDropDown</pre>
503	<pre>app.HSStepmotorConfComPortDropDown = uidropdown(app.</pre>
FOI	HSStepmotorConfGridLayout);
504	app.HSStepmotorConfComPortDropDown.Layout.Row = 2;
505	app.HSStepmotorConfComPortDropDown.Layout.Column = 3;
506	app.HSStepmotorConfComPortDropDown.Items = {};
507	app.HSStepmotorContComPortDropDown.FontSize = app.tontsize14;
508	%% Create HSStepmotorContConnectButton
509	app.HSStepmotorContConnectButton = ulbutton(app.
F10	HSStepmotorConfGridLayout);
01U 511	app.HSStepmotorConfConnectButton.Text = Connect;
011 E10	app.HSStepmotorConfConnectButton.HorizontalAlignment = 'Center';
01Z E19	app. HSStepmotorConfConnectButton_FontSize = app. TontSize14;
010 514	app.HSStepmotorConfConnectButton_Layout.Row = 3;
014 515	$app.BSStepmotorConfConnectButton_ButtonButtonCotumn = 2;$
010	$app.nsstepmotorConfConnectButtonButtonButtonFushedEcn(app_src_ovent),$
516	** Create HSStepmeterConfDisconnectButton
517	ann HSStenmotorConfDisconnectButton $-$ uibutton(ann
017	HestopmotorConfigridLayout):
518	ann HSStenmotorConfDisconnectButton Text - 'Disconnect'
$518 \\ 510$	<pre>app.HSStepmotorConfDisconnectButton.Text = 'Disconnect'; app HSStepmotorConfDisconnectButton Layout Row = 3;</pre>
518 519 520	<pre>app.HSStepmotorConfDisconnectButton.Text = 'Disconnect'; app.HSStepmotorConfDisconnectButton.Layout.Row = 3; app.HSStepmotorConfDisconnectButton.Layout Column = 3;</pre>

app.HSSTepmotorContDisconnectButton.FontSize = app.fonts	sizel4;
<pre>523 app.HSStepmotorConfDisconnectButton.Enable = 'off';</pre>	
524 app.HSStepmotorConfDisconnectButton.ButtonPushedFcn = @	(src, event)
${\sf HSStepmotorConfDisconnectButton_ButtonPushedFcn(app,$	<pre>src, event);</pre>
525 %% Create HSStepmotorConfConnectionLabel	
<pre>526 app.HSStepmotorConfConnectionLabel = uilabel(app.</pre>	
<pre>HSStepmotorConfGridLayout);</pre>	
<pre>527 app.HSStepmotorConfConnectionLabel.Layout.Row = 3;</pre>	
<pre>528 app.HSStepmotorConfConnectionLabel.Layout.Column = 1;</pre>	
529 app.HSStepmotorConfConnectionLabel.FontSize = app.fontsi	ize14;
530 app.HSStepmotorConfConnectionLabel.Interpreter = 'latex'	';
531app.HSStepmotorConfConnectionLabel.BackgroundColor = [1	0.07 0.2];
532 app.HSStepmotorConfConnectionLabel.HorizontalAlignment =	= 'center';
533 app.HSStepmotorConfConnectionLabel.Text = 'Disconnected'	;
534 %% Create HSStepmotorConfStepsPerRevolutionLabel	
<pre>535 app.HSStepmotorConfStepsPerRevolutionLabel = uilabel(app</pre>	0.
HSStepmotorConfGridLayout);	
536 app.HSStepmotorConfStepsPerRevolutionLabel.Layout.Row =	4;
537 app.HSStepmotorConfStepsPerRevolutionLabel.Layout.Column	n = [1 2];
538 app.HSStepmotorConfStepsPerRevolutionLabel.Text = 'Steps	s Per Revolution'
	c
app.HSStepmotorContStepsPerRevolutionLabel.FontSize = ap	op.fontsize14;
app.HSStepmotorContStepsPerRevolutionLabel.Interpreter =	= 'latex';
b41 app.HSSTepmotorContStepsPerRevolutionLabel.BackgroundCol	lor =
DackGroundcolorLabel;	ianmont - Leonton
	Ignillent = center
, 5/3 %% Create HSStenmotorConfStensPerPevolutionEditEield	
544 ann HSStepmotorConfStepsPerRevolutionEditField = uiedit	field(ann
HSStepmotorConfGridLavout 'numeric'):	riced (app.
545 app. HSStepmotorConfStepsPerRevolutionEditEield.Lavout.Rc	w = 4:
546 app.HSStepmotorConfStepsPerRevolutionEditField.Lavout.Co	olumn = 3:
547 app.HSStepmotorConfStepsPerRevolutionEditField.Horizonta	alAlignment = '
center';	5
548 app.HSStepmotorConfStepsPerRevolutionEditField.FontSize	= app.fontsize14
;	
549 app.HSStepmotorConfStepsPerRevolutionEditField.Value = 2	200;
550 app.HSStepmotorConfStepsPerRevolutionEditField.Enable =	'off';
551 %% Create HSStepmotorConfRpmLabel	
<pre>552 app.HSStepmotorConfRpmLabel = uilabel(app.HSStepmotorCor</pre>	nfGridLayout);
<pre>553 app.HSStepmotorConfRpmLabel.Layout.Row = 5;</pre>	
<pre>554 app.HSStepmotorConfRpmLabel.Layout.Column = [1 2];</pre>	
555app.HSStepmotorConfRpmLabel.Text = 'Revolutions Per Minu	ute (RPM)';
<pre>556 app.HSStepmotorConfRpmLabel.FontSize = app.fontsize14;</pre>	
<pre>557 app.HSStepmotorConfRpmLabel.Interpreter = 'latex';</pre>	
558 app.HSStepmotorConfRpmLabel.BackgroundColor = backGround	dColorLabel;
559 app.HSStepmotorConfRpmLabel.HorizontalAlignment = 'center	er';
560 %% Create HSStepmotorConfRpmEditField	

561	ann UCCtanmatarCanfDnmEditEiald - wieditfiald/ann
106	app. HSStephnotorConTRphnEditField = dieditField(app.
5.00	HSStepmotorContGridLayout, 'numeric');
562	app.HSStepmotorContRpmEditField.Layout.Row = 5;
563	app.HSStepmotorContRpmEditField.Layout.Column = 3;
564	<pre>app.HSStepmotorConfRpmEditField.HorizontalAlignment = 'center';</pre>
565	<pre>app.HSStepmotorConfRpmEditField.FontSize = app.fontsize14;</pre>
566	<pre>app.HSStepmotorConfRpmEditField.Value = 10;</pre>
567	<pre>app.HSStepmotorConfRpmEditField.Enable = 'off';</pre>
568	<pre>%% Create HSStepmotorConfApplySettingsButton</pre>
569	<pre>app.HSStepmotorConfApplySettingsButton = uibutton(app.</pre>
	HSStepmotorConfGridLayout);
570	<pre>app.HSStepmotorConfApplySettingsButton.Text = 'Apply Settings';</pre>
571	<pre>app.HSStepmotorConfApplySettingsButton.HorizontalAlignment = 'center';</pre>
572	<pre>app.HSStepmotorConfApplySettingsButton.FontSize = app.fontsize14;</pre>
573	<pre>app.HSStepmotorConfApplySettingsButton.Layout.Row = 6;</pre>
574	<pre>app.HSStepmotorConfApplySettingsButton.Layout.Column = [2 3];</pre>
575	app.HSStepmotorConfApplySettingsButton.ButtonPushedFcn = @(src, event)
	HSStepmotorConfApplySettingsButton_ButtonPushedFcn(app, src, event);
576	<pre>app.HSStepmotorConfApplySettingsButton.Enable = 'off';</pre>
577	<pre>%% Create HSTestHardwarePanel</pre>
578	app.HSTestHardwarePanel = uipanel(app.HSGridLayout);
579	app.HSTestHardwarePanel.Layout.Row = 1;
580	app.HSTestHardwarePanel.Layout.Column = 3;
581	<pre>%% Create HSTestHardwareGridLayout</pre>
582	<pre>app.HSTestHardwareGridLayout = uigridlayout(app.HSTestHardwarePanel);</pre>
583	<pre>app.HSTestHardwareGridLayout.RowHeight = {'1x','1x','1x','1x','1x','1x','1x','1x'</pre>
	'1x','1x','1x','1x'};
584	<pre>app.HSTestHardwareGridLayout.ColumnWidth = {'1x','1x','1x'};</pre>
585	%% Create HSTestHardwareReferenceStageButton
586	<pre>app.HSTestHardwareReferenceStageButton = uibutton(app.</pre>
	HSTestHardwareGridLayout);
587	<pre>app.HSTestHardwareReferenceStageButton.Text = 'Reference Stage';</pre>
588	<pre>app.HSTestHardwareReferenceStageButton.HorizontalAlignment = 'center';</pre>
589	<pre>app.HSTestHardwareReferenceStageButton.FontSize = app.fontsize14;</pre>
590	<pre>app.HSTestHardwareReferenceStageButton.Layout.Row = 1;</pre>
591	<pre>app.HSTestHardwareReferenceStageButton.Layout.Column = [1 3];</pre>
592	app.HSTestHardwareReferenceStageButton.ButtonPushedFcn = @(src, event)
	HSTestHardwareReferenceStageButton_ButtonPushedFcn(app, src, event);
593	%% Create HSTestHardwareGoToZeroButton
594	app.HSTestHardwareGoToZeroButton = uibutton(app.HSTestHardwareGridLayout
);
595	<pre>app.HSTestHardwareGoToZeroButton.Text = 'Go to Zero';</pre>
596	<pre>app.HSTestHardwareGoToZeroButton.HorizontalAlignment = 'center';</pre>
597	<pre>app.HSIestHardwareGoToZeroButton.FontSize = app.fontsize14;</pre>
598	<pre>app.HSTestHardwareGoToZeroButton.Layout.Row = 2;</pre>
599	<pre>app.HSTestHardwareGoToZeroButton.Layout.Column = 3;</pre>
600	<pre>app.HSTestHardwareGoToZeroButton.ButtonPushedFcn = @(src, event)</pre>
0.01	HSTestHardwareGoToZeroButton_ButtonPushedFcn(app, src, event);
601	<pre>%% Create HSTestHardwareGoToPositionLabel</pre>

602	<pre>app.HSTestHardwareGoToPositionLabel = uilabel(app.</pre>
	<pre>HSTestHardwareGridLayout);</pre>
603	<pre>app.HSTestHardwareGoToPositionLabel.Layout.Row = 4;</pre>
604	<pre>app.HSTestHardwareGoToPositionLabel.Layout.Column = 1;</pre>
605	<pre>app.HSTestHardwareGoToPositionLabel.Text = 'Go To Position';</pre>
606	<pre>app.HSTestHardwareGoToPositionLabel.FontSize = app.fontsize14;</pre>
607	<pre>app.HSTestHardwareGoToPositionLabel.Interpreter = 'latex';</pre>
608	<pre>app.HSTestHardwareGoToPositionLabel.BackgroundColor =</pre>
	<pre>backGroundColorLabel;</pre>
609	<pre>app.HSTestHardwareGoToPositionLabel.HorizontalAlignment = 'center';</pre>
610	% Create HSTestHardwareGoToPositionEditField
611	<pre>app.HSTestHardwareGoToPositionEditField = uieditfield(app.</pre>
	HSTestHardwareGridLayout, 'numeric');
612	<pre>app.HSTestHardwareGoToPositionEditField.Layout.Row = 4;</pre>
613	<pre>app.HSTestHardwareGoToPositionEditField.Layout.Column = 2;</pre>
614	<pre>%% app.HSTestHardwareGoToPositionEditField.ValueChangedFcn = @(src,</pre>
	<pre>event) ConfEvalWavelengthSlider_AllEvents(app, src, event);</pre>
615	<pre>app.HSTestHardwareGoToPositionEditField.HorizontalAlignment = 'center';</pre>
616	<pre>app.HSTestHardwareGoToPositionEditField.FontSize = app.fontsize14;</pre>
617	<pre>app.HSTestHardwareGoToPositionEditField.Value = 10;</pre>
618	%% Create HSTestHardwareGoToPositionButton
619	app.HSTestHardwareGoToPositionButton = uibutton(app.
	HSTestHardwareGridLayout);
620	app.HSTestHardwareGoToPositionButton.Text = 'GO';
621	<pre>app.HSTestHardwareGoToPositionButton.HorizontalAlignment = 'center';</pre>
622	<pre>app.HSTestHardwareGoToPositionButton.FontSize = app.fontsize14;</pre>
623	app.HSTestHardwareGoToPositionButton.Layout.Row = 4;
624	app.HSTestHardwareGoToPositionButton.Layout.Column = 3;
625	app.HSTestHardwareGoToPositionButton.ButtonPushedFon = @(src, event)
COC	HSTestHardwareGoTOPOSITIOnButton_ButtonPushedFch(app, src, event);
020 697	%% Create Automeasisoiiii40PartistatusLabel
027	app.nsrestnaruwarenotorstatustabet = uitabet(app.
628	app HSTostHardwareMotorStatuclabol Layout Pow - 3:
620	app. HSTestHardwareMotorStatusLabel Layout Column - 1:
630	app.HSTestHardwareMotorStatusLabel FontSize = app fontsize14:
631	app.HSTestHardwareMotorStatusLabel Interpreter = 'latex':
632	app HSTestHardwareMotorStatusLabel BackgroundColor = $[1 \ 0 \ 07 \ 0 \ 2]$.
633	app.HSTestHardwareMotorStatusLabel.HorizontalAlignment = 'center':
634	app.HSTestHardwareMotorStatusLabel.Text = 'Motor Off':
635	% Create HSTestHardwarePositionUIAxes
636	app.HSTestHardwarePositionUIAxes = uiaxes(app.HSTestHardwareGridLavout):
637	app.HSTestHardwarePositionUIAxes.Lavout.Row = [5 7]:
638	app.HSTestHardwarePositionUIAxes.Lavout.Column = [1 3];
639	app.HSTestHardwarePositionUIAxes.XLim = [0-250 app.maxValueStage+250]:
640	app.HSTestHardwarePositionUIAxes.YLim = [-0.5 0.5];
641	<pre>app.HSTestHardwarePositionUIAxes.XAxisLocation = 'origin';</pre>
642	<pre>app.HSTestHardwarePositionUIAxes.Interactions = [];</pre>
643	<pre>app.HSTestHardwarePositionUIAxes.Toolbar.Visible = 'off';</pre>

645 app.HSTestHardwarePositionUIAxes.Title.String = 'Camera	a z — Position';
646 app.HSTestHardwarePositionUIAxes.Title.Interpreter = '1	latex';
647 xtickangle(app.HSTestHardwarePositionUIAxes,45);	
648 drawnow;	
649 %% Create currPosRoi	
650 app.currPosRoi = drawpoint(app.HSTestHardwarePositionUJ	[Axes,'Position',
<pre>app.currPos,'Color','yellow');</pre>	
<pre>651 app.currPosRoi.InteractionsAllowed = 'none';</pre>	
652 %% Create HSTestHardwareCurrPosLabel	
653 app.HSTestHardwareCurrPosLabel = uilabel(app.HSTestHard	<pre>dwareGridLayout);</pre>
654 app.HSTestHardwareCurrPosLabel.Layout.Row = 8;	
655 app.HSTestHardwareCurrPosLabel.Layout.Column = 1;	
656 app.HSTestHardwareCurrPosLabel.FontSize = app.fontsize1	L4;
657 app.HSTestHardwareCurrPosLabel.Interpreter = 'latex';	
658 app.HSTestHardwareCurrPosLabel.BackgroundColor = backGr	roundColorLabel;
659 app.HSTestHardwareCurrPosLabel.HorizontalAlignment = 'c	center';
660 app.HSTestHardwareCurrPosLabel.Text = 'Current Position	ו';
661 %% Create HSTestHardwareCurrPosEditField	
662 app.HSTestHardwareCurrPosEditField = uieditfield(app.	
<pre>HSTestHardwareGridLayout, 'numeric');</pre>	
<pre>663 app.HSTestHardwareCurrPosEditField.Layout.Row = 8;</pre>	
664 app.HSTestHardwareCurrPosEditField.Layout.Column = 2;	
665 app.HSTestHardwareCurrPosEditField.HorizontalAlignment	<pre>= 'center';</pre>
<pre>666 app.HSTestHardwareCurrPosEditField.Editable = 'off';</pre>	
667 %% Create AutomatedMeasTab	
<pre>668 app.AutoMeasTab = uitab(app.TabGroup);</pre>	
<pre>669 app.AutoMeasTab.Title = 'Automated Measurement';</pre>	
<pre>670 app.AutoMeasTab.Scrollable = 'on';</pre>	
671 %% Create AutoMeasTabGroup	
<pre>672 app.AutoMeasTabGroup = uitabgroup(app.AutoMeasTab);</pre>	
<pre>673 app.AutoMeasTabGroup.Units = 'normalized';</pre>	
<pre>674 app.AutoMeasTabGroup.Position = [0 0 1 1];</pre>	
<pre>675 app.AutoMeasTabGroup.TabLocation = 'bottom';</pre>	
676 %% Create AutoMeasIS0111146Part1Tab	
677 app.AutoMeasIS0111146Part1Tab = uitab(app.AutoMeasTabGr	<pre>roup);</pre>
678 app.AutoMeasIS0111146Part1Tab.Title = 'IS011146-1';	
<pre>679 app.AutoMeasIS0111146Part1Tab.Scrollable = 'on';</pre>	
680 %% Create AutoMeasIS0111146Part1TabGridLayout	
681 app.AutoMeasIS0111146Part1TabGridLayout = uigridlayout((app.
AutoMeasIS0111146Part1Tab);	
682 app.AutoMeasIS0111146Part1TabGridLayout.RowHeight = {']	lx','1x','1x','1x'
,'1x',3,'1x','1x','1x','1x','1x'};	
683 app.AutoMeasISO111146Part1TabGridLayout.ColumnWidth = {	['1x','1x','1x','1
x','1x','1x','1x','1x','1x','1x','1x','	','1x'};
684	
685 %% Create AutoMeasIS0111146Part1MethodRBGroup	
686 app.AutoMeasIS0111146Part1MethodRBGroup = uibuttongroup	o(app.
<pre>AutoMeasIS0111146Part1TabGridLayout);</pre>	
687	<pre>app.AutoMeasIS0111146Part1MethodRBGroup.Layout.Row = 1;</pre>
------------	--
688	<pre>app.AutoMeasIS0111146Part1MethodRBGroup.Layout.Column = [14 15];</pre>
689	<pre>app.AutoMeasIS0111146Part1MethodRBGroup.SelectionChangedFcn = @(src,</pre>
	event) AutoMeasISO111146Part1MethodRBGroup_SelectionChangedFcn(app,
	<pre>src, event);</pre>
690	<pre>%% Create AutoMeasIS0111146Part1FromCenterRB</pre>
691	<pre>app.AutoMeasIS0111146Part1FromCenterRB = uiradiobutton(app.</pre>
	<pre>AutoMeasIS0111146Part1MethodRBGroup,'Position',[10 35 200 15]);</pre>
692	<pre>app.AutoMeasIS0111146Part1FromCenterRB.Text = 'Configure around</pre>
	centerpoint';
693	<pre>%% Create AutoMeasIS0111146Part1FromZeroRB</pre>
694	<pre>app.AutoMeasIS0111146Part1FromZeroRB = uiradiobutton(app.</pre>
	<pre>AutoMeasIS0111146Part1MethodRBGroup,'Position',[10 15 200 15]);</pre>
695	<pre>app.AutoMeasIS0111146Part1FromZeroRB.Text = 'Configure from zero';</pre>
696	
697	%% Create AutoMeasIS0111146Part10ffsetLabel
698	app.AutoMeasIS0111146Part10ffsetLabel = uilabel(app.
	AutoMeasIS0111146Part1TabGridLayout);
699	<pre>app.AutoMeasIS0111146Part10ffsetLabel.Layout.Row = 1;</pre>
700	<pre>app.AutoMeasIS0111146Part10ffsetLabel.Layout.Column = [1 3];</pre>
701	<pre>app.AutoMeasIS0111146Part10ffsetLabel.FontSize = app.fontsize14;</pre>
702	<pre>app.AutoMeasIS0111146Part10ffsetLabel.Interpreter = 'latex';</pre>
703	<pre>app.AutoMeasIS0111146Part10ffsetLabel.BackgroundColor =</pre>
	<pre>backGroundColorLabel;</pre>
704	<pre>app.AutoMeasIS0111146Part10ffsetLabel.HorizontalAlignment = 'center';</pre>
705	<pre>app.AutoMeasIS0111146Part10ffsetLabel.Text = '\$\textrm{Center Offsetin }</pre>
-	\mu m\$';
706	%% Create AutoMeasISO111146Part10ffseEditField
101	app.AutoMeas150111146Part10TTSeEd1TF1eld = uled1TT1eld(app.
	Automeasisolill46PartilabGridLayout, 'numeric', 'valueDisplayFormat', '
700	%.IT');
708	app.AutoMeasISUIII146Part10ffseEditField.Layout.Row = 1;
709	app.AutoMeasISUIII140PartiOffseEditField.Layout.Column = 4;
(1U 711	app.AutoMeasISUIII140PartIOFISEEdILFIEtd.HorizontalAtignment = center;
(1 1	app.Automeasisoiiii40PartioriseEuttrietu.vatuechangeurch = @(src, event)
719	upualepreviewAxes(app, Src, event);
(14 719	app. Automeasisoiiii40PartiloiiseEuitrietu. Vatue = 5000;
713 714	%% Cleale Automeasisoiiii40ral LikayteigiLabel
114	app. Automeasisoiiii40raltinayteiyiLabet – uitabet(app.
715	AutoMeasiSolilii40FartiPavloidblabol Lavout Pov $= 2$
716	app.AutoMeasISOIIII40Fart1RayleighLabel.Layout.Now = 2,
717	app.AutoMeasISOIIII40FartIRayleighLabel.Layout.Cotumn = $[I S]$,
718	app.AutoMeasIS0III140FartIRayleighLabel.TontSIZE = app.TontSIZE14,
710	app.AutoMeasISOIIII40PartIRay[eighLabel.Interpreter = tatex,
113	app.Automeasisoiiii+0 ratitinayteiyiiLabet.bdtkytoullutotot = backGroundColorlabel.
720	ann AutoMeasISO111146Part1Rayleighlabel HorizontalAlignment - 'contor'.
791	app. AutomeasIS0111146Part1Ravleightabel. Text $- \frac{1}{2} + \frac{1}{$
ILL	in } \mu m\$'.
	τη j (mα mφ ,

722	<pre>%% Create AutoMeasIS0111146Part1RayleighEditField</pre>
723	<pre>app.AutoMeasIS0111146Part1RayleighEditField = uieditfield(app.</pre>
	<pre>AutoMeasIS0111146Part1TabGridLayout, 'numeric');</pre>
724	<pre>app.AutoMeasIS0111146Part1RayleighEditField.Layout.Row = 2;</pre>
725	<pre>app.AutoMeasIS0111146Part1RayleighEditField.Layout.Column = 4;</pre>
726	<pre>app.AutoMeasIS0111146Part1RayleighEditField.HorizontalAlignment = '</pre>
	center';
727	<pre>app.AutoMeasIS0111146Part1RayleighEditField.ValueChangedFcn = @(src,</pre>
	<pre>event) UpdatePreviewAxes(app, src, event);</pre>
728	<pre>app.AutoMeasIS0111146Part1RayleighEditField.Value = 3000;</pre>
729	<pre>%% Create AutoMeasIS0111146Part1TimesRayleighLabel</pre>
730	<pre>app.AutoMeasIS0111146Part1TimesRayleighLabel = uilabel(app.</pre>
	<pre>AutoMeasIS0111146Part1TabGridLayout);</pre>
731	<pre>app.AutoMeasIS0111146Part1TimesRayleighLabel.Layout.Row = 3;</pre>
732	<pre>app.AutoMeasIS0111146Part1TimesRayleighLabel.Layout.Column = [1 3];</pre>
733	<pre>app.AutoMeasIS0111146Part1TimesRayleighLabel.FontSize = app.fontsize14;</pre>
734	<pre>app.AutoMeasIS0111146Part1TimesRayleighLabel.Interpreter = 'latex';</pre>
735	<pre>app.AutoMeasIS0111146Part1TimesRayleighLabel.BackgroundColor =</pre>
	<pre>backGroundColorLabel;</pre>
736	app.AutoMeasISO111146Part1TimesRayleighLabel.HorizontalAlignment = '
	center';
737	<pre>app.AutoMeasIS0111146Part1TimesRayleighLabel.Text = 'Number of Rayleigh</pre>
	Lengths\$';
738	% Create AutoMeasIS0111146Part1TimesRayleighEditField
739	<pre>app.AutoMeasIS0111146Part1TimesRayleighEditField = uieditfield(app.</pre>
	<pre>AutoMeasIS0111146Part1TabGridLayout, 'numeric');</pre>
740	<pre>app.AutoMeasIS0111146Part1TimesRayleighEditField.Layout.Row = 3;</pre>
741	<pre>app.AutoMeasIS0111146Part1TimesRayleighEditField.Layout.Column = 4;</pre>
742	<pre>app.AutoMeasIS0111146Part1TimesRayleighEditField.HorizontalAlignment = '</pre>
	center';
743	<pre>app.AutoMeasIS0111146Part1TimesRayleighEditField.ValueChangedFcn = @(src</pre>
	<pre>, event) UpdatePreviewAxes(app, src, event);</pre>
744	app.AutoMeasIS0111146Part1TimesRayleighEditField.Value = 4;
745	%% Create AutoMeasISO111146Part1NoMeasPointsLabel
740	app.AutoMeasISUIII146PartINoMeasPointsLabel = uilabel(app.
D 4 D	AutoMeasISUIIII46PartITabGridLayout);
141	app.AutoMeasISUIII146PartINoMeasPointsLabel.Layout.Row = 4;
748	app.AutoMeasISUIII146PartINoMeasPointsLabel.Layout.Column = [I 3];
749	app.AutoMeasISUIII146PartINoMeasPointsLabel.FontSize = app.fontSize14;
700 751	app.AutoMeasISOIIII46PartINoMeasPointsLabel.Interpreter = latex;
16)	app.Automeasisoiiii46PartinomeasPointsLabel.Backgroundcolor =
759	DackGroundColorLabel;
752	app.AutomeasiSUIIII46PartINOMeasPointsLabel.HorizontalAlignment = '
759	Center ;
199	app.Automeds150111140PartimomedSP01ntSLaDet.Text = "Number OT
754	Medsurement Points; % Croate AutoMeasTS0111146Part1NeMeasDeinteEditEisld
755 755	<pre>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>
100	AutoMeasIS0111146Part1TabGrid avout 'numeric').
	Automeasisoiiiii40FaltilaboliuLayout, Humerit);

756	<pre>app.AutoMeasIS0111146Part1NoMeasPointsEditField.Layout.Row = 4;</pre>
757	<pre>app.AutoMeasIS0111146Part1NoMeasPointsEditField.Layout.Column = 4;</pre>
758	<pre>app.AutoMeasIS0111146Part1NoMeasPointsEditField.HorizontalAlignment = ' center';</pre>
759	<pre>app.AutoMeasIS0111146Part1NoMeasPointsEditField.ValueChangedFcn = @(src, event) UpdatePreviewAxes(app, src, event);</pre>
760	app.AutoMeasIS0111146Part1NoMeasPointsEditField.Value = 3:
761	<pre>%% Create AutoMeasIS0111146Part1MeasPerPointLabel</pre>
762	<pre>app.AutoMeasIS0111146Part1MeasPerPointLabel = uilabel(app.</pre>
	AutoMeasIS0111146Part1TabGridLayout);
763	<pre>app.AutoMeasIS0111146Part1MeasPerPointLabel.Layout.Row = 5;</pre>
764	<pre>app.AutoMeasIS0111146Part1MeasPerPointLabel.Layout.Column = [1 3];</pre>
765	<pre>app.AutoMeasIS0111146Part1MeasPerPointLabel.FontSize = app.fontsize14;</pre>
766	<pre>app.AutoMeasIS0111146Part1MeasPerPointLabel.Interpreter = 'latex';</pre>
767	<pre>app.AutoMeasIS0111146Part1MeasPerPointLabel.BackgroundColor =</pre>
	<pre>backGroundColorLabel;</pre>
768	<pre>app.AutoMeasIS0111146Part1MeasPerPointLabel.HorizontalAlignment = ' center':</pre>
769	<pre>app.AutoMeasIS0111146Part1MeasPerPointLabel.Text = 'Measurements per Point':</pre>
770	% Create AutoMeasIS0111146Part1MeasPerPointEditField
771	app.AutoMeasIS0111146Part1MeasPerPointEditField = uieditfield(app.
	AutoMeasIS0111146Part1TabGridLavout.'numeric'):
772	app.AutoMeasIS0111146Part1MeasPerPointEditField.Lavout.Row = 5:
773	app.AutoMeasIS0111146Part1MeasPerPointEditField.Lavout.Column = 4:
774	<pre>app.AutoMeasIS0111146Part1MeasPerPointEditField.HorizontalAlignment = ' center':</pre>
775	app.AutoMeasIS0111146Part1MeasPerPointEditField.Value = 5:
776	% Create AutoMeasIS0111146Part1MeasAreaLabel
777	app.AutoMeasIS0111146Part1MeasAreaLabel = uilabel(app.
	<pre>AutoMeasIS0111146Part1TabGridLayout);</pre>
778	<pre>app.AutoMeasIS0111146Part1MeasAreaLabel.Layout.Row = 1;</pre>
779	<pre>app.AutoMeasIS0111146Part1MeasAreaLabel.Layout.Column = [5 7];</pre>
780	<pre>app.AutoMeasIS0111146Part1MeasAreaLabel.FontSize = app.fontsize14;</pre>
781	<pre>app.AutoMeasIS0111146Part1MeasAreaLabel.Interpreter = 'latex';</pre>
782	<pre>app.AutoMeasIS0111146Part1MeasAreaLabel.BackgroundColor =</pre>
	<pre>backGroundColorLabel;</pre>
783	<pre>app.AutoMeasIS0111146Part1MeasAreaLabel.HorizontalAlignment = 'center';</pre>
784	<pre>app.AutoMeasIS0111146Part1MeasAreaLabel.Text = '\$\textrm{Measurement Area in } \mu m\$';</pre>
785	<pre>%% Create AutoMeasIS0111146Part1MeasAreaEditField</pre>
786	<pre>app.AutoMeasIS0111146Part1MeasAreaEditField = uieditfield(app.</pre>
	<pre>AutoMeasIS0111146Part1TabGridLayout,'numeric','ValueDisplayFormat',' %.lf');</pre>
787	<pre>app.AutoMeasIS0111146Part1MeasAreaEditField.Layout.Row = 1;</pre>
788	<pre>app.AutoMeasIS0111146Part1MeasAreaEditField.Layout.Column = 8;</pre>
789	<pre>app.AutoMeasIS0111146Part1MeasAreaEditField.HorizontalAlignment = ' center';</pre>
790	<pre>app.AutoMeasIS0111146Part1MeasAreaEditField.Value = 10000;</pre>

<pre>app.AutoMeasISO111146PartIMeasAreaEditField.ValueChangedFcn = @(src, event) UpdatePreviewAxes(app, src, event); % Create AutoMeasISO111146PartIStartMeasButton = uibutton(app. AutoMeasISO111146PartIStartMeasButton = uibutton(app. AutoMeasISO111146PartIStartMeasButton.Layout.Column = [14 15]; app.AutoMeasISO111146PartIStartMeasButton.Layout.Column = [14 15]; app.AutoMeasISO111146PartIStartMeasButton.Layout.Column = [14 15]; app.AutoMeasISO111146PartIStartMeasButton.ButtonPushedFcn = @(src, event) AutoMeasISO111146PartIStartMeasButton.ButtonPushedFcn(app, src, event); app.AutoMeasISO111146PartIStartMeasButton.Enable = 'off'; wontoMeasISO111146PartIStartMeasButton.Enable = 'off'; wontoMeasISO111146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLim = [-0:50 e3pp. maxXalueStage+256]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLim = [-0:50 e3p; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLim = [-0:50 e3p; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = '</pre>	791	<pre>app.AutoMeasIS0111146Part1MeasAreaEditField.Enable = 'off';</pre>
<pre>event) UpdatePreviewAxes(app, src, event); 703 %* Create AutoMeasISOIII146PartIStartMeasButton = ubutton(app. AutoMeasISOIII146PartIStartMeasButton = ubutton(app. AutoMeasISOIII146PartIStartMeasButton.Layout.Column = [14 15]; 707 app.AutoMeasISOIII146PartIStartMeasButton.Layout.Column = [14 15]; 708 app.AutoMeasISOIII146PartIStartMeasButton.Ext = 'Start'; 709 app.AutoMeasISOIII146PartIStartMeasButton.Ext = 'Start'; 709 app.AutoMeasISOIII146PartIStartMeasButton.Ext = 'Start'; 709 app.AutoMeasISOIII146PartIStartMeasButton.Ext = 'off'; 709 app.AutoMeasISOIII146PartIStartMeasButton.EutonPushedFcn(app, src, event); 709 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Layout.Row = [7 8]; 801 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; 803 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; 804 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Lime = [-0.5 0.5]; 805 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XisLocation = ' origin'; 807 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Interactions = []; 808 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Tite.String = ' Measuring Points'; 813 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; 813 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.String = '2 ccordinates'; 814 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.String = '2 ccordinates'; 815 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.String = '2 ccordinates'; 816 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.String = '2 ccordinates'; 817 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.String = '2 ccordinates'; 818 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.String = '2 ccordinates'; 819 app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes,'Position', app. CurrPosRoiAutoMeasIII4PartIPreviewMeasPointsUIAxes,'Position', app. CurrPosRoiAutoMeasIII4PartIPreviewMeasPointsUIAxes,'Position', app. CurrP</pre>	792	<pre>app.AutoMeasIS0111146Part1MeasAreaEditField.ValueChangedFcn = @(src,</pre>
<pre>703 %% Create AutoMeasIS011146PartIStartMeasButton 704 app.AutoMeasIS011146PartIStartMeasButton = uibutton(app. AutoMeasIS011146PartIStartMeasButton.Layout.Row = 5; 706 app.AutoMeasIS011146PartIStartMeasButton.Layout.Column = [14 15]; 707 app.AutoMeasIS011146PartIStartMeasButton.ButtonPushedFcn = @(src, event) 709 app.AutoMeasIS011146PartIStartMeasButton.ButtonPushedFcn(app, src, event); 709 app.AutoMeasIS011146PartIStartMeasButton.ButtonPushedFcn(app, src, event); 709 app.AutoMeasIS011146PartIStartMeasButton.ButtonPushedFcn(app, src, event); 709 app.AutoMeasIS011146PartIStartMeasButton.ButtonPushedFcn(app, src, event); 709 app.AutoMeasIS011146PartIStartMeasButton.Eutel = 'off'; 800 %% Create AutoMeasIS011146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; 801 app.AutoMeasIS011146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; 802 app.AutoMeasIS011146PartIPreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStape=250]; 805 app.AutoMeasIS011146PartIPreviewMeasPointsUIAxes.VLim = [-0.5 0.5]; 806 app.AutoMeasIS011146PartIPreviewMeasPointsUIAxes.Ntim = [-0.5 0.5]; 807 app.AutoMeasIS011146PartIPreviewMeasPointsUIAxes.Interactions = []; 808 app.AutoMeasIS011146PartIPreviewMeasPointsUIAxes.TateIsting = '</pre>		<pre>event) UpdatePreviewAxes(app, src, event);</pre>
<pre>794 app.AutoMeasISOIII146PartIStartMeasButton = uibutton(app. AutoMeasISOIII146PartIStartMeasButton.Layout.Row = 5; app.AutoMeasISOIII146PartIStartMeasButton.Layout.Column = [14 15]; app.AutoMeasISOIII146PartIStartMeasButton.Layout.Column = [14 15]; app.AutoMeasISOIII146PartIStartMeasButton.ButtonPushedFcn = @(src, event) AutoMeasISOIII146PartIStartMeasButton.ButtonPushedFcn = @(src, event) AutoMeasISOIII146PartIStartMeasButton.ButtonPushedFcn(app, src,</pre>	793	<pre>%% Create AutoMeasIS0111146Part1StartMeasButton</pre>
AutoMeasISOIII146PartITabGridLayout);795app.AutoMeasISOIII146PartIStartMeasButton.Layout.Row = 5;796app.AutoMeasISOIII146PartIStartMeasButton.Layout.Column = [14 15];797app.AutoMeasISOIII146PartIStartMeasButton.ButtonPushedFcn = @(src, event> AutoMeasISOIII146PartIStartMeasButton.ButtonPushedFcn = @(src, event);798app.AutoMeasISOIII146PartIStartMeasButton.Enable = 'off';800% Create AutoMeasISOIII146PartIPreviewMeasPointsUIAxes811app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Layout.Row = [7 8];801app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15];802app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Lim = [0-250 app. maxValueStage+250];803app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLim = [0.5 0.5];806app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Theractions = [];807app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Theractions = [];808app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Title.Interpreter = 'origin';809app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Title.Interpreter = 'latex';811app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Title.String = ' Measuring Points';812app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - ccoordinates';813app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - ccoordinates';814app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - ccoordinates';815% Create currPosRoi816 </th <th>794</th> <th><pre>app.AutoMeasIS0111146Part1StartMeasButton = uibutton(app.</pre></th>	794	<pre>app.AutoMeasIS0111146Part1StartMeasButton = uibutton(app.</pre>
<pre>795 app.AutoMeasISO111146PartIStartMeasButton.Layout.Row = 5; 706 app.AutoMeasISO111146PartIStartMeasButton.Layout.Column = [14 15]; 707 app.AutoMeasISO111146PartIStartMeasButton.Layout.Column = [14 15]; 708 app.AutoMeasISO111146PartIStartMeasButton_ButtonPushedFcn = @(src, event) AutoMeasISO111146PartIStartMeasButton_ButtonPushedFcn(app, src, event); 709 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; 803 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; 804 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; 805 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; 806 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; 807 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Interactions = []; 808 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Interactions = []; 809 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.Linterpreter = ' 810 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' 811 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' 812 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.String = ' 813 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' 814 titkangle(app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.String = '2 - ccordinates'; 814 xtickangle(app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YLabel.String = '2 - 815 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YLabel.String = '2 - 816 app.currPosRoiAutoMeas = drawpoint(app. 817 AutoMeasISO111146PartIPreviewMeasPointsUIAxes,'Position', app. 818 xtickangle(app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes,'Position', app. 819 app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes,'Position', app. 811 app.AutoMeasISO11114</pre>		<pre>AutoMeasIS0111146Part1TabGridLayout);</pre>
<pre>796 app.AutoMeasISOIIII46PartIStartMeasButton.Layout.Column = [14 15]; 707 app.AutoMeasISOIIII46PartIStartMeasButton.Text = 'Start'; 708 app.AutoMeasISOIIII46PartIStartMeasButton.ButtonPushedFcn = @(src, event) AutoMeasISOIIII46PartIStartMeasButton_ButtonPushedFcn(app, src, event); 709 app.AutoMeasISOIIII46PartIStartMeasButton.Enable = 'off'; 800 %% Create AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes 801 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Layout.Row = [7 8]; 803 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; 804 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; 805 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; 806 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; 807 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; 808 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Theractions = []; 809 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; 800 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; 810 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' 811 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' 812 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' 813 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' 814 xtickangle(app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' 814 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - 815 coordinates'; 814 xtickangle(app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - 816 app.currPosRoiAutoMeas = drawpoint(app. 817 AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes,'Position', app. 818 create currPosRoi 819 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes,'Position', app. 817 app.AutoMeasISOIIII46PartIPreviewMeasPointsUIAxes,'Position', app. 818 create AutoMeasISOIIII</pre>	795	<pre>app.AutoMeasIS0111146Part1StartMeasButton.Layout.Row = 5;</pre>
<pre>797 app.AutoMeasISO111146Part1StartMeasButton.Text = 'Start'; 708 app.AutoMeasISO111146Part1StartMeasButton.ButtonPushedFcn(app, src, event); 709 app.AutoMeasISO111146Part1StartMeasButton.Enable = 'off'; 800 %% Create AutoMeasISO111146Part1PreviewMeasPointSUIAxes 801 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes = uiaxes(app. 802 AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Layout.Row = [7 8]; 803 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Layout.Column = [6 15]; 804 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Layout.Column = [6 15]; 805 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Layout.Column = [6 15]; 806 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.XLim = [-0.5 0.5]; 807 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.YLim = [-0.5 0.5]; 808 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Interactions = []; 809 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Interactions = []; 809 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Title.String = ' 811 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Title.Interpreter = ' 812 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Title.Interpreter = ' 813 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.Title.String = ' 814 MeasUring Points'; 815 %% Create currPosRoi 816 app.CurrPosRoiAutoMeasISO111146Part1PreviewMeasPointSUIAxes.XLabel.String = 'z - 817 coordinates'; 818 xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.XLabel.String = 'z - 818 cored currPosRoi 816 app.CurrPosRoiAutoMeasISO111146Part1PreviewMeasPointSUIAxes.YDistion', app. 817 app.AutoMeasISO111146Part1PreviewMeasPointSUIAxes.YDistion', app. 818 create currPosRoiAutoMeas = drawpoint(app. 819 AutoMeasISO111146Part1PreviewMeasPointSUIAxes,'Position', app. 810 app.CurrPosRoiAutoMeas = drawpoint(app. 811 AutoMeasISO111146Part1PreviewMeasPointSUIAxes,'Position', app. 817 app.CurrPosRoiAutoMeas = drawpoint(app. 818 % Create AutoMeasISO111146Part1PreviewMeasPointSUIAxes,'Position', app. 819 AutoMeasISO111146Part1PreviewMeasPointSUIAxes,'Position', app.</pre>	796	<pre>app.AutoMeasIS0111146Part1StartMeasButton.Layout.Column = [14 15];</pre>
<pre>app.AutoMeasISO111146Part1StartMeasButton.ButtonPushedFcn = @(src, event) AutoMeasISO111146Part1StartMeasButton.ButtonPushedFcn(app, src, event); app.AutoMeasISO111146Part1StartMeasButton.Enable = 'off'; %% Create AutoMeasISO111146Part1PreviewMeasPointsUIAxes app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Row = [7 8]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.YLim = [0-5 0.5]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Interactions = []; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Totlar.Visible = 'off '; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes, 'Position', app. currPosRoiAutoMeasISO111146Part1PreviewMeasPointsUIAxes, 'Position', app. currPosRoiAutoMeasISO111146Part1PreviewMeasPointsUIAxes, 'Position', app. currPosRoiAutoMeasISO111146Part1PreviewMeasPointsUIAxes, 'Position', app. currPosRoiAutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMea</pre>	797	<pre>app.AutoMeasIS0111146Part1StartMeasButton.Text = 'Start';</pre>
<pre>) AutoMeasISO111146PartIStartMeasButton_ButtonPushedFcn(app, src, event); app.AutoMeasISO111146PartIStartMeasButton.Enable = 'off'; % Create AutoMeasISO111146PartIPreviewMeasPointsUIAxes app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Row = [7 8]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XAxisLocation = ' origin'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Interactions = []; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.String = '2 - coordinates'; xtickangle(app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.String = '2 - coordinates'; xtickangle(app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes, 'Position', app. currPosRoi.Position, 'Color', 'yellow'); app.CurrPosRoiAutoMeas InteractionsAllowed = 'none'; % Create AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoM</pre>	798	app.AutoMeasIS0111146Part1StartMeasButton.ButtonPushedFcn = @(src, event
<pre>event); app.AutoMeasISO111146Part1StartMeasButton.Enable = 'offf'; % Create AutoMeasISO111146Part1PreviewMeasPointsUIAxes 801 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Row = [7 8]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Column = [6 15]; 804 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Column = [6 15]; 805 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLim = [-0.5 0.5]; 806 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; 807 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XAxisLocation = ' origin'; 808 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Interactions = []; 809 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'offf '; 800 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'offf '; 810 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; 811 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; 812 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; 813 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - ccoordinates'; 814 xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - ccordinates'; 815 % Create currPosRoi 816 app.currPosRoiAutoMeas = drawpoint(app. AutoMeasISO111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoiAutoMeasISO11146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoiAutoMeasISO11146Part1CurrPosLabel app.AutoMeasISO11146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO11146Part1CurrPosLabel = uilabel(app. AutoMeasISO11146Part1CurrPosLabel = uilabel(app. AutoMeasISO11146Part1CurrPosLabel = uilabel(app. AutoMeasISO11146Part1CurrPosLabel = uilabel(app. AutoMeasISO11146Part1CurrPosLabel = uilabel(app. AutoMeasISO11146Part1CurrPosL</pre>) AutoMeasIS0111146Part1StartMeasButton_ButtonPushedFcn(app, src,
<pre>app.AutoMeasISO111146PartIStartMeasButton.Enable = 'off'; % Create AutoMeasISO111146PartIPreviewMeasPointsUIAxes app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes = uiaxes(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Row = [7 8]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XAxisLocation = ' origin'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Tuteractions = []; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; xtickangle(app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; stickangle(app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YIsUIAxes.45); % Create currPosRoi app.CurrPosRoiAutoMeas = drawpoint(app. AutoMeasISO111146PartIPreviewMeasPointsUIAxes, 'Position', app. currPosRoiAutoMeasISO111146PartIPreviewMeasPointsUIAxes, 'Position', app. currPosRoi.Position, 'Color', 'yellow'); app.CurrPosRoiAutoMeasISO111146PartICurrPosLabel = uilabel(app. AutoMeasISO111146PartICurrPosLabel = uilabel(app. AutoMeasISO111146P</pre>		event);
<pre>800 %% Create AutoMeasISO111146Part1PreviewMeasPointsUIAxes 801 app.AutoMeasISO111146Part1TabGridLayout); 802 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Row = [7 8]; 803 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Column = [6 15]; 804 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; 805 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; 806 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLim = [-0.5 0.5]; 807 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Interactions = []; 808 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Interactions = []; 809 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Interactions = []; 810 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.ToteLatel = 'off '; 811 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' 812 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.String = ' 813 map.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' 814 tatex'; 813 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = '2 - 814 cordinates'; 814 xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = '2 - 815 cordinates'; 815 %% Create currPosRoi 816 app.currPosRoiAutoMeas = drawpoint(app. 817 app.CurrPosRoiAutoMeas = drawpoint(app. 818 %% Create AutoMeasISO111146Part1PreviewMeasPointsUIAxes,'Position', app. 817 currPosRoiAutoMeas.InteractionsAllowed = 'none'; 818 %% Create AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel = uilabel(app. 811 AutoMeasISO111146Part1CurrPosLabel = uilabel(app. 812 AutoMeasISO111146Part1CurrPosLabel = uilabel(app. 813 AutoMeasISO111146Part1CurrPosLabel = uilabel(app. 814 Aut</pre>	799	<pre>app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'off';</pre>
<pre>801 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes = uiaxes(app. AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Layout.Row = [7 8]; 802 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Layout.Column = [6 15]; 804 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; 805 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; 806 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLim = [-0.5 0.5]; 807 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Interactions = []; 808 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Interactions = []; 809 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; 809 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' 811 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Title.String = ' 812 MeasUring Points'; 813 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' 814 latex'; 815 b% Create currPosRoi 816 app.currPosRoiAutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - ccoordinates'; 817 create currPosRoi 818 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - ccoordinates'; 818 map.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - ccoordinates'; 819 mp.CurrPosRoiAutoMeasIS0111146Part1PreviewMeasPointsUIAxes,'Position', app. 810 currPosRoiAutoMeasIS0111146Part1PreviewMeasPointsUIAxes,'Position', app. 811 currPosRoiAutoMeasIS0111146Part1PreviewMeasPointsUIAxes,'Position', app. 812 create AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,'Position', app. 813 create AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,'Position', app. 814 create AutoMeasIS0111146Part1CurrPosLabel = uilabel(app. 819 AutoMeasIS0111146Part1CurrPosLabel = uilabel(app. 810 AutoMeasIS0111146Part1CurrPosLabel = uilabel(app. 811 AutoMeasIS0111146Part1CurrPosLabel = uilabel(app. 812 AutoMeasIS0111146Part1CurrPosLabel = uilabel(app. 813 AutoMeasIS0111146Part1CurrPosLabel = uilabel(app. 814 AutoMeasIS0111146Part1Cur</pre>	800	% Create AutoMeasIS0111146Part1PreviewMeasPointsUIAxes
AutoMeasIS0111146PartITabGridLayout);802app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.Layout.Row = [7 8];803app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.Layout.Column = [615];804805app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250];805app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5];806app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5];807app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.Interactions = [];808app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.Toolbar.Visible = 'off';app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = '809app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = '810app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.Title.String = '811app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = '812app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.XLabel.String = 'z -813app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.XLabel.String = 'z -814xtickangle(app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes,45);815% Create currPosRoi816app.currPosRoiAutoMeas Is0111146PartIPreviewMeasPointsUIAxes,'Position',app.817app.currPosRoiAutoMeas InteractionsAllowed = 'none';818% Create AutoMeasIS0111146PartICurrPosLabel819app.currPosRoiAutoMeasIS0111146PartICurrPosLabel810app.currPosRoiAutoMeasIS0111146PartICurrPosLabel811app.currPosRoiAutoMeasIS0111146PartICurrPosLabel<	801	<pre>app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes = uiaxes(app.</pre>
<pre>802 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Row = [7 8]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Column = [6 15]; 804 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; 805 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; 806 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Interactions = []; 807 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Interactions = []; 808 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; 809 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; 811 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; 812 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; 813 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - ccordinates'; 814 xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - ccordinates'; 815 % Create currPosRoi 816 app.currPosRoiAutoMeas = drawpoint(app. AutoMeasISO111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoiAutoMeasIsO111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoiAutoMeasIsO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO1</pre>		<pre>AutoMeasIS0111146Part1TabGridLayout);</pre>
<pre>app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Layout.Column = [6 15]; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XAxisLocation = ' origin'; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; xtickangle(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; xtickangle(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,YPosition',app. currPosRoiAutoMeasIS0111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoiAutoMeasIS011146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoiAutoMeasIS011146Part1CurrPosLabel app.AutoMeasIS011146Part1CurrPosLabel = ulabel(app. AutoMeasIS011146Part1CurrPosLabel = ulabe</pre>	802	<pre>app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Layout.Row = [7 8];</pre>
<pre>15]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XAxisLocation = ' origin'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Interactions = []; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes,YPosition',app. currPosRoiAutoMeas = drawpoint(app. AutoMeasISO111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoiAutoMeas.InteractionsAllowed = 'none'; % Create AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel = ullabel(app. AutoMeasISO111146Part1CurrPosLabel = ullabel(app. AutoMeasI</pre>	803	app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Layout.Column = [6
<pre>app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLim = [0-250 app. maxValueStage+250]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XAxisLocation = ' origin'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Interactions = []; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; % Create currPosRoi app.currPosRoiAutoMeasISO111146Part1PreviewMeasPointsUIAxes, 'Position', app. currPosRoi.Position, 'Color', 'yellow'); app.CurrPosRoiAutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1CurrPosLabel = uilabel(app.</pre>		15];
<pre>maxValueStage+250); app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.YLim = [-0.5 0.5]; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XAxisLocation = ' origin'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Interactions = []; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; xtickangle(app.AutoMeasISO111146PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; app.CurrPosRoiAutoMeasISO111146PartIPreviewMeasPointsUIAxes,YDitAxes,45); % Create currPosRoi app.currPosRoiAutoMeasISO111146PartIPreviewMeasPointsUIAxes,'Position', app. currPosRoi.Position,'Color','yellow'); app.currPosRoiAutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO111146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.AutoMeasISO11146PartICurrPosLabel app.A</pre>	804	app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLim = [0-250 app.
<pre>app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.YLIm = [-0.5 0.5]; app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XAxisLocation = ' origin'; app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Interactions = []; app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.TitkLabel = []; app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; xtickangle(app.AutoMeasISOIII146PartIPreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; %Create currPosRoi app.currPosRoiAutoMeas = drawpoint(app. AutoMeasISOIII146PartIPreviewMeasPointsUIAxes,'Position', app. currPosRoi.Position,'Color','yellow'); app.currPosRoiAutoMeas.InteractionsAllowed = 'none'; %% Create AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel app.AutoMeasISOIII146PartICurrPosLabel</pre>	005	maxValueStage+250];
<pre>app.AutoMeasIS0111146PartIPreviewMeasPointsUIAxes.XAXisLocation = '</pre>	805	app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.YLim = [-0.5 0.5];
<pre>807 807 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Interactions = []; 808 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Toolbar.Visible = 'off '; 809 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; 811 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Title.String = ' Measuring Points'; 812 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; 813 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; 814 xtickangle(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,XLabel.String = 'z - coordinates'; 815 % Create currPosRoi 816 app.currPosRoiAutoMeas = drawpoint(app. AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoi.Position,'Color','yellow'); 817 app.CurrPosRoiAutoMeas.InteractionsAllowed = 'none'; 818 % Create AutoMeasIS0111146Part1CurrPosLabel app.AutoMeasIS0111146Part1CurrPosLabel 819 app.AutoMeasIS0111146Part1CurrPosLabel = ulabel(app. AutoMeasIS0111146Part1CurrPosLabel = ulabel(app. AutoMeasIS0111146PartICurrPosLabel = ulabel(app. AutoMeasIS0111146PartICurrPosLabel = ulabel(app. AutoMeasIS0111146PartIPatel = ul</pre>	806	app.AutoMeasISUIIII46PartIPreviewMeasPointsUIAxes.XAxisLocation =
<pre>app.AutoMeasISOIIII40PartIPreviewMeasPointsUIAxes.Interactions = []; app.AutoMeasISOIIII40PartIPreviewMeasPointsUIAxes.Toolbar.Visible = 'off</pre>	007	origin';
<pre>app.AutoMeasISOIIII40PartIPreviewMeasPointSUIAxes.Tootbar.VISIble = 011</pre>	807	app.AutoMeasISUIII146Part1PreviewMeasPointSUIAxes.Interactions = [];
<pre>809 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.YTickLabel = []; app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' latex'; 811 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' Measuring Points'; 812 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; 813 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; 814 xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes,45); 815 % Create currPosRoi 816 app.currPosRoiAutoMeas = drawpoint(app. AutoMeasISO111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoi.Position,'Color','yellow'); 817 app.currPosRoiAutoMeasII1146Part1CurrPosLabel = 'none'; 818 % Create AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1TabCrid! aventby</pre>	000	app.Automeasisoiiii46PartiPreviewMeasPointsoiAxes.Tootbar.Visible = 011
<pre>3053 app.AutoReasIS011146Part1PreviewMeasPointsUIAxes.Title.Laber = []; 310 app.AutoReasIS0111146Part1PreviewMeasPointsUIAxes.Title.Interpreter = ' app.AutoReasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = ' latex'; 313 app.AutoReasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; 314 xtickangle(app.AutoReasIS0111146Part1PreviewMeasPointsUIAxes,45); 315 %% Create currPosRoi 316 app.currPosRoiAutoMeas = drawpoint(app. AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,'Position', app. currPosRoi.Position,'Color','yellow'); 317 app.currPosRoiAutoMeasIS0111146Part1CurrPosLabel 319 app.AutoMeasIS0111146Part1CurrPosLabel = ulabel(app. AutoMeasIS0111146Part1TabCridLawett);</pre>	800	, ann AutoMeasISO1111146Part1PreviewMeasPointsUITAxes YTickLabel = [].
<pre>siss app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.Title.String = '</pre>	810	app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes Title Interpreter = '
<pre>811 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Title.String = '</pre>	010	latex':
Measuring Points'; 812 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = 'latex'; 813 app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z - coordinates'; 814 xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes,45); 815 %% Create currPosRoi 816 app.currPosRoiAutoMeas = drawpoint(app. AutoMeasISO111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoi.Position,'Color','yellow'); 817 app.currPosRoiAutoMeas.InteractionsAllowed = 'none'; 818 %% Create AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel = uilabel(app.	811	app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.Title.String = '
<pre>812 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = '</pre>	011	Measuring Points':
<pre>latex'; latex'; app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z -</pre>	812	app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.Interpreter = '
<pre>813 app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z -</pre>		latex';
<pre>coordinates'; 814 xtickangle(app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes,45); 815 %% Create currPosRoi 816 app.currPosRoiAutoMeas = drawpoint(app. AutoMeasISO111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoi.Position,'Color','yellow'); 817 app.currPosRoiAutoMeas.InteractionsAllowed = 'none'; 818 %% Create AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1TabCridtauout); </pre>	813	app.AutoMeasISO111146Part1PreviewMeasPointsUIAxes.XLabel.String = 'z —
<pre>814 xtickangle(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,45); 815 %% Create currPosRoi 816 app.currPosRoiAutoMeas = drawpoint(app.</pre>		coordinates';
<pre>815 %% Create currPosRoi 816 app.currPosRoiAutoMeas = drawpoint(app.</pre>	814	<pre>xtickangle(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,45);</pre>
<pre>816 app.currPosRoiAutoMeas = drawpoint(app.</pre>	815	%% Create currPosRoi
AutoMeasISO111146Part1PreviewMeasPointsUIAxes,'Position',app. currPosRoi.Position,'Color','yellow'); 817 app.currPosRoiAutoMeas.InteractionsAllowed = 'none'; 818 % Create AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1TabCridtareut);	816	app.currPosRoiAutoMeas = drawpoint(app.
<pre>currPosRoi.Position, 'Color', 'yellow'); app.currPosRoiAutoMeas.InteractionsAllowed = 'none'; %% Create AutoMeasISO111146Part1CurrPosLabel app.AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1TabCridterurt);</pre>		<pre>AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,'Position',app.</pre>
<pre>817 app.currPosRoiAutoMeas.InteractionsAllowed = 'none'; 818 % Create AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1TabCridLaugut);</pre>		<pre>currPosRoi.Position,'Color','yellow');</pre>
<pre>818 %% Create AutoMeasISO111146Part1CurrPosLabel 819 app.AutoMeasISO111146Part1CurrPosLabel = uilabel(app. AutoMeasISO111146Part1TabCridLaugut);</pre>	817	<pre>app.currPosRoiAutoMeas.InteractionsAllowed = 'none';</pre>
819 app.AutoMeasIS0111146Part1CurrPosLabel = uilabel(app.	818	<pre>%% Create AutoMeasIS0111146Part1CurrPosLabel</pre>
$\Lambda_{\rm u}$ to $\Lambda_{\rm o}$ to \Lambda_{\rm o} to \Lambda_{\rm o} to $\Lambda_{\rm o}$ to $\Lambda_{\rm $	819	app.AutoMeasIS0111146Part1CurrPosLabel = uilabel(app.
Automeasisoiiii40rartiia06r10Layout);		AutoMeasIS0111146Part1TabGridLayout);
app.AutoMeasISO111146Part1CurrPosLabel.Layout.Row = 9;	820	<pre>app.AutoMeasIS0111146Part1CurrPosLabel.Layout.Row = 9;</pre>
821 app.AutoMeasISUIII146Part1CurrPosLabel.Layout.Column = [6 8];	821	<pre>app.AutoMeasISUIIII46Part1CurrPosLabel.Layout.Column = [6 8];</pre>
app.Automeasisuiiii46PartiCurrPosLabel.FontSize = app.fontsizel4;	822	app.AutomeasiSUIIII46PartiCurrPosLabel.FontSize = app.fontsize14;
app.Automeasisoiiii4oParticurrPosLabet.interpreter = `latex`;	023	app.Automeasisoiiii46ParticurrPosLabet.interpreter = `latex`;

BacksformucoforLaber;app.AutoMeasISO111146Part1CurrPosLabel.HorizontalAlignment = 'center';app.AutoMeasISO111146Part1CurrPosLabel.Text = 'Current Position';%% Create AutoMeasISO111146Part1CurrPosEditFieldapp.AutoMeasISO111146Part1CurrPosEditFieldapp.AutoMeasISO111146Part1CurrPosEditFieldapp.AutoMeasISO111146Part1CurrPosEditFieldapp.AutoMeasISO111146Part1CurrPosEditField.Layout.Row = 9;app.AutoMeasISO111146Part1CurrPosEditField.Layout.Column = [9 10];app.AutoMeasISO111146Part1CurrPosEditField.HorizontalAlignment = 'cent';app.AutoMeasISO111146Part1CurrPosEditField.Value = app.currPos(1);app.AutoMeasISO111146Part1CurrPosEditField.Editable = 'off';%% Create AutoMeasISO111146DistanceHolderLabelapp.AutoMeasISO111146DistanceHolderLabel = uilabel(app. AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15];app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex';app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex';app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center'app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center'app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex';app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center'app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center'app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center'app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center'app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center'app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center'	
app.AutoMeasISOIII1460 artiCurrPosLabel.Text = 'Current Position';826827%% Create AutoMeasISOIII146Part1CurrPosEditField828app.AutoMeasISOIII146Part1CurrPosEditField = uieditfield(app. AutoMeasISOIII146Part1CurrPosEditField.Layout.Row = 9;820app.AutoMeasISOIII146Part1CurrPosEditField.Layout.Column = [9 10];821822app.AutoMeasISOIII146Part1CurrPosEditField.Layout.Column = [9 10];823824app.AutoMeasISOIII146Part1CurrPosEditField.HorizontalAlignment = 'cent ';825826827828829app.AutoMeasISOIII146Part1CurrPosEditField.Value = app.currPos(1);829820820821822822822823824824825826827828829829829820820820820820821822822822823824825824825825826827828829829829820820820821821822822823824825825826827828829829<	
<pre>327 %% Create AutoMeasISOILI146PartICurrPosEditField = uieditfield(app. AutoMeasISOILI146PartICurrPosEditField = uieditfield(app. AutoMeasISOILI146PartICurrPosEditField.Layout.Row = 9; app.AutoMeasISOILI146PartICurrPosEditField.Layout.Column = [9 10]; 829 app.AutoMeasISOILI146PartICurrPosEditField.Layout.Column = [9 10]; 831 app.AutoMeasISOILI146PartICurrPosEditField.Layout.Column = [9 10]; 832 app.AutoMeasISOILI146PartICurrPosEditField.HorizontalAlignment = 'cent '; 832 app.AutoMeasISOILI146PartICurrPosEditField.Value = app.currPos(1); 833 app.AutoMeasISOILI146PartICurrPosEditField.Editable = 'off'; 834 %% Create AutoMeasISOILI146DistanceHolderLabel 835 app.AutoMeasISOILI146DistanceHolderLabel = uilabel(app. AutoMeasISOILI146DistanceHolderLabel = uilabel(app. AutoMeasISOILI146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasISOILI146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasISOILI146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasISOILI146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasISOILI146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasISOILI146DistanceHolderLabel.HorizontalAlignment = 'center 844 app.AutoMeasISOILI146PartIProgressCurrActionLabel 844 app.AutoMeasISOILI146PartIProgressCurrActionLabel = uilabel(app. AutoMeasISOILI146PartIProgressCurrActionLabel = uilabel(app. AutoMeasISOILI146P</pre>	
S28app.AutoMeasISO111146Part1CurrPosEditField = uieditfield(app. AutoMeasISO111146Part1TabGridLayout, 'numeric');829app.AutoMeasISO111146Part1CurrPosEditField.Layout.Row = 9;830app.AutoMeasISO111146Part1CurrPosEditField.Layout.Column = [9 10];831app.AutoMeasISO111146Part1CurrPosEditField.Layout.Column = [9 10];832app.AutoMeasISO111146Part1CurrPosEditField.HorizontalAlignment = 'cent ';833app.AutoMeasISO111146Part1CurrPosEditField.Editable = 'off';834%% Create AutoMeasISO111146DistanceHolderLabel app.AutoMeasISO111146DistanceHolderLabel = uilabel(app. AutoMeasISO111146DistanceHolderLabel.Layout.Row = 6;837app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15];838app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15];839app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex';840app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex';841app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center842app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center841app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center842app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center844app.AutoMeasISO111146Part1ProgressCurrActionLabel844app.AutoMeasISO111146Part1ProgressCurrActionLabel	-
AutoMeasIS0111146PartITabGridLayout, 'numeric');829app.AutoMeasIS0111146PartICurrPosEditField.Layout.Row = 9;830app.AutoMeasIS0111146PartICurrPosEditField.Layout.Column = [9 10];831app.AutoMeasIS0111146PartICurrPosEditField.Value = app.currPos(1);832app.AutoMeasIS0111146PartICurrPosEditField.Value = app.currPos(1);833app.AutoMeasIS0111146PartICurrPosEditField.Editable = 'off';834%% Create AutoMeasIS0111146DistanceHolderLabel835app.AutoMeasIS0111146DistanceHolderLabel = uilabel(app. AutoMeasIS0111146DistanceHolderLabel.Layout.Column = [1 15];836app.AutoMeasIS0111146DistanceHolderLabel.Layout.Column = [1 15];837app.AutoMeasIS0111146DistanceHolderLabel.FontSize = app.fontsize14;839app.AutoMeasIS0111146DistanceHolderLabel.Interpreter = 'latex';840app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center841app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center842app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center843%% Create AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center844app.AutoMeasIS0111146PartIProgressCurrActionLabel844app.AutoMeasIS0111146PartITabGridLayout);844	
<pre>829 app.AutoMeasIS0111146Part1CurrPosEditField.Layout.Row = 9; 830 app.AutoMeasIS0111146Part1CurrPosEditField.Layout.Column = [9 10]; 831 app.AutoMeasIS0111146Part1CurrPosEditField.HorizontalAlignment = 'cent '; 832 app.AutoMeasIS0111146Part1CurrPosEditField.Value = app.currPos(1); 833 app.AutoMeasIS0111146Part1CurrPosEditField.Editable = 'off'; 834 % Create AutoMeasIS0111146DistanceHolderLabel 835 app.AutoMeasIS0111146DistanceHolderLabel = uilabel(app. AutoMeasIS0111146Part1TabGridLayout); 836 app.AutoMeasIS0111146DistanceHolderLabel.Layout.Row = 6; 837 app.AutoMeasIS0111146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasIS0111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasIS0111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasIS0111146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasIS0111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasIS0111146Part1ProgressCurrActionLabel 844 app.AutoMeasIS0111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasIS0111146Part1TabGridLayout);</pre>	
<pre>830 app.AutoMeasIS0111146Part1CurrPosEditField.Layout.Column = [9 10]; 831 app.AutoMeasIS0111146Part1CurrPosEditField.HorizontalAlignment = 'cent '; 832 app.AutoMeasIS0111146Part1CurrPosEditField.Value = app.currPos(1); 833 app.AutoMeasIS0111146Part1CurrPosEditField.Editable = 'off'; 834 %% Create AutoMeasIS0111146DistanceHolderLabel 835 app.AutoMeasIS0111146DistanceHolderLabel = uilabel(app. AutoMeasIS0111146DistanceHolderLabel.Layout.Row = 6; 837 app.AutoMeasIS0111146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasIS0111146DistanceHolderLabel.Layout.Column = [1 15]; 839 app.AutoMeasIS0111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasIS0111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasIS0111146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasIS0111146DistanceHolderLabel.Text = ''; 843 %% Create AutoMeasIS0111146Part1ProgressCurrActionLabel 844 app.AutoMeasIS0111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasIS0111146Part1TabGridLayout);</pre>	
<pre>831 app.AutoMeasISO111146Part1CurrPosEditField.HorizontalAlignment = 'cent '; 832 app.AutoMeasISO111146Part1CurrPosEditField.Value = app.currPos(1); 833 app.AutoMeasISO111146Part1CurrPosEditField.Editable = 'off'; 834 % Create AutoMeasISO111146DistanceHolderLabel 835 app.AutoMeasISO111146DistanceHolderLabel = uilabel(app. AutoMeasISO111146DistanceHolderLabel.Layout.Row = 6; 837 app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15]; 839 app.AutoMeasISO111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center 841 app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasISO111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasISO111146Part1ProgressCurrActionLabel 844 app.AutoMeasISO111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasISO111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasISO111146Part1TabGridLayout);</pre>	
<pre>'; 832 app.AutoMeasISO111146Part1CurrPosEditField.Value = app.currPos(1); 833 app.AutoMeasISO111146Part1CurrPosEditField.Editable = 'off'; 834 %% Create AutoMeasISO111146DistanceHolderLabel 835 app.AutoMeasISO111146DistanceHolderLabel = uilabel(app. AutoMeasISO111146DistanceHolderLabel.Layout.Row = 6; 837 app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasISO111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasISO111146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasISO111146DistanceHolderLabel.Text = ''; 843 %% Create AutoMeasISO111146Part1ProgressCurrActionLabel 844 app.AutoMeasISO111146Part1TabGridLayout); </pre>	r
<pre>832 app.AutoMeasISO111146Part1CurrPosEditField.Value = app.currPos(1); 833 app.AutoMeasISO111146Part1CurrPosEditField.Editable = 'off'; 834 % Create AutoMeasISO111146DistanceHolderLabel 835 app.AutoMeasISO111146DistanceHolderLabel = uilabel(app. AutoMeasISO111146Part1TabGridLayout); 836 app.AutoMeasISO111146DistanceHolderLabel.Layout.Row = 6; 837 app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15]; 839 app.AutoMeasISO111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasISO111146DistanceHolderLabel.BackgroundColor = [0.4 0.4</pre>	
<pre>833 app.AutoMeasISO111146Part1CurrPosEditField.Editable = 'off'; 834 %% Create AutoMeasISO111146DistanceHolderLabel 835 app.AutoMeasISO111146DistanceHolderLabel = uilabel(app.</pre>	
<pre>834 %% Create AutoMeasISO111146DistanceHolderLabel 835 app.AutoMeasISO111146DistanceHolderLabel = uilabel(app. AutoMeasISO111146Part1TabGridLayout); 836 app.AutoMeasISO111146DistanceHolderLabel.Layout.Row = 6; 837 app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasISO111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasISO111146DistanceHolderLabel.BackgroundColor = [0.4 0.4</pre>	
<pre>835 app.AutoMeasIS0111146DistanceHolderLabel = uilabel(app. AutoMeasIS0111146DistanceHolderLabel.Layout.Row = 6; 836 app.AutoMeasIS0111146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasIS0111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasIS0111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasIS0111146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasIS0111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasIS0111146Part1ProgressCurrActionLabel 844 app.AutoMeasIS0111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasIS0111146Part1TabGridLayout);</pre>	
AutoMeasIS0111146Part1TabGridLayout);836app.AutoMeasIS0111146DistanceHolderLabel.Layout.Row = 6;837app.AutoMeasIS0111146DistanceHolderLabel.Layout.Column = [1 15];838app.AutoMeasIS0111146DistanceHolderLabel.FontSize = app.fontsize14;839app.AutoMeasIS0111146DistanceHolderLabel.Interpreter = 'latex';840app.AutoMeasIS0111146DistanceHolderLabel.BackgroundColor = [0.4 0.40.4];app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center841app.AutoMeasIS0111146DistanceHolderLabel.Text = '';843%* Create AutoMeasIS0111146Part1ProgressCurrActionLabel844app.AutoMeasIS0111146Part1TabGridLayout);	
<pre>836 app.AutoMeasISO111146DistanceHolderLabel.Layout.Row = 6; 837 app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasISO111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasISO111146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasISO111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasISO111146Part1ProgressCurrActionLabel 844 app.AutoMeasISO111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasISO111146Part1TabGridLayout);</pre>	
<pre>837 app.AutoMeasISO111146DistanceHolderLabel.Layout.Column = [1 15]; 838 app.AutoMeasISO111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasISO111146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasISO111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasISO111146Part1ProgressCurrActionLabel 844 app.AutoMeasISO111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasISO111146Part1TabGridLayout);</pre>	
<pre>838 app.AutoMeasISO111146DistanceHolderLabel.FontSize = app.fontsize14; 839 app.AutoMeasISO111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasISO111146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasISO111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasISO111146Part1ProgressCurrActionLabel 844 app.AutoMeasISO111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasISO111146Part1TabGridLayout);</pre>	
<pre>839 app.AutoMeasIS0111146DistanceHolderLabel.Interpreter = 'latex'; 840 app.AutoMeasIS0111146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasIS0111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasIS0111146Part1ProgressCurrActionLabel 844 app.AutoMeasIS0111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasIS0111146Part1TabGridLayout);</pre>	
<pre>840 app.AutoMeasIS0111146DistanceHolderLabel.BackgroundColor = [0.4 0.4 0.4]; 841 app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasIS0111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasIS0111146Part1ProgressCurrActionLabel 844 app.AutoMeasIS0111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasIS0111146Part1TabGridLayout);</pre>	
0.4]; 841 app.AutoMeasISO111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasISO111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasISO111146Part1ProgressCurrActionLabel 844 app.AutoMeasISO111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasISO111146Part1TabGridLayout);	
841 app.AutoMeasIS0111146DistanceHolderLabel.HorizontalAlignment = 'center 842 app.AutoMeasIS0111146DistanceHolderLabel.Text = ''; 843 %% Create AutoMeasIS0111146Part1ProgressCurrActionLabel 844 app.AutoMeasIS0111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasIS0111146Part1TabGridLayout);	
842 app.AutoMeasIS0111146DistanceHolderLabel.Text = ''; 843 % Create AutoMeasIS0111146Part1ProgressCurrActionLabel 844 app.AutoMeasIS0111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasIS0111146Part1TabGridLayout); 845 AutoMeasIS0111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasIS0111146Part1TabGridLayout);	;
<pre>843 %% Create AutoMeasISO111146Part1ProgressCurrActionLabel 844 app.AutoMeasISO111146Part1ProgressCurrActionLabel = uilabel(app. AutoMeasISO111146Part1TabGridLayout);</pre>	_
844 app.AutoMeasIS0111146Part1ProgressCurrActionLabel = uitabel(app. AutoMeasIS0111146Part1TabGridLayout);	
AutoMeasISUIIII46PartIlabGridLayout);	
app.AutoMeasISOIIII40PartIProgressCurrActionLabel.Layout.Row = 7;	
app. Automeds150111140Part1ProgressCurrActionLabel. Layout. Cotumn = [1 5	,
fontsizo14:	
8/8 ann AutoMeasISO111146Part1ProgressCurrActionLabel Interpreter = 'later	
app.AutoMeasISO111146Part1ProgressCurrActionLabel.BackgroundColor =	'
backGroundColorLabel:	
850 app. AutoMeasIS0111146Part1ProgressCurrActionLabel. HorizontalAlignment	.
'center':	
851 app.AutoMeasIS0111146Part1ProgressCurrActionLabel.Text = 'Current Acti	n
852 % Create AutoMeasIS0111146Part1ProgressCurrActionEditField	_
853 app.AutoMeasIS0111146Part1ProgressCurrActionEditField = uieditfield(ap	
AutoMeasIS0111146Part1TabGridLayout);	
<pre>854 app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Layout.Row = 7;</pre>	
855 app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Layout.Column =	4
5];	
856 app.AutoMeasIS0111146Part1ProgressCurrActionEditField.	
<pre>HorizontalAlignment = 'center';</pre>	
857 app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Editable = 'off'	
858 % Create AutoMeasIS0111146Part1ProgressCurrMeasPointLabel	
<pre>859 app.AutoMeasIS0111146Part1ProgressCurrMeasPointLabel = uilabel(app.</pre>	

	<pre>AutoMeasIS0111146Part1TabGridLayout);</pre>
860	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointLabel.Layout.Row = 8;</pre>
861	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointLabel.Layout.Column = [1</pre>
	3];
862	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointLabel.FontSize = app.</pre>
	fontsize14;
863	app.AutoMeasISO111146Part1ProgressCurrMeasPointLabel.Interpreter = '
	latex';
864	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointLabel.BackgroundColor =</pre>
	<pre>backGroundColorLabel;</pre>
865	app.AutoMeasIS0111146Part1ProgressCurrMeasPointLabel.HorizontalAlignment
	= 'center';
866	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointLabel.Text = 'Current</pre>
	Measuring point';
867	% Create AutoMeasIS0111146Part1ProgressCurrMeasPointEditField
868	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField = uieditfield(</pre>
	<pre>app.AutoMeasIS0111146Part1TabGridLayout);</pre>
869	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Layout.Row = 8;</pre>
870	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Layout.Column =</pre>
	4;
871	app.AutoMeasISO111146Part1ProgressCurrMeasPointEditField.
	HorizontalAlignment = 'center';
872	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Editable = 'off</pre>
	';
873	% Create AutoMeasIS0111146Part1ProgressCurrMeasPointMeasLabel
874	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasLabel = uilabel(app.</pre>
	<pre>AutoMeasIS0111146Part1TabGridLayout);</pre>
875	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasLabel.Layout.Row = 9;</pre>
876	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasLabel.Layout.Column =</pre>
	[1 3];
877	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasLabel.FontSize = app.</pre>
0 -	fontsize14;
878	app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasLabel.Interpreter =
070	Latex';
879	app.Automeasisoiiiii46PartiProgressCurrmeasPointMeasLabel.BackgroundColor
000	= DackGroundcotorLabet;
000	app.Automeds150111140PartirrogresscurrmedsPointmedsLabet.
001	norizonidialignment = center;
001	app.Automeds150111140PartirrogresscurmedsPointmedsLabet.Text = Current
882	%% (reate AutoMeasIS011111/6Part1Progress(urrMeasPointMeasEditEield
883	app AutoMeasTSO111146Part1ProgressCurrMeasPointMeasEditField =
000	ujeditfield(app_AutoMeasISO111146Part1TabGridLayout):
884	app AutoMeasISO111146Part1ProgressCurrMeasPointMeasEditEield Lavout Row
UUT	
885	app AutoMeasISO111146Part1ProgressCurrMeasPointMeasEditField Layout
000	Column = 4:
886	app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField
000	HorizontalAlignment = 'center':
ļ	norizonea energy and the content y

TU **Bibliotheks** Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar wien vour knowedge hub The approved original version of this thesis is available in print at TU Wien Bibliothek.

887	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField.Editable =</pre>
888	% Create AutoMeasIS0111146Part1StatusLabel
889	<pre>app.AutoMeasIS0111146Part1StatusLabel = uilabel(app.</pre>
	<pre>AutoMeasIS0111146Part1TabGridLayout);</pre>
890	<pre>app.AutoMeasIS0111146Part1StatusLabel.Layout.Row = 10;</pre>
891	app.AutoMeasIS0111146Part1StatusLabel.Lavout.Column = [1 3]:
892	app.AutoMeasIS0111146Part1StatusLabel.FontSize = app.fontsize14;
893	<pre>app.AutoMeasIS0111146Part1StatusLabel.Interpreter = 'latex';</pre>
894	app.AutoMeasIS0111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]:
895	app.AutoMeasIS0111146Part1StatusLabel.HorizontalAlignment = 'center':
896	app.AutoMeasIS0111146Part1StatusLabel.Text = 'Inactive':
897	%% Create AutoMeasCancelButton
898	app.AutoMeasCancelButton = uibutton(app.
	AutoMeasIS0111146Part1TabGridLavout.'state'):
899	app.AutoMeasCancelButton.Text = 'Cancel':
900	app.AutoMeasCancelButton.HorizontalAlignment = 'center':
901	app.AutoMeasCancelButton.FontSize = app.fontsize14:
902	app.AutoMeasCancelButton.Layout.Row = 5:
903	app.AutoMeasCancelButton.Layout.Column = $[12, 13]$:
904	\approx app. AutoMeasCancelButton.ValueChangedEcn = @(src. event)
001	HSCamConfCamPreviewButton ValueChangedEcn(app. src. event):
905	
906	%% Create EvaluateMeasurementTab
907	<pre>app.EvaluateMeasurementTab = uitab(app.TabGroup);</pre>
908	<pre>app.EvaluateMeasurementTab.Title = 'Evaluate Measurement';</pre>
909 010	<pre>app.EvaluateMeasurementTab.Scrollable = 'on';</pre>
910 911	%% Create TabGroup
912	<pre>app.EvalTabGroup = uitabgroup(app.EvaluateMeasurementTab);</pre>
913	<pre>app.EvalTabGroup.Units = 'normalized';</pre>
914	<pre>app.EvalTabGroup.Position = [0 0 1 1];</pre>
915	<pre>app.EvalTabGroup.TabLocation = 'top';</pre>
916	% app.EvalTabGroup.
917	
918	%% Create ConfEvalTab
919	app.ConfEvalTab = uitab(app.EvalTabGroup);
920	<pre>app.ConfEvalTab.Title = 'Configure Evaluation';</pre>
921	%app.ConfEvalTab.BackgroundColor = [0.75 0.75 0.75];
922	<pre>app.ConfEvalTab.Scrollable = 'on';</pre>
923	<pre>%% Create ConfEvalTabGridLayout</pre>
924	app.ConfEvalTabGridLayout = uigridlayout(app.ConfEvalTab);
925	<pre>app.ConfEvalTabGridLayout.RowHeight = {'1x','1x','1x','1x','1x','1x','1x','1x'</pre>
	','1x','1x','1x','1x','1x','1x','1x','1
926	<pre>app.ConfEvalTabGridLayout.ColumnWidth = {'1x','1x','1x','1x','1x','1x','1x','1x'</pre>
0.07	1x','1x','1x'};
927	
928	%% Create ContevalLoadMeasButton
929	app.ContEvalLoadMeasButton = ulbutton(app.ContEvallabGridLayout);

930	<pre>app.ConfEvalLoadMeasButton.Text = 'Load Measurement';</pre>
931	<pre>app.ConfEvalLoadMeasButton.Layout.Row = 1;</pre>
932	<pre>app.ConfEvalLoadMeasButton.Layout.Column = [1 2];</pre>
933	<pre>app.ConfEvalLoadMeasButton.ButtonPushedFcn = @(src, event)</pre>
	<pre>ConfEvalLoadMeasButton_ButtonPushedFcn(app, src, event);</pre>
934	<pre>app.ConfEvalLoadMeasButton.FontSize = app.fontsize14;</pre>
935	
936	<pre>%% Create ConfEvalBackCorrMethodLabel</pre>
937	<pre>app.ConfEvalBackCorrMethodLabel = uilabel(app.ConfEvalTabGridLayout);</pre>
938	<pre>app.ConfEvalBackCorrMethodLabel.Text = 'Coarse Background Correction Method';</pre>
939	<pre>app.ConfEvalBackCorrMethodLabel.Layout.Row = 2;</pre>
940	<pre>app.ConfEvalBackCorrMethodLabel.Layout.Column = [1 2];</pre>
941	<pre>app.ConfEvalBackCorrMethodLabel.FontSize = app.fontsize14;</pre>
942	<pre>app.ConfEvalBackCorrMethodLabel.Interpreter = 'latex';</pre>
943	<pre>app.ConfEvalBackCorrMethodLabel.BackgroundColor = backGroundColorLabel;</pre>
944	<pre>app.ConfEvalBackCorrMethodLabel.HorizontalAlignment = 'center';</pre>
945	%% Create ConfEvalBackCorrMethodDropDown
946	app.ConfEvalBackCorrMethodDropDown = uidropdown(app.
	ConfEvalTabGridLayout);
947	<pre>app.ConfEvalBackCorrMethodDropDown.Layout.Row = 3;</pre>
948	<pre>app.ConfEvalBackCorrMethodDropDown.Layout.Column = [1 2];</pre>
949	<pre>app.ConfEvalBackCorrMethodDropDown.Items = {'Background Map Substraction</pre>
	<pre>','Average Background Substraction'};</pre>
950	<pre>% app.ConfEvalBackCorrMethodDropDown.ValueChangedFcn = @(src</pre>
	<pre>, event) ConfEvalBackCorrMethodDropDown_ValueChangedFcn(app, src,</pre>
	event);
951	<pre>app.ConfEvalBackCorrMethodDropDown.FontSize = app.fontsize14;</pre>
952	%% Create ConfEvalFineBackCorrMethodLabel
953	app.ConfEvalFineBackCorrMethodLabel = uilabel(app.ConfEvalTabGridLayout)
054	
954	<pre>app.ConfEvalFineBackCorrMethodLabel.Text = 'Fine Background Correction Method';</pre>
955	<pre>app.ConfEvalFineBackCorrMethodLabel.Layout.Row = 2;</pre>
956	<pre>app.ConfEvalFineBackCorrMethodLabel.Layout.Column = [3 4];</pre>
957	app.ConfEvalFineBackCorrMethodLabel.FontSize = app.fontsize14;
958	app.ConfEvalFineBackCorrMethodLabel.Interpreter = 'latex';
959	app.ConfEvalFineBackCorrMethodLabel.BackgroundColor =
	<pre>backGroundColorLabel;</pre>
960	<pre>app.ConfEvalFineBackCorrMethodLabel.HorizontalAlignment = 'center';</pre>
961	
962	<pre>%% Create ConfEvalFineBackCorrMethodDropDown</pre>
963	app.ConfEvalFineBackCorrMethodDropDown = uidropdown(app.
	ConfEvalTabGridLayout);
964	<pre>app.ConfEvalFineBackCorrMethodDropDown.Layout.Row = 3;</pre>
965	<pre>app.ConfEvalFineBackCorrMethodDropDown.Layout.Column = [3 4];</pre>
966	<pre>app.ConfEvalFineBackCorrMethodDropDown.Items = {'Statistical Method',' Manual Value','Off'};</pre>
967	<pre>% app.ConfEvalFineBackCorrMethodDropDown.ValueChangedFcn =</pre>

	<pre>@(src, event) ConfEvalBackCorrMethodDropDown_ValueChangedFcn(app,</pre>
0.00	<pre>src, event);</pre>
968	<pre>app.ConfEvalFineBackCorrMethodDropDown.FontSize = app.fontsize14;</pre>
969	% Create ConfEvalFineCorrValueLabel
970	<pre>app.ConfEvalFineCorrValueLabel = uilabel(app.ConfEvalTabGridLayout);</pre>
971	app.ConfEvalFineCorrValueLabel.Text = 'Fine Corr Value';
972	<pre>app.ConfEvalFineCorrValueLabel.Layout.Row = 2;</pre>
973	<pre>app.ConfEvalFineCorrValueLabel.Layout.Column = 5;</pre>
974	app.ConfEvalFineCorrValueLabel.FontSize = app.fontsize14;
975	app.ConfEvalFineCorrValueLabel.Interpreter = 'latex';
976	app.ConfEvalFineCorrValueLabel.BackgroundColor = backGroundColorLabel;
977	<pre>app.ConfEvalFineCorrValueLabel.HorizontalAlignment = 'center';</pre>
978	%% Create ContevalFineCorrvalueEditField
979	app.ConfEvalFineCorrValueEditField = uieditfield(app.
0.20	ContevallabGridLayout, 'numeric');
980	app.ConfEvalFineCorrValueEditField.Layout.Row = 3;
981	app.ConfEvalFineCorrValueEditField.Layout.Column = 5;
982	app.ConfEvalFineCorrValueEditField.Limits = [0 10];
983	<pre>% app.ContEvalNtEditField.ValueChangedFch = @(src, event)</pre>
0.9.4	ConfEvalNtFactorStider_AttEvents(app, src, event);
984 085	<pre>app.confEvalFineCorrValueEditField.HorizontalAlignment = "Center";</pre>
900	app.confevalFineCorrValueEditField.FontSize = app.fontSize14;
980	app.confevalrinecorrvatueEditrietd.vatue = 0;
901	<pre>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>
900	app. ConfEvalConvergeCriterionLabel Text $= \frac{1}{2}$
303	$(riterion) + textrm{ in } mum s'$
990	ann ConfEvalConvergeCriterionLabel Lavout Row = 4:
991	app ConfEvalConvergeCriterionLabel Layout Column = [1 2]:
992	app.ConfEvalConvergeCriterionLabel.FontSize = app.fontsize14:
993	app.ConfEvalConvergeCriterionLabel.Interpreter = 'latex':
994	app.ConfEvalConvergeCriterionLabel.BackgroundColor =
	backGroundColorLabel:
995	<pre>app.ConfEvalConvergeCriterionLabel.HorizontalAlignment = 'center';</pre>
996	%% Create ConfEvalConvergeCriterionEditField
997	<pre>app.ConfEvalConvergeCriterionEditField = uieditfield(app.</pre>
	<pre>ConfEvalTabGridLayout, 'numeric');</pre>
9 98	<pre>app.ConfEvalConvergeCriterionEditField.Layout.Row = 5;</pre>
999	<pre>app.ConfEvalConvergeCriterionEditField.Layout.Column = [1 2];</pre>
1000	<pre>app.ConfEvalConvergeCriterionEditField.Limits = [0.1 10];</pre>
1001	<pre>% app.ConfEvalConvergeCriterionEditField.ValueChangedFcn =</pre>
	<pre>@(src, event) ConfEvalConvergeCriterionSlider_AllEvents(app, src,</pre>
	event);
1002	<pre>app.ConfEvalConvergeCriterionEditField.HorizontalAlignment = 'center';</pre>
1003	<pre>app.ConfEvalConvergeCriterionEditField.FontSize = app.fontsize14;</pre>
1004	<pre>%% Create ConfEvalWavelengthLabel</pre>
1005	<pre>app.ConfEvalWavelengthLabel = uilabel(app.ConfEvalTabGridLayout);</pre>
1006	<pre>app.ConfEvalWavelengthLabel.Text = '\$\textrm{Wavelength }\lambda \textrm</pre>
	{ in } nm \$';

1007	<pre>app.ConfEvalWavelengthLabel.Layout.Row = 4;</pre>
1008	<pre>app.ConfEvalWavelengthLabel.Layout.Column = [3 4];</pre>
1009	<pre>app.ConfEvalWavelengthLabel.FontSize = app.fontsize14;</pre>
1010	<pre>app.ConfEvalWavelengthLabel.Interpreter = 'latex';</pre>
1011	<pre>app.ConfEvalWavelengthLabel.BackgroundColor = backGroundColorLabel;</pre>
1012	<pre>app.ConfEvalWavelengthLabel.HorizontalAlignment = 'center';</pre>
1013	%% Create ConfEvalWavelengthEditField
1014	<pre>app.ConfEvalWavelengthEditField = uieditfield(app.ConfEvalTabGridLayout,</pre>
	'numeric');
1015	<pre>app.ConfEvalWavelengthEditField.Layout.Row = 5;</pre>
1016	<pre>app.ConfEvalWavelengthEditField.Layout.Column = [3 4];</pre>
1017	app.ConfEvalWavelengthEditField.Limits = [300 1100];
ਸ਼ 1018	<pre>% app.ConfEvalWavelengthEditField.ValueChangedFcn = @(src,</pre>
lgb	<pre>event) ConfEvalWavelengthSlider_AllEvents(app, src, event);</pre>
	<pre>app.ConfEvalWavelengthEditField.HorizontalAlignment = 'center';</pre>
× 1020	app.ConfEvalWavelengthEditField.FontSize = app.fontsize14;
1021 -	% Create ConfEvalNtFactorLabel
<u>1022</u>	<pre>app.ConfEvalNtFactorLabel = uilabel(app.ConfEvalTabGridLayout);</pre>
E 41023	app.ConfEvalNtFactorLabel.Text = $'nT$ standard deviation multiplicator':
plio 1024	<pre>app.ConfEvalNtFactorLabel.Lavout.Row = 6:</pre>
⊇ <u> </u>	<pre>app.ConfEvalNtFactorLabel.Lavout.Column = [1 2]:</pre>
L Aier 1026	app.ConfEvalNtFactorLabel.FontSize = app.fontsize14;
p > ∟ ⊃ 1027	<pre>app.ConfEvalNtFactorLabel.Interpreter = 'latex':</pre>
H H 1028	app.ConfEvalNtFactorLabel.BackgroundColor = $backGroundColorLabel$:
it in 1029	<pre>app.ConfEvalNtFactorLabel.HorizontalAlignment = 'center':</pre>
June 1030	%% Create ConfEvalNtEditField
ق <u>ا</u> 1031	<pre>app.ConfEvalNtEditField = uieditfield(app.ConfEvalTabGridLayout, 'numeric</pre>
Dipl	');
ava 1035	<pre>app.ConfEvalNtEditField.Layout.Row = 7;</pre>
1033	<pre>app.ConfEvalNtEditField.Layout.Column = [1 2];</pre>
uo isi 1034	<pre>app.ConfEvalNtEditField.Limits = [2 4];</pre>
1035	<pre>% app.ConfEvalNtEditField.ValueChangedFcn = @(src, event)</pre>
this	<pre>ConfEvalNtFactorSlider_AllEvents(app, src, event);</pre>
ibi 1036	<pre>app.ConfEvalNtEditField.HorizontalAlignment = 'center';</pre>
Ö	<pre>app.ConfEvalNtEditField.FontSize = app.fontsize14;</pre>
- 1038 -	%% Create ConfEvalIntegAreaFactorLabel
prind 1039	app.ConfEvalIntegAreaFactorLabel = uilabel(app.ConfEvalTabGridLayout);
ibi 1040	<pre>app.ConfEvalIntegAreaFactorLabel.Text = 'Integrationrange multiplicator'</pre>
erte ed o	
a 1041	<pre>app.ConfEvalIntegAreaFactorLabel.Layout.Row = 8;</pre>
idd 1042	<pre>app.ConfEvalIntegAreaFactorLabel.Layout.Column = [1 2];</pre>
e e 1043	<pre>app.ConfEvalIntegAreaFactorLabel.FontSize = app.fontsize14;</pre>
¤⊨ ₁₀₄₄	<pre>app.ConfEvalIntegAreaFactorLabel.Interpreter = 'latex';</pre>
1 045	<pre>app.ConfEvalIntegAreaFactorLabel.BackgroundColor = backGroundColorLabel;</pre>
U 1046	app.ConfEvalIntegAreaFactorLabel.HorizontalAlignment = 'center';
5 1047	<pre>%% Create ConfEvalIntegAreaEditField</pre>
0 # 1048	app.ConfEvalIntegAreaEditField = uieditfield(app.ConfEvalTabGridLayout,'
	<pre>numeric');</pre>
1049	<pre>app.ConfEvalIntegAreaEditField.Layout.Row = 9;</pre>

1050	<pre>app.ConfEvalIntegAreaEditField.Layout.Column = [1 2];</pre>
1051	<pre>app.ConfEvalIntegAreaEditField.Limits = [1 3];</pre>
1052	<pre>app.ConfEvalIntegAreaEditField.HorizontalAlignment = 'center';</pre>
1053	<pre>app.ConfEvalIntegAreaEditField.FontSize = app.fontsize14;</pre>
1054	%% Create ConfEvalResLabel
1055	<pre>app.ConfEvalResLabel = uilabel(app.ConfEvalTabGridLayout);</pre>
1056	<pre>app.ConfEvalResLabel.Text = 'Image Resolution W x H in pixel';</pre>
1057	<pre>app.ConfEvalResLabel.Layout.Row = 12;</pre>
1058	<pre>app.ConfEvalResLabel.Layout.Column = [1 2];</pre>
1059	<pre>app.ConfEvalResLabel.FontSize = app.fontsize14;</pre>
1060	<pre>app.ConfEvalResLabel.Interpreter = 'latex';</pre>
1061	<pre>app.ConfEvalResLabel.BackgroundColor = backGroundColorLabel;</pre>
1062	<pre>app.ConfEvalResLabel.HorizontalAlignment = 'center';</pre>
1063	<pre>%% Create ConfEvalResWidthEditField</pre>
1064	<pre>app.ConfEvalResWidthEditField = uieditfield(app.ConfEvalTabGridLayout,'</pre>
	numeric');
1065	<pre>app.ConfEvalResWidthEditField.Layout.Row = 13;</pre>
<u>z</u> 1066	<pre>app.ConfEvalResWidthEditField.Layout.Column = 1;</pre>
1067	<pre>app.ConfEvalResWidthEditField.HorizontalAlignment = 'center';</pre>
1068	<pre>app.ConfEvalResWidthEditField.FontSize = app.fontsize14;</pre>
1069	<pre>app.ConfEvalResWidthEditField.Value = 1282;</pre>
1070	<pre>%% Create ConfEvalResHeightEditField</pre>
2 1071	<pre>app.ConfEvalResHeightEditField = uieditfield(app.ConfEvalTabGridLayout,'</pre>
ומו	numeric');
1072	<pre>app.ConfEvalResHeightEditField.Layout.Row = 13;</pre>
1073	<pre>app.ConfEvalResHeightEditField.Layout.Column = 2;</pre>
1074	<pre>app.ConfEvalResHeightEditField.HorizontalAlignment = 'center';</pre>
1075	<pre>app.ConfEvalResHeightEditField.FontSize = app.fontsize14;</pre>
g 1076	<pre>app.ConfEvalResHeightEditField.Value = 1026;</pre>
<u>2</u> 1077	%% Create ConfEvalPixelsizeLabel
g 1078	<pre>app.ConfEvalPixelsizeLabel = uilabel(app.ConfEvalTabGridLayout);</pre>
1079	app.ConfEvalPixelsizeLabel.Text = '\$\textrm{Pixelsize} \textrm{ in } \mu
5	M\$';
1080 D	app.confevalPixelsizeLabel.Layout.Row = 12;
	app.ConfEvalPixelsizeLabel.Layout.Column = 3;
5 1082 7 1082	app.ConfEvalPixelsizeLabel.FontSize = app.fontSize14;
1083	app.ConfevalPixelsizeLabel.Interpreter = 'latex';
1084	app.ConfEvalPixelsizeLabel.BackgroundColor = backgroundColorLabel;
2 1000	<pre>app.confevalPixelsizeLabet.nofizontatAtignment = center;</pre>
1087	<pre>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>
2 1001	app.comevacrizecuritietu - ureurrietu(app.comevacraboriulayout,
1088	app ConfEvalPixelsizeEditField Layout Row = 13:
1089	app.ConfEvalPixelsizeEditField.Lavout.Column = 3
1000	app.ConfEvalPixelsizeEditField HorizontalAlignment = 'center':
€ 1091	app.ConfEvalPixelsizeEditField.FontSize = app.fontsize14
1092	app.ConfEvalPixelsizeEditField.Value = 5.3:
1 093	%% Create ConfEvalKernelSizeLabel
¥ 1094	app.ConfEvalKernelSizeLabel = uilabel(app.ConfEvalTabGridLavout):
\$	

1095	app.ConfEvalKernelSizeLabel.Text = 'Kernelsize in \%';
1096	<pre>app.ConfEvalKernelSizeLabel.Layout.Row = 6;</pre>
1097	app.ConfEvalKernelSizeLabel.Lavout.Column = [3 4]:
1008	app. ConfEvalKernelSizeLabel FontSize - app. fontsize14;
1000	app.confEvalKernelSizeLabel.TontSize = app.TontSize14,
1100	app.comevalkernetsizeLabet.interpreter = tatex;
1100	app.confevalkernelSizeLabel.Backgroundcolor = backGroundcolorLabel;
1101	<pre>app.ConfEvalKernelSizeLabel.HorizontalAlignment = 'center';</pre>
1102	<pre>%% Create ConfEvalKernelSizeEditField</pre>
1103	<pre>app.ConfEvalKernelSizeEditField = uieditfield(app.ConfEvalTabGridLayout,</pre>
	'numeric');
1104	<pre>app.ConfEvalKernelSizeEditField.Layout.Row = 7;</pre>
1105	<pre>app.ConfEvalKernelSizeEditField.Layout.Column = [3 4];</pre>
1106	app.ConfEvalKernelSizeEditField.Limits = [2 5]:
1107	app ConfEvalKernelSizeEditEield HorizontalAlignment = 'center'
1108	approonfEvalKernelSizeEditField FontSize = ann fontsize14:
1100	es Create ConfEvalMethod abol
1110	opp Configuration and a set a
1110	app.confevalmethodLabet = ultabet(app.confevalTaberidLayout);
	app.ConfevalMethodLabel.Text = 'Evaluation Norm';
<u>5</u> 1112	app.ConfEvalMethodLabel.Layout.Row = 16;
a 1113	app.ConfEvalMethodLabel.Layout.Column = [1 2];
1114	<pre>app.ConfEvalMethodLabel.FontSize = app.fontsize14;</pre>
1 115	<pre>app.ConfEvalMethodLabel.Interpreter = 'latex';</pre>
> 1116	<pre>app.ConfEvalMethodLabel.BackgroundColor = backGroundColorLabel;</pre>
ਸ਼ੂ 1117	<pre>app.ConfEvalMethodLabel.HorizontalAlignment = 'center';</pre>
Ĕ 1118	% Create ConfEvalMethodDropDown
- 1119	app.ConfEvalMethodDropDown = uidropdown(app.ConfEvalTabGridLavout):
ש 1120	app ConfEvalMethodDropDown Layout Row = 17:
a 1120	app $ConfEvalMethodDropDown Layout Column = [1, 2];$
1121	app. ConfEvalMethodDropDown. Items $- \{ \{ I \} \}$
<u>N</u> 1192	app.confEvalMethodDropDown.items = { isoiii40 i },
$\frac{1120}{2}$	app.contevatiechodbropbowin.fontsize = app.fontsizei4,
<u>1124</u>	%% Create ContevalevaluationButton
1 120	app.confevalevaluationButton = uibutton(app.confevaliabGridLayout);
= 1126	app.ConfEvalEvaluationButton.Text = 'Evaluate';
1127	<pre>app.ConfEvalEvaluationButton.Layout.Row = 17;</pre>
<u>5</u> 1128	<pre>app.ConfEvalEvaluationButton.Layout.Column = 3;</pre>
₿ 1129	<pre>app.ConfEvalEvaluationButton.ButtonPushedFcn = @(src, event)</pre>
g	ConfEvalEvaluationButton_ButtonPushedFcn(app, src, event);
P 1130	<pre>app.ConfEvalEvaluationButton.FontSize = app.fontsize14;</pre>
n 1131	<pre>%% Create ConfEvaluationStatusLabel</pre>
Š 1132	<pre>app.ConfEvalEvaluationStatusLabel = uilabel(app.ConfEvalTabGridLayout);</pre>
a 1133	app.ConfEvalEvaluationStatusLabel.Lavout.Row = 16:
ช บ 1134	app.ConfEvalEvaluationStatusLabel.Lavout.Column = 3:
1135	app.ConfEvalEvaluationStatusLabel_FontSize = app_fontsize14
1126	approximited the transfer of
1127	approximevatevature constant about the process -1 and
110/ 9 1190	app.confevalevaluationStatusLabel.BackgroundCotor = [1 0.0/ 0.2];
2 1138 8 1190	app.contevalevaluationStatusLabel.HorizontalAlignment = 'center';
m 1139	app.contevalevaluationStatusLabel.lext = 'Standby';
6 1140	%% Create ContEvalSmpLabel
<u>א</u> 1141	<pre>app.ConfEvalSmpLabel = uilabel(app.ConfEvalTabGridLayout);</pre>
•	

1142	<pre>app.ConfEvalSmpLabel.Text = 'Measuring Point No.';</pre>
1143	<pre>app.ConfEvalSmpLabel.Layout.Row = 1;</pre>
1144	<pre>app.ConfEvalSmpLabel.Layout.Column = [7 8];</pre>
1145	<pre>app.ConfEvalSmpLabel.FontSize = app.fontsize14;</pre>
1146	<pre>app.ConfEvalSmpLabel.Interpreter = 'latex';</pre>
1147	<pre>app.ConfEvalSmpLabel.HorizontalAlignment = 'center';</pre>
1148	<pre>app.ConfEvalSmpLabel.BackgroundColor = backGroundColorLabel;</pre>
1149	%% Create ConfEvalSmpDropDown
1150	<pre>app.ConfEvalSmpDropDown = uidropdown(app.ConfEvalTabGridLayout);</pre>
1151	<pre>app.ConfEvalSmpDropDown.Layout.Row = 2;</pre>
1152	<pre>app.ConfEvalSmpDropDown.Layout.Column = [7 8];</pre>
1153	app.ConfEvalSmpDropDown.ValueChangedFcn = $@(src, event)$
	<pre>ConfEvalSmpDropDown_ValueChangedFcn(app, src, event);</pre>
1154	<pre>app.ConfEvalSmpDropDown.Items = {};</pre>
1155	<pre>app.ConfEvalSmpDropDown.Enable = 'off';</pre>
1156	<pre>app.ConfEvalSmpDropDown.FontSize = app.fontsize14;</pre>
1157	<pre>%% Create ConfEvalSmpImLabel</pre>
± 1158	<pre>app.ConfEvalSmpImLabel = uilabel(app.ConfEvalTabGridLayout);</pre>
1159	<pre>app.ConfEvalSmpImLabel.Text = 'Image No.';</pre>
1160	<pre>app.ConfEvalSmpImLabel.Layout.Row = 1;</pre>
1161	<pre>app.ConfEvalSmpImLabel.Layout.Column = 9 ;</pre>
1162	<pre>app.ConfEvalSmpImLabel.FontSize = app.fontsize14;</pre>
D 1163	<pre>app.ConfEvalSmpImLabel.Interpreter = 'latex';</pre>
H 1164	<pre>app.ConfEvalSmpImLabel.HorizontalAlignment = 'center';</pre>
1 165	<pre>app.ConfEvalSmpImLabel.BackgroundColor = backGroundColorLabel;</pre>
1166	<pre>%% Create ConfEvalSmpImDropDown</pre>
P 1167	<pre>app.ConfEvalSmpImDropDown = uidropdown(app.ConfEvalTabGridLayout);</pre>
1168	<pre>app.ConfEvalSmpImDropDown.Layout.Row = 2;</pre>
2 1169	<pre>app.ConfEvalSmpImDropDown.Layout.Column = 9;</pre>
<u>n</u> 1170	<pre>app.ConfEvalSmpImDropDown.ValueChangedFcn = @(src, event)</pre>
GSI	ConfEvalSmpImDropDown_ValueChangedFcn(app, src, event);
1171	<pre>app.ConfEvalSmpImDropDown.Items = {};</pre>
1172	<pre>app.ConfEvalSmpImDropDown.Enable = 'off';</pre>
2 1173	<pre>app.ConfEvalSmpImDropDown.FontSize = app.fontsize14;</pre>
2 1174	<pre>%% Create ConfEvalDrawRectButton</pre>
P 1175	<pre>app.ConfEvalDrawRectButton = uibutton(app.ConfEvalTabGridLayout);</pre>
1176 g	<pre>app.ConfEvalDrawRectButton.Text = 'Guess Beamarea';</pre>
B 1177	<pre>app.ConfEvalDrawRectButton.Layout.Row = 2;</pre>
n 1178	<pre>app.ConfEvalDrawRectButton.Layout.Column = 10;</pre>
Š 1179	<pre>app.ConfEvalDrawRectButton.ButtonPushedFcn = @(src, event)</pre>
ddr	ConfEvalDrawRectButton_ButtonPushedFcn(app, src, event);
1180 פ	<pre>app.ConfEvalDrawRectButton.FontSize = app.fontsize14;</pre>
1181	<pre>%% Create ConfEvalPreviewImUIAxes</pre>
1182	<pre>app.ConfEvalPreviewImUIAxes = uiaxes(app.ConfEvalTabGridLayout);</pre>
1183	<pre>app.ConfEvalPreviewImUIAxes.Layout.Row = [3 14];</pre>
1 184	<pre>app.ConfEvalPreviewImUIAxes.Layout.Column = [7 10];</pre>
B 1185	%%Create LiveMeasTab
b 1186	<pre>app.LiveMeasTab = uitab(app.TabGroup);</pre>
1 187	<pre>app.LiveMeasTab.Title = 'Live Measurement';</pre>
<u>≻</u>	

1188	<pre>app.LiveMeasTab.Scrollable = 'on';</pre>
1189	<pre>app.LiveMeasTabGridLayout = uigridlayout(app.LiveMeasTab);</pre>
1190	<pre>app.LiveMeasTabGridLayout.RowHeight = {'1x','1x','1x','1x','1x','1x','1x','1x'</pre>
	','1x','1x','1x','1x','1x','1x','1x','1
1191	<pre>app.LiveMeasTabGridLayout.ColumnWidth = {'1x','1x','1x','1x','1x','1x','1x','1x'</pre>
	1x', '1x', '1x'}
1192	<pre>%% Create LiveMeasBackCorrMethodLabel</pre>
1193	<pre>app.LiveMeasBackCorrMethodLabel = uilabel(app.LiveMeasTabGridLayout);</pre>
1194	app.LiveMeasBackCorrMethodLabel.Text = $'Coarse Background Correction$
	Method':
1195	<pre>app.LiveMeasBackCorrMethodLabel.Layout.Row = 2;</pre>
1196	app.LiveMeasBackCorrMethodLabel.Lavout.Column = [1 2];
1197	app.LiveMeasBackCorrMethodLabel.FontSize = app.fontsize14;
1198	app.LiveMeasBackCorrMethodLabel.Interpreter = 'latex':
1199	<pre>app.LiveMeasBackCorrMethodLabel.BackgroundColor = backGroundColorLabel:</pre>
1200	<pre>app.LiveMeasBackCorrMethodLabel.HorizontalAlignment = 'center':</pre>
1201	%% Create LiveMeasBackCorrMethodDropDown
1202	app.LiveMeasBackCorrMethodDropDown = uidropdown(app.
	LiveMeasTabGridLayout);
1203	app.LiveMeasBackCorrMethodDropDown.Layout.Row = 3;
1204	app.LiveMeasBackCorrMethodDropDown.Layout.Column = [1 2];
1205	<pre>% app.LiveMeasBackCorrMethodDropDown.Items = {'Background</pre>
)	<pre>Map Substraction', 'Average Background Substraction'};</pre>
1206	<pre>app.LiveMeasBackCorrMethodDropDown.FontSize = app.fontsize14;</pre>
1207	linkprop([app.ConfEvalBackCorrMethodDropDown app.
5	LiveMeasBackCorrMethodDropDown], 'Items');
1208	<pre>%% Create LiveMeasFineBackCorrMethodLabel</pre>
1209	app.LiveMeasFineBackCorrMethodLabel = uilabel(app.LiveMeasTabGridLayout)
2	;
2 1210	<pre>app.LiveMeasFineBackCorrMethodLabel.Text = 'Fine Background Correction</pre>
5	Method';
1211	<pre>app.LiveMeasFineBackCorrMethodLabel.Layout.Row = 2;</pre>
1212	<pre>app.LiveMeasFineBackCorrMethodLabel.Layout.Column = [3 4];</pre>
1213	<pre>app.LiveMeasFineBackCorrMethodLabel.FontSize = app.fontsize14;</pre>
3 1214	<pre>app.LiveMeasFineBackCorrMethodLabel.Interpreter = 'latex';</pre>
2 1215	<pre>app.LiveMeasFineBackCorrMethodLabel.BackgroundColor =</pre>
3	<pre>backGroundColorLabel;</pre>
21216	<pre>app.LiveMeasFineBackCorrMethodLabel.HorizontalAlignment = 'center';</pre>
3 1217	<pre>%% Create LiveMeasFineBackCorrMethodDropDown</pre>
3 1218	app.LiveMeasFineBackCorrMethodDropDown = uidropdown(app.
2	LiveMeasTabGridLayout);
1219	<pre>app.LiveMeasFineBackCorrMethodDropDown.Layout.Row = 3;</pre>
1220	<pre>app.LiveMeasFineBackCorrMethodDropDown.Layout.Column = [3 4];</pre>
1221	linkprop([app.ConfEvalFineBackCorrMethodDropDown app.
	LiveMeasFineBackCorrMethodDropDown],'Items');
1222	<pre>app.LiveMeasFineBackCorrMethodDropDown.FontSize = app.fontsize14;</pre>
a 1223	% Create LiveMeasFineCorrValueLabel
1224	<pre>app.LiveMeasFineCorrValueLabel = uilabel(app.LiveMeasTabGridLayout);</pre>
1225	<pre>app.LiveMeasFineCorrValueLabel.Text = 'Fine Corr Value';</pre>
_	

1226	app.liveMeasFineCorrValueLabel.lavout.Row = 2:
1227	applized and a second
1228	applized and a second
1220	appliveMeasFineCorrValueLabel Interpreter = 'latey':
1220	app.LiveMeasFineCorrValueLabel BackgroundColor = backGroundColorLabel:
1230	app.LiveMeasFineCorrValueLabel HorizontalAlignment = 'cepter':
1231	%% (reate LiveMeasEineCorrValueEditEield
1232	ann LiveMeasEineCorrValueEditEield - uieditfield(ann
1200	LiveMeasTabGridLavout 'numeric'):
1934	app LiveMeasEineCorrValueEditEield Lavout Row = 3;
1235	app.LiveMeasFineCorrValueEditField Lavout Column - 5:
1236	app.LiveMeasFineCorrValueEditField Limits = [0, 10]
1200 ⊾ 1237	\approx app. Civeneasi inecon vacuelui (i retu. Limits = [0 i0],
gba	ConfEvalNtEactorSlider AllEvents(ann_src_event)
1938	app LiveMeacEineCorrValueEditEiold HerizentalAlignment = 'conter';
A 1230	app.LiveMeasFineCorrValueEditField_FontSize = app_fontsize14;
1209 4 1240	app.LiveMeasFineCorrValueEditField.FontSize = app.TontSize14,
1240	app.LiveMeasFineConvergeCriterienLabel
Here Here Here Here Here Here Here Here	%% Create LiveMeasConvergeCriterionLabel
1242	app.LiveMeasConvergeCriterionLabel = uitabet(app.LiveMeasTabGridLayout);
A dia 1740 Vala	app.LiveMedsconvergecriterionLabet.rext = \$\textrm{convergence}
r Tl	Criterion} (textrm{ in } (mu m\$;
P > 1244	app.LiveMeasConvergeCriterionLabel.Layout.Row = 4;
1240 1240	app.LiveMeasConvergeCriterionLabel.Layout.Cotumn = [1 2];
1240 1247	app.LiveMeasConvergeCriterionLabel.FontSize = app.TontSize14;
ip 1247	app.LiveMeasConvergeCriterionLabel.Interpreter = 'latex';
	app.LiveMeasconvergecriterionLabel.Backgroundcolor =
oldi able	backGroundLolorLabel;
G 1249	app.LiveMeasConvergeCriterionLabel.HorizontalAlignment = "center";
9 S 1 250	%% Create LiveMeasConvergeCriterionEditField
n di sis i	app.LiveMeasConvergecriterionEditrietd = uieditrietd(app.
sion these	LiveMeasTabGridLayout, "numeric");
Ver 1252	app.LiveMeasConvergeCriterionEditField.Layout.Row = 5;
0 1203	app.LiveMeasConvergeCriterionEditField.Layout.Column = [1 2];
01204 0100	app.LiveMeasConvergeCriterionEditField.Limits = [0.1 10];
tte (app.LiveMeasConvergeCriterionEditField.HorizontalAlignment = center;
D 1057	app.LiveMeasConvergecriterionEditrietd.rontSize = app.rontSize14;
ledr 1201	% Credie LiveMeasNiFactorLabel
1200 J200	app.LiveMeasNtFactorLabel = ullabel(app.LiveMeasTabGridLayout);
01209	app.LiveMeasNtFactorLabel.Text = III standard deviation muttipticator;
0 1200	app.LiveMeasNtFactorLabel.Layout.Row = 0;
d 1201	app.LiveMeasNtFactorLabel.Layout.column = [1 2];
Die 1202	app.LiveMeasNtFactorLabel.FontSize = app.TontSize14;
1203	app.LiveMeasNtFactorLabel.Interpreter = tatex ;
	app.LiveMeasNtFactorLabel.Backgroundcolor = backgroundcolorLabel;
4 1205	app.LiveMeasNtFactorLabel.HorizontalAlignment = 'center';
	%% UFEATE LIVEMEASNTEOITFIELO
	app.LivemeasNiteGitField = uieGitTield(app.LivemeasTabGridLayOut, 'numeric
);
m § 1208	app.LivemeaswiteditField.Layout.Kow = /;
I	

1000	
1209	app.LiveMeasNtEditField.Layout.Column = [1 2];
1270	app.LiveMeasNtEditField.Limits = [2 4];
1271	app.LiveMeasNtEditField.HorizontalAlignment = 'center';
1272	<pre>app.LiveMeasNtEditField.FontSize = app.fontsize14;</pre>
1273	<pre>%% Create LiveMeasIntegAreaFactorLabel</pre>
1274	<pre>app.LiveMeasIntegAreaFactorLabel = uilabel(app.LiveMeasTabGridLayout);</pre>
1275	<pre>app.LiveMeasIntegAreaFactorLabel.Text = 'Integrationrange multiplicator'</pre>
1276	app.LiveMeasIntegAreaFactorLabel.Layout.Row = 8 ;
1277	app.LiveMeasIntegAreaFactorLabel.Lavout.Column = $[1 2]$:
1278	app.liveMeasIntegAreaEactorLabel.FontSize = app.fontsize14:
1279	applieurositegAreaEactorlabel.Interpreter = 'latex':
1210	app LiveMeasIntegAreaEactorLabel BackgroundColor = backGroundColorLabel:
1200	app.LiveMeasIntegAreaEactorLabel HorizontalAlignment = 'center':
1201	% Create LiveMeasIntegAreaEditEiold
1902	opp LiveMeasIntegAreaEditEiold = wieditfield(app LiveMeasTabCridlavout /
1200	app.LiveMedSintegAreaEultrietu = uieultrietu(app.LiveMedSiabGrutayout,
1004	numeric);
1204	app.LiveMeasIntegAreaEditField.Layout.Row = 9;
1285	app.LiveMeasIntegAreaEditField.Layout.Column = [1 2];
1286	<pre>app.LiveMeasIntegAreaEditField.Limits = [1 3];</pre>
1287	<pre>app.LiveMeasIntegAreaEditField.HorizontalAlignment = 'center';</pre>
§ 1288	<pre>app.LiveMeasIntegAreaEditField.FontSize = app.fontsize14;</pre>
<u>p</u> 1289	<pre>%% Create LiveMeasResLabel</pre>
g 1290	app.LiveMeasResLabel = uilabel(app.LiveMeasTabGridLayout);
1291	<pre>app.LiveMeasResLabel.Text = 'Image Resolution W x H in pixel';</pre>
1292	<pre>app.LiveMeasResLabel.Layout.Row = 12;</pre>
1293	<pre>app.LiveMeasResLabel.Layout.Column = [1 2];</pre>
1294	<pre>app.LiveMeasResLabel.FontSize = app.fontsize14;</pre>
a 1295	<pre>app.LiveMeasResLabel.Interpreter = 'latex';</pre>
2 1296	<pre>app.LiveMeasResLabel.BackgroundColor = backGroundColorLabel;</pre>
g 1297	<pre>app.LiveMeasResLabel.HorizontalAlignment = 'center';</pre>
1298	<pre>%% Create LiveMeasResWidthEditField</pre>
1299	<pre>app.LiveMeasResWidthEditField = uieditfield(app.LiveMeasTabGridLayout,'</pre>
5	numeric');
1300	<pre>app.LiveMeasResWidthEditField.Layout.Row = 13;</pre>
1301	app.LiveMeasResWidthEditField.Lavout.Column = 1:
d 1302	app.LiveMeasResWidthEditEield.HorizontalAlignment = 'center':
2 1303	app.liveMeasResWidthEditEield.FontSize = app.fontsize14:
1304	applied Heastern and the second applied to
3 1305	% Create LiveMeasResHeightEditField
1306	app liveMeasBesHeightEditEield = uieditfield(app liveMeasTabGridLavout '
a 1000	numeric'):
1307	app LiveMeasResHeightEditEield Lavout Row - 13:
1202	app. LiveMeasResHeightEditField Lavout Column - 2.
1200	app.LiveMeasCocHeightEditEield HerizentalAlignment - Leonterly
1910	app.LiveMeasResnerghtEditEicld EastSize = app fastsize14;
	app.LiveMeasResnerghiceurrietu.Fontsize = app.Tontsize14;
1919	app.Livenedskesneighteutrietu.vatue = 1020;
1919	% Uledle Liveredsrikelsizeldbel
1919 1919	app.Livemeasrixe(sizeLabet = ultabet(app.LiveMeasTabGridLayout);

TU **Bibliothek**, Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar ^{WIEN} ^{vur knowedge hub} The approved original version of this thesis is available in print at TU Wien Bibliothek.

1015	<pre>app.LiveMeasFixe(sizeLabe(.lext = \$\textfm{Fixe(size} \textfm{ if } \mu m\$'; </pre>
1315	app.LiveMeasPixelsizeLabel.Layout.Row = 12;
1316	<pre>app.LiveMeasPixelsizeLabel.Layout.Column = 3;</pre>
1317	<pre>app.LiveMeasPixelsizeLabel.FontSize = app.fontsize14;</pre>
1318	<pre>app.LiveMeasPixelsizeLabel.Interpreter = 'latex';</pre>
1319	<pre>app.LiveMeasPixelsizeLabel.BackgroundColor = backGroundColorLabel;</pre>
1320	<pre>app.LiveMeasPixelsizeLabel.HorizontalAlignment = 'center';</pre>
1321	%% Create LiveMeasPixelsizeEditField
1322	app.LiveMeasPixelsizeEditField = uieditfield(app.LiveMeasTabGridLayout,' numeric');
1323	<pre>app.LiveMeasPixelsizeEditField.Layout.Row = 13;</pre>
ਸ਼ੂ 1324	<pre>app.LiveMeasPixelsizeEditField.Layout.Column = 3;</pre>
ap 1325	<pre>app.LiveMeasPixelsizeEditField.HorizontalAlignment = 'center';</pre>
1326	<pre>app.LiveMeasPixelsizeEditField.FontSize = app.fontsize14;</pre>
> x 1327	<pre>app.LiveMeasPixelsizeEditField.Value = 5.3;</pre>
tio 1328	<pre>%% Create LiveMeasKernelSizeLabel</pre>
1329	<pre>app.LiveMeasKernelSizeLabel = uilabel(app.LiveMeasTabGridLayout);</pre>
u 1330	app.LiveMeasKernelSizeLabel.Text = 'Kernelsize in \%';
	<pre>app.LiveMeasKernelSizeLabel.Layout.Row = 6;</pre>
<u>∩</u> <u></u> 1332	<pre>app.LiveMeasKernelSizeLabel.Layout.Column = [3 4];</pre>
a ¥ 1333	<pre>app.LiveMeasKernelSizeLabel.FontSize = app.fontsize14;</pre>
u ⊇ 1334	<pre>app.LiveMeasKernelSizeLabel.Interpreter = 'latex';</pre>
1335 at 13	app.LiveMeasKernelSizeLabel.BackgroundColor = backGroundColorLabel;
1336 Line ti	<pre>app.LiveMeasKernelSizeLabel.HorizontalAlignment = 'center';</pre>
d u 1337	<pre>%% Create LiveMeasKernelSizeEditField</pre>
<u>e</u> 1338	app.LiveMeasKernelSizeEditField = uieditfield(app.LiveMeasTabGridLayout,
Dip	'numeric');
ag 1339	<pre>app.LiveMeasKernelSizeEditField.Layout.Row = 7;</pre>
<u>e</u> 3 1340	<pre>app.LiveMeasKernelSizeEditField.Layout.Column = [3 4];</pre>
<u>1341</u>	<pre>app.LiveMeasKernelSizeEditField.Limits = [2 5];</pre>
s tr 1342	<pre>app.LiveMeasKernelSizeEditField.HorizontalAlignment = 'center';</pre>
Tar 1343	<pre>app.LiveMeasKernelSizeEditField.FontSize = app.fontsize14;</pre>
1344 l	<pre>%% Create LiveMeasMethodLabel</pre>
O is 1345	app.LiveMeasMethodLabel = uilabel(app.LiveMeasTabGridLayout);
<u>ty 9</u> 1346	<pre>app.LiveMeasMethodLabel.Text = 'Evaluation Norm';</pre>
1347	app.LiveMeasMethodLabel.Layout.Row = 16;
a) 01348	app.LiveMeasMethodLabel.Layout.Column = [1 2];
ti p 1349	app.LiveMeasMethodLabel.FontSize = app.fontsize14;
g o 1350	app.LiveMeasMethodLabel.Interpreter = 'latex';
dd 1351	app.LiveMeasMethodLabel.BackgroundColor = backGroundColorLabel;
e 1352	<pre>app.LiveMeasMethodLabel.HorizontalAlignment = 'center';</pre>
□ ⊢ 1353	<pre>%% Create LiveMeasMethodDropDown</pre>
1354	<pre>app.LiveMeasMethodDropDown = uidropdown(app.LiveMeasTabGridLayout);</pre>
	<pre>app.LiveMeasMethodDropDown.Layout.Row = 17;</pre>
	<pre>app.LiveMeasMethodDropDown.Layout.Column = [1 2];</pre>
	<pre>app.LiveMeasMethodDropDown.Items = {'IS011146-1'};</pre>
9 1358	<pre>app.LiveMeasMethodDropDown.FontSize = app.fontsize14;</pre>
1359	%% Create LiveMeasPreviewImUIAxes

1360	<pre>app.LiveMeasPreviewImUIAxes = uiaxes(app.LiveMeasTabGridLayout);</pre>
1361	<pre>app.LiveMeasPreviewImUIAxes.Layout.Row = [1 8];</pre>
1362	<pre>app.LiveMeasPreviewImUIAxes.Layout.Column = [7 10];</pre>
1363	<pre>%% Create LiveMeasResultImUIAxes</pre>
1364	<pre>app.LiveMeasResultImUIAxes = uiaxes(app.LiveMeasTabGridLayout);</pre>
1365	<pre>app.LiveMeasResultImUIAxes.Layout.Row = [9 16];</pre>
1366	<pre>app.LiveMeasResultImUIAxes.Layout.Column = [7 10];</pre>
1367	<pre>%% Create LiveMeasStartStopButton</pre>
1368	<pre>app.LiveMeasStartStopButton = uibutton(app.LiveMeasTabGridLayout,'state'</pre>
);
1369	<pre>app.LiveMeasStartStopButton.Text = 'Start/Stop Live Measurement';</pre>
1370	<pre>app.LiveMeasStartStopButton.Layout.Row = 17;</pre>
1371	<pre>app.LiveMeasStartStopButton.Layout.Column = [3 4];</pre>
1372	<pre>app.LiveMeasStartStopButton.ValueChangedFcn = @(src, event)</pre>
	<pre>LiveMeasStartStopButton_ValueChangedFcn(app, src, event);</pre>
1373	<pre>app.LiveMeasStartStopButton.FontSize = app.fontsize14;</pre>
1374	<pre>app.LiveMeasStartStopButton.Enable = 'off';</pre>
1 375	<pre>%% Create LiveMeasBackgroundMeasButton</pre>
1376	<pre>app.LiveMeasBackgroundMeasButton = uibutton(app.LiveMeasTabGridLayout);</pre>
1377	<pre>app.LiveMeasBackgroundMeasButton.Text = 'Measure Background';</pre>
<u>1378</u>	<pre>app.LiveMeasBackgroundMeasButton.Layout.Row = 16;</pre>
1379	<pre>app.LiveMeasBackgroundMeasButton.Layout.Column = [3 4];</pre>
5 1380	<pre>app.LiveMeasBackgroundMeasButton.ButtonPushedFcn = @(src, event)</pre>
વા	LiveMeasBackgroundMeasButton_ButtonPushedFcn(app, src, event);
1 381	<pre>app.LiveMeasBackgroundMeasButton.FontSize = app.fontsize14;</pre>
1382	<pre>%% Create AutoMeasIS0111146Part1StatusLabel</pre>
1383	<pre>app.LiveMeasStatusLabel = uilabel(app.LiveMeasTabGridLayout);</pre>
1384	<pre>app.LiveMeasStatusLabel.Layout.Row = 17;</pre>
1385	<pre>app.LiveMeasStatusLabel.Layout.Column = [5 6];</pre>
<u>n</u> 1386	<pre>app.LiveMeasStatusLabel.FontSize = app.fontsize14;</pre>
1387	<pre>app.LiveMeasStatusLabel.Interpreter = 'latex';</pre>
1388	<pre>app.LiveMeasStatusLabel.BackgroundColor = [1 0.07 0.2];</pre>
1 389	<pre>app.LiveMeasStatusLabel.HorizontalAlignment = 'center';</pre>
2 1390	<pre>app.LiveMeasStatusLabel.Text = 'Inactive';</pre>
<u>R</u> 1391	<pre>%% Create LiveMeasImData_dWx_Label</pre>
P 1392	app.LiveMeasImData_dWx_Label = uilabel(app.LiveMeasTabGridLayout);
a 1393	<pre>app.LiveMeasImData_dWx_Label.Text = '\$\textrm{Beamdiameter } d_{x} \</pre>
Bur	<pre>textrm{ in } \mu m\$';</pre>
g 1394	<pre>app.LiveMeasImData_dWx_Label.Interpreter = 'latex';</pre>
Š 1395	<pre>app.LiveMeasImData_dWx_Label.HorizontalAlignment = 'center';</pre>
d 1396	<pre>app.LiveMeasImData_dWx_Label.BackgroundColor = backGroundColorLabel;</pre>
<u>1397</u> נ	<pre>app.LiveMeasImData_dWx_Label.Layout.Row = 9;</pre>
1398	app.LiveMeasImData_dWx_Label.Layout.Column = [4 5];
1399	<pre>%% Create LiveMeasImData_dWx</pre>
1400	<pre>app.LiveMeasImData_dWx = uieditfield(app.LiveMeasTabGridLayout, 'numeric'</pre>
and and a second s);
b 1401	<pre>app.LiveMeasImData_dWx.HorizontalAlignment = 'center';</pre>
2 1402	<pre>app.LiveMeasImData_dWx.Editable = 'off';</pre>
1 403	<pre>app.LiveMeasImData_dWx.Layout.Row = 9;</pre>
≻	

1404	app.LiveMeasImData_dWx.Lavout.Column = 6:
1405	%% Create LiveMeasImData_dWv_Label
1406	app.LiveMeasImData dWv Label = uilabel(app.LiveMeasTabGridLavout):
1407	app.liveMeasImData dWv label.Text = '\$\textrm{Beamdiameter } d {v} \
1 10 1	textrm{ in } \mu m\$':
1408	ann LiveMeasImData dWy Label Interpreter = 'latex':
1409	app LiveMeasImData dWy Label HorizontalAlignment = 'center':
1/10	app.LiveMeasImData_dWy_Label_BackgroundColor = backGroundColorlabel;
1410	$app.LiveMeasImData_dwy_Label.backgroundcotor = backgroundcotorLabet,$
1/12	app.LiveNeasImData_dWy_Label Layout Column = $[4, 5]$.
1/12	% Croate LiveMeasImData dWy
1415	ann LiveMeasImData dWy - uioditfield(ann LiveMeasTabGridlaveut 'numeri
1414	app.liveneasimbata_dwy - diedititetd(app.liveneasiaborideayout, numeri
1/15	/, app LiveMeasTmData dww HerizentalAlignment - 'conter',
1410	app.LiveMeasImData_dwy.HorizontatAtignment = center;
1410	app.LiveMeasImData_uwy.Euitable = 011 ;
1410	app.LiveMeasImData_uwy.Layout.Row = 10;
1410	app.LivereasimData_uwy.Layout.Cotumn = 0;
¥ 1419	%% Create LiveMeasimpata_pni_Label
1420	app.LiveMeasImData_phi_Label = uilabel(app.LiveMeasIabGridLayout);
<u>e</u> 1421	app.LiveMeasImData_phi_Label.Text = '\$\textrm{Azimutangle } \varphi \
eu la construction	<pre>textrm{ in } A°\$';</pre>
₹ 1422 1.400	<pre>app.LiveMeasImData_phi_Label.Interpreter = 'latex';</pre>
2 1423	<pre>app.LiveMeasImData_phi_Label.HorizontalAlignment = 'center';</pre>
te 1424	<pre>app.LiveMeasImData_phi_Label.BackgroundColor = backGroundColorLabel;</pre>
E 1425	app.LiveMeasImData_phi_Label.Layout.Row = 11;
⊑ ¹⁴²⁶	app.LiveMeasImData_phi_Label.Layout.Column = [4 5];
e 1427	%% Create LiveMeasImData_phi
alla 1428	app.LiveMeasImData_phi = uieditfield(app.LiveMeasTabGridLayout,'numeri
av);
<u>s</u> 1429	app.LiveMeasImData_phi.HorizontalAlignment = 'center';
S 1430	app.LiveMeasImData_phi.Editable = 'off';
<u>s</u> 1431	app.LiveMeasImData_phi.Layout.Row = 11;
E 1432	app.LiveMeasImData_phi.Layout.Column = 6;
o 1433	<pre>%% Create LiveMeasTriggerImageButton</pre>
S 1434	app.LiveMeasTriggerImageButton = uibutton(app.LiveMeasTabGridLayout);
9 1435	app.LiveMeasTriggerImageButton.Text = 'Trigger Image';
E 1436	app.LiveMeasTriggerImageButton.Layout.Row = 2;
Buo 1437	app.LiveMeasTriggerImageButton.Layout.Column = 6;
p 1438	app.LiveMeasTriggerImageButton.ButtonPushedFcn = @(src, event)
NO.	LiveMeasTriggerImageButton_ButtonPushedFcn(app, src, event);
d 1439	<pre>app.LiveMeasTriggerImageButton.FontSize = app.fontsize14;</pre>
e 1440	<pre>%% Create LiveMeasDrawRectButton</pre>
⊢ 1441	app.LiveMeasDrawRectButton = uibutton(app.LiveMeasTabGridLayout);
1442	<pre>app.LiveMeasDrawRectButton.Text = 'Guess Beamarea';</pre>
1443	<pre>app.LiveMeasDrawRectButton.Layout.Row = 3;</pre>
<mark>ਊ</mark> 1444	<pre>app.LiveMeasDrawRectButton.Layout.Column = 6;</pre>
ag 1445	app.LiveMeasDrawRectButton.ButtonPushedFcn = @(src, event)
YMOL	LiveMeasDrawRectButton_ButtonPushedFcn(app, src, event);
¥ 1446	<pre>app.LiveMeasDrawRectButton.FontSize = app.fontsize14;</pre>

1447	<pre>%% Create LiveMeasChangeColorMapButton</pre>
1448	<pre>app.LiveMeasChangeColorMapButton = uibutton(app.LiveMeasTabGridLayout, '</pre>
	<pre>state');</pre>
1449	<pre>app.LiveMeasChangeColorMapButton.Text = 'Switch Color Map';</pre>
1450	<pre>app.LiveMeasChangeColorMapButton.Layout.Row = 13;</pre>
1451	<pre>app.LiveMeasChangeColorMapButton.Layout.Column = 6;</pre>
1452	<pre>app.LiveMeasChangeColorMapButton.FontSize = app.fontsize14;</pre>
1453	<pre>%% Create LiveMeasColorMinLabel</pre>
1454	<pre>app.LiveMeasColorMinLabel = uilabel(app.LiveMeasTabGridLayout);</pre>
1455	<pre>app.LiveMeasColorMinLabel.Text = 'Min Value';</pre>
1456	<pre>app.LiveMeasColorMinLabel.Layout.Row = 14;</pre>
1457	<pre>app.LiveMeasColorMinLabel.Layout.Column = 5;</pre>
1458	<pre>app.LiveMeasColorMinLabel.FontSize = app.fontsize14;</pre>
1459	<pre>app.LiveMeasColorMinLabel.Interpreter = 'latex';</pre>
1460	app.LiveMeasColorMinLabel.BackgroundColor = backGroundColorLabel;
1461	<pre>app.LiveMeasColorMinLabel.HorizontalAlignment = 'center';</pre>
1462	<pre>%% Create LiveMeasColorMinEditField</pre>
1463	<pre>app.LiveMeasColorMinEditField = uieditfield(app.LiveMeasTabGridLayout,'</pre>
10	numeric');
1464	<pre>app.LiveMeasColorMinEditField.Layout.Row = 14;</pre>
1465	<pre>app.LiveMeasColorMinEditField.Layout.Column = 6;</pre>
§ 1466	<pre>app.LiveMeasColorMinEditField.Limits = [0 255];</pre>
2 1467	<pre>app.LiveMeasColorMinEditField.HorizontalAlignment = 'center';</pre>
B 1468	<pre>app.LiveMeasColorMinEditField.FontSize = app.fontsize14;</pre>
1469	<pre>app.LiveMeasColorMinEditField.Value = app.LiveMeasColorMap(1);</pre>
1470	%% Create LiveMeasColorMaxLabel
9 1471 1472	<pre>app.LiveMeasColorMaxLabel = uilabel(app.LiveMeasTabGridLayout);</pre>
1472	app.LiveMeasColorMaxLabel.lext = 'Max Value';
g 1473	app.LiveMeasColorMaxLabel.Layout.Row = 15;
1474	app.LiveMeasColorMaxLabel.Layout.Column = 5;
3 1475	app.LiveMeasColorMaxLabel.FontSize = app.fontsize14;
<u>1476</u>	app.LiveMeasColorMaxLabel.Interpreter = 'latex';
5 1477	app.LivemeascolormaxLabel.Backgroundcolor = backGroundcolorLabel;
1478	<pre>app.LivemeascolorMaxLabel.HorizontalAlignment = 'center';</pre>
<u>5</u> 14(9 1 100	%% LFEATE LIVEMEASLOLOFMINECTTFIELC
5 148U	app.Livemeascolormaxcultrietu = uleultrietu(app.LivemeasiaDGr10Layout,
δ 1 / 01	$\frac{1}{10000000000000000000000000000000000$
5 1481 5 1489	app.LivemeascolormaxEditField.Layout.Kow = 15;
2 148Z 2 1402	app.LivemeasColorMaxEditField Limits = [0, 255];
0 1400 0 1400	$a\mu\mu.LiverieasCulurinasCullField HarizantalAlianment = 1 contart.$
1/94 1/95	app.LivereasColorMaxEditField EartSize = app.fontsize14;
= 1400 1/96	$a\mu\mu$. LiveneasColorMaxEditField Value - app LiveMeasColorMap(2).
1400 1/87	app.Liveneascotornaxcuttrietu.vatue = app.Liveneascotornap(2);
1/22	on liveMeasNoOfImlabel - uilabel(app liveMeasTabGridlaveut);
1400 91/00	app.LiverieaSNOUTIMLabel = uttabet(app.LiverieaSTabGTuLayOut);
E 1409	$a\mu\mu$. Liveneasinourimizabet. Text = ino or background images;
₽ 1490 1/01	$a\mu\mu$. Livereasivour mildbel. Layout. Kow = 14;
u 1/09	app.Livereasivourimeabel.Layour.corumn = [5, 4];
n 1492 X	app.LiveneasivoorimLabet.FullSize – app.FullSize14;
-	

1493	<pre>app.LiveMeasNoOfImLabel.Interpreter = 'latex';</pre>
1494	<pre>app.LiveMeasNoOfImLabel.BackgroundColor = backGroundColorLabel;</pre>
1495	<pre>app.LiveMeasNoOfImLabel.HorizontalAlignment = 'center';</pre>
1496	%% Create LiveMeasNo0fImEditField
1497	app.LiveMeasNoOfImEditField = uieditfield(app.LiveMeasTabGridLavout.
	numeric'):
1498	app.LiveMeasNoOfImEditField.Lavout.Row = 15 :
1499	app.LiveMeasNoOfImEditField.Lavout.Column = [3 4]:
1500	app.LiveMeasNoOfImEditField.Limits = [10 9999]:
1501	appliveMeasNoOfImEditField HorizontalAlignment = 'center':
1502	appliveMeasNoOfImEditField FontSize = app fontsize14:
1502	app LiveMeasNoOfImEditField Value = 20 .
<u>1503</u>	% Create EvalResTab
eq 1505	ann EvalResTab = $uitab(ann EvalTabGroup)$:
iju 1506	app.EvallesTab Title - 'Evaluation Result':
1507	app.EvalNesTab.Fitte = Evaluation Result ;
1507 .	% Croate EvalPosTabCridLavout
to 1500	on EvalPacTabCridLayout - uigridlayout(ann EvalPacTab).
Here A Tero	app.EvalkesiaburiuLayout = utgriulayoul(dpp.EvalkesiaD);
0161 tiot	app.EvalResTabGridLayout.RowHeight = { 1x };
	app.EvalResTabGridLayout.Columnwidth = { 'Ix', 'Ix', 'Ix'};
H 1512	%% Create Evalkesmeasuataranel
ē≥ 1513	app.EvalResMeasDataPanel = ulpanel(app.EvalResTabGridLayout);
1514 1515	%% Create EvalResMeasDataGridLayout
CIG1 at ist	app.EvalResMeasDataGridLayout = uigridlayout(app.EvalResMeasDataPanel);
0161 peri	app.EvalResMeasDataGridLayout.RowHeight = {30,30, '1x',30,30,30};
	<pre>app.EvalResMeasDataGridLayout.ColumnWidth = {'1x','1x'};</pre>
<u>ple</u> 1518	%% Create EvalResMeasLoadButton
1519	<pre>app.EvalResMeasLoadButton = uibutton(app.EvalResMeasDataGridLayout);</pre>
a 1520	<pre>app.EvalResMeasLoadButton.Layout.Row = 1;</pre>
ip is 1521	<pre>app.EvalResMeasLoadButton.Layout.Column = [1 2];</pre>
lise 1522	<pre>app.EvalResMeasLoadButton.Text = 'Load Measurement';</pre>
1523	<pre>app.EvalResMeasLoadButton.FontSize = app.fontsize14;</pre>
1524 It H	<pre>app.EvalResMeasLoadButton.ButtonPushedFcn = @(src, event)</pre>
Drigi	ConfEvalLoadMeasButton_ButtonPushedFcn(app, src, event);
0 .5. 1525	<pre>%% Create EvalResMeasDataResultDropDown_Label</pre>
¥ <u></u> 1526	app.EvalResMeasDataResultDropDown_Label = uilabel(app.
jina	EvalResMeasDataGridLayout);
ο θ 1527	app.EvalResMeasDataResultDropDown_Label.Layout.Row = 2;
ti p 1528	app.EvalResMeasDataResultDropDown_Label.Layout.Column = 1;
g o 1529	app.EvalResMeasDataResultDropDown_Label.Text = 'Result No.';
dd a 1530	app.EvalResMeasDataResultDropDown_Label.FontSize = app.fontsize14;
je 1531	app.EvalResMeasDataResultDropDown_Label.HorizontalAlignment = 'center';
$\Box \vdash 1532$	app.EvalResMeasDataResultDropDown_Label.BackgroundColor =
₹	<pre>backGroundColorLabel;</pre>
D 1533	<pre>app.EvalResMeasDataResultDropDown_Label.Interpreter = 'latex';</pre>
1 534	<pre>%% Create EvalResMeasDataResultDropDown</pre>
9 # 1535	app.EvalResMeasDataResultDropDown = uidropdown(app.
	<pre>EvalResMeasDataGridLayout);</pre>
1 536	<pre>app.EvalResMeasDataResultDropDown.Layout.Row = 2;</pre>

1537	app EvalResMeasDataResultDropDown Layout Column = 2:
1538	app. EvalResMeasDataResultDronDown ValueChangedEcn = β (src_event)
1000	EvalPosMoscDataPosultDropDown.ValueChangedEcn(app_src_ovent);
1520	EvalResMeasDataResultDropDown_ValueChangedich(app, sic, event),
1540	app. EvalPacMassDataPasultDropDown. Items $- \{ \},$
1540 15/11	<pre>app.EvalNessBataNessBataParamsGrid avout</pre>
1541 1549	ann EvalPocMoacDataParameGridLayout - uigridlayout(ann
1042	$E_{val} Res Resc Data GridLavout) \cdot$
15/13	app EvalResMeasDataDatameGridLayout Layout Row - 3:
1544	app. Evalles MeasDataParamsGridLayout Layout Column = $[1, 2]$.
1545	app.EvalleshedsbatararamsGridLayout.Edyout.Cotumn = [1 2];
1040	$\int 20 \ 20 \ 20 \ 20 \ 20 \ 20 \ 20 \ 20 $
1546	[20, 20, 20, 20, 20, 20, 20, 20, 20, 20,
1540 1547	app. Evalles MeasData and instructayout. Cordination $- \{1, 7, 7, 7\}$
1548	%% Create EvalResMeasData EvalNorm Label
15/10	ann EvalResMeasData EvalNorm Label = uilabel(ann
1049	EvalResMeasData_EvalNorm_Laber = ditaber(app.
: 1550	app EvalResMeasData EvalNorm Label Text - 'Evaluation Norm':
1551	app.EvalNesheasData_EvalNorm_Label_Interpreter - 'latex':
1552	app.EvalNesMeasData_EvalNorm_Label_HorizontalAlignment = 'contor';
1552	app.EvalNesMeasData_EvalNorm_Label_RackgroundColor -
1000	hackGroundColorlabel:
5 155/	Sector ounded for Laber,
1555	ann EvalResMeasData EvalNorm - uieditfield(ann
E 1000	EvalPosMoasData_EvalNorm = dieditietd(app:
1556	app EvalResMeasData EvalNorm HorizontalAlignment - 'center'
1557	app.EvallesMeasData_EvalNorm_Editable = off' .
1558	%% Create EvalResMeasData BackCorrMethod Label
1559	ann EvalResMeasData Back(orrMethod Label = uilabel(ann
0 1000	EvalResMeasDataParamsGridLavout):
2 1560	app EvalResMeasData BackCorrMethod Label Text = 'Background Correction
	Method':
1561	app EvalResMeasData BackCorrMethod Label Interpreter = 'latex':
5 1562	app.EvalResMeasData BackCorrMethod Label HorizontalAlignment = 'center':
1563	app.EvalResMeasData_BackCorrMethod_Label_BackgroundColor =
0 1000	hackGroundColorLabel:
1564	% Create EvalResMeasData BackCorrMethod
2 1565	app.EvalResMeasData BackCorrMethod = uieditfield(app.
5 1000	EvalResMeasDataParamsGridLavout. 'text'):
5 1566	app.EvalResMeasData BackCorrMethod.HorizontalAlignment = 'center':
1567	app.EvalResMeasData BackCorrMethod.Editable = 'off':
u 1568	% Create EvalResMeasData_ConvCrit_Label
1569	app.EvalResMeasData ConvCrit Label = uilabel(app.
_000	EvalResMeasDataParamsGridLavout):
1570	app.EvalResMeasData ConvCrit Label.Text = '\$Convergence
9	Criterion} \textrm{ in } \mu m\$':
1 571	app.EvalResMeasData_ConvCrit_Label.Interpreter = 'latex':
1572	app.EvalResMeasData ConvCrit Label HorizontalAlignment = 'center'
1573	app.EvalResMeasData_ConvCrit_Label.BackgroundColor =
2	

	backGroundColorLabel;
1574	<pre>%% Create EvalResMeasData_ConvCrit</pre>
1575	app.EvalResMeasData_ConvCrit = uieditfield(app.
	<pre>EvalResMeasDataParamsGridLayout, 'numeric');</pre>
1576	<pre>app.EvalResMeasData_ConvCrit.HorizontalAlignment = 'center';</pre>
1577	<pre>app.EvalResMeasData_ConvCrit.Editable = 'off';</pre>
1578	%% Create EvalResMeasData_Lambda_Label
1579	app.EvalResMeasData_Lambda_Label = uilabel(app.
	EvalResMeasDataParamsGridLayout);
1580	<pre>app.EvalResMeasData_Lambda_Label.Text = '\$\textrm{Wavelength }\lambda \</pre>
	<pre>textrm{ in } nm \$';</pre>
1581	<pre>app.EvalResMeasData_Lambda_Label.Interpreter = 'latex';</pre>
1582	<pre>app.EvalResMeasData_Lambda_Label.HorizontalAlignment = 'center';</pre>
1583	<pre>app.EvalResMeasData_Lambda_Label.BackgroundColor = backGroundColorLabel;</pre>
1584	% Create EvalResMeasData_Lambda
1585	app.EvalResMeasData_Lambda = uieditfield(app.
	<pre>EvalResMeasDataParamsGridLayout, 'numeric');</pre>
1586	<pre>app.EvalResMeasData_Lambda.HorizontalAlignment = 'center':</pre>
1587	app.EvalResMeasData_Lambda.Editable = 'off':
1588	%% Create EvalResMeasData_IntegAreaFactor_Label
1589	app.EvalResMeasData_IntegAreaFactor_Label = uilabel(app.
	EvalResMeasDataParamsGridLavout):
1590	app.EvalResMeasData_IntegAreaFactor_Label.Text = 'Integration Area Size
	Multiplicator'
1591	app.EvalResMeasData_IntegAreaFactor_Label.Interpreter = 'latex':
1592	app.EvalResMeasData_IntegAreaFactor_Label.HorizontalAlignment = 'center'
	:
1593	<pre>app.EvalResMeasData_IntegAreaFactor_Label.BackgroundColor =</pre>
	backGroundColorLabel;
1594	%% Create EvalResMeasData_IntegAreaFactor
1595	<pre>app.EvalResMeasData_IntegAreaFactor = uieditfield(app.</pre>
	EvalResMeasDataParamsGridLayout, 'numeric');
1596	<pre>app.EvalResMeasData_IntegAreaFactor.HorizontalAlignment = 'center';</pre>
1597	<pre>app.EvalResMeasData_IntegAreaFactor.Editable = 'off';</pre>
1598	%% Create EvalResMeasData_Nt_Label
1599	app.EvalResMeasData_Nt_Label = uilabel(app.
	<pre>EvalResMeasDataParamsGridLayout);</pre>
1600	<pre>app.EvalResMeasData_Nt_Label.Text = 'nT';</pre>
1601	<pre>app.EvalResMeasData_Nt_Label.Interpreter = 'latex';</pre>
1602	<pre>app.EvalResMeasData_Nt_Label.HorizontalAlignment = 'center';</pre>
1603	<pre>app.EvalResMeasData_Nt_Label.BackgroundColor = backGroundColorLabel;</pre>
1604	%% Create EvalResMeasData_Nt
1605	app.EvalResMeasData_Nt = uieditfield(app.EvalResMeasDataParamsGridLayout
	<pre>,'numeric');</pre>
1606	<pre>app.EvalResMeasData_Nt.HorizontalAlignment = 'center';</pre>
1607	<pre>app.EvalResMeasData_Nt.Editable = 'off';</pre>
1608	<pre>%% Create EvalResMeasData_KernelSizeInPercent_Label</pre>
1609	app.EvalResMeasData_KernelSizeInPercent_Label = uilabel(app.
2	EvalResMeasDataParamsGridLavout):

1610	<pre>app.EvalResMeasData_KernelSizeInPercent_Label.Text = 'Kernelsize in \%';</pre>
1611	<pre>app.EvalResMeasData_KernelSizeInPercent_Label.Interpreter = 'latex';</pre>
1612	app.EvalResMeasData_KernelSizeInPercent_Label.HorizontalAlignment = '
	center';
1613	app.EvalResMeasData_KernelSizeInPercent_Label.BackgroundColor =
	<pre>backGroundColorLabel;</pre>
1614	<pre>%% Create EvalResMeasData_KernelSizeInPercent</pre>
1615	app.EvalResMeasData_KernelSizeInPercent = uieditfield(app.
	<pre>EvalResMeasDataParamsGridLayout, 'numeric');</pre>
1616	<pre>app.EvalResMeasData_KernelSizeInPercent.HorizontalAlignment = 'center';</pre>
1617	<pre>app.EvalResMeasData_KernelSizeInPercent.Editable = 'off';</pre>
1618	<pre>%% Create EvalResMeasData_z_0_X_Label</pre>
م 1619	$app.EvalResMeasData_z_0_X_Label = uilabel(app.$
qĝ	<pre>EvalResMeasDataParamsGridLayout);</pre>
1620	<pre>app.EvalResMeasData_z_0_X_Label.Text = '\$\textrm{z}_{\textrm{0x}}\textrm</pre>
× ×	{ in } \mu m\$';
to 1621	<pre>app.EvalResMeasData_z_0_X_Label.Interpreter = 'latex';</pre>
1622	<pre>app.EvalResMeasData_z_0_X_Label.HorizontalAlignment = 'center';</pre>
u 40 1623	<pre>app.EvalResMeasData_z_0_X_Label.BackgroundColor = backGroundColorLabel;</pre>
	%% Create EvalResMeasData_z_0_X
⊇ <u>a</u> 1625	app.EvalResMeasData_ z_0_X = uieditfield(app.
Nie	<pre>EvalResMeasDataParamsGridLayout, 'numeric', 'ValueDisplayFormat', '%.1f</pre>
	');
8 t 1626	<pre>app.EvalResMeasData_z_0_X.HorizontalAlignment = 'center';</pre>
	app.EvalResMeasData_ z_0_X .Editable = 'off';
arb 1628	%% Create EvalResMeasData_z_R_X_Label
1629	app.EvalResMeasData_z_R_X_Label = uilabel(app.
Dipl	<pre>EvalResMeasDataParamsGridLayout);</pre>
aga 1630	<pre>app.EvalResMeasData_z_R_X_Label.Text = '\$\textrm{z}_{\textrm{Rx}}\textrm</pre>
s is	{ in } \mu m\$';
u is 1631	<pre>app.EvalResMeasData_z_R_X_Label.Interpreter = 'latex';</pre>
1632	<pre>app.EvalResMeasData_z_R_X_Label.HorizontalAlignment = 'center';</pre>
1633	<pre>app.EvalResMeasData_z_R_X_Label.BackgroundColor = backGroundColorLabel;</pre>
in 1634	%% Create EvalResMeasData_z_R_X
Ö 🤤 1635	$app.EvalResMeasData_z_R_X = uieditfield(app.$
ckte	EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.1f
nal	');
ab 1636	<pre>app.EvalResMeasData_z_R_X.HorizontalAlignment = 'center';</pre>
t p 1637	<pre>app.EvalResMeasData_z_R_X.Editable = 'off';</pre>
g 0 1638	<pre>%% Create EvalResMeasData_d_0_X_Label</pre>
dd 1639	$app.EvalResMeasData_d_0_X_Label = uilabel(app.$
he a	<pre>EvalResMeasDataParamsGridLayout);</pre>
$\Box \vdash 1640$	<pre>app.EvalResMeasData_d_0_X_Label.Text = '\$\textrm{d}_{\textrm{0x}}\textrm</pre>
<u> </u>	{ in } \mu m\$';
D 1641	<pre>app.EvalResMeasData_d_0_X_Label.Interpreter = 'latex';</pre>
1642 g	app.EvalResMeasData_d_0_X_Label.HorizontalAlignment = 'center';
O # 1643	<pre>app.EvalResMeasData_d_0_X_Label.BackgroundColor = backGroundColorLabel;</pre>
1 644	<pre>%% Create EvalResMeasData_d_0_X</pre>
T 1645	app.EvalResMeasData_d_0_X = uieditfield(app.

	EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.1f
1646	app.FvalResMeasData d 0 X.HorizontalAlignment = 'center':
1647	app Evalues MeasData d $0 \times \text{Editable} = 'off':$
1648	% Create EvalResMeasData theta X Label
1649	ann EvalResMeasData theta X Label = uilabel(ann
1010	EvalResMeasDataParamsGridLavout):
1650	app.EvalResMeasData theta X Label.Text = '\$\Theta {\textrm{x}}
1000	in rad}\$''
1651	app.EvalResMeasData theta X Label.Interpreter = 'latex':
1652	app EvalResMeasData theta X Label HorizontalAlignment = 'center':
1653	app.EvalResMeasData theta X Label.BackgroundColor = backGroundColorLabel
1000	:
1654	% Create EvalResMeasData_theta_X
1655	app.EvalResMeasData theta $X = uieditfield(app.)$
	EvalResMeasDataParamsGridLavout.'numeric'.'ValueDisplayFormat'.'%.6f
	·):
± 1656	<pre>app.EvalResMeasData_theta_X.HorizontalAlignment = 'center';</pre>
1657	<pre>app.EvalResMeasData_theta_X.Editable = 'off';</pre>
1658	% Create EvalResMeasData_M_squared_X_Label
1659	app.EvalResMeasData_M_squared_X_Label = uilabel(app.
MIG	EvalResMeasDataParamsGridLayout);
1 660	<pre>app.EvalResMeasData_M_squared_X_Label.Text = '\$\textrm{M}^{2}_{x</pre>
all	}}\$';
1 661	<pre>app.EvalResMeasData_M_squared_X_Label.Interpreter = 'latex';</pre>
1662	<pre>app.EvalResMeasData_M_squared_X_Label.HorizontalAlignment = 'center';</pre>
P 1663	<pre>app.EvalResMeasData_M_squared_X_Label.BackgroundColor =</pre>
IIIar	<pre>backGroundColorLabel;</pre>
1664	<pre>%% Create EvalResMeasData_M_squared_X</pre>
<u>n</u> 1665	<pre>app.EvalResMeasData_M_squared_X = uieditfield(app.</pre>
C C C C C C C C C C C C C C C C C C C	EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.6f
	');
1666	<pre>app.EvalResMeasData_M_squared_X.HorizontalAlignment = 'center';</pre>
2 1667	<pre>app.EvalResMeasData_M_squared_X.Editable = 'off';</pre>
<u>n</u> 1668	<pre>%% Create EvalResMeasData_z_0_Y_Label</pre>
b 1669	$app.EvalResMeasData_z_0_Y_Label = uilabel(app.$
a	<pre>EvalResMeasDataParamsGridLayout);</pre>
§ 1670	<pre>app.EvalResMeasData_z_0_Y_Label.Text = '\$\textrm{z}_{\textrm{0y}}\textrm</pre>
en	{ in } \mu m\$';
§ 1671	<pre>app.EvalResMeasData_z_0_Y_Label.Interpreter = 'latex';</pre>
1672	<pre>app.EvalResMeasData_z_0_Y_Label.HorizontalAlignment = 'center';</pre>
ני 1673	<pre>app.EvalResMeasData_z_0_Y_Label.BackgroundColor = backGroundColorLabel;</pre>
-1674	%% Create EvalResMeasData_z_0_Y
1675	$app.EvalResMeasData_z_0_Y = uieditfield(app.$
-	EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.1f
e hut	');
b 1676	<pre>app.EvalResMeasData_z_0_Y.HorizontalAlignment = 'center';</pre>
6 1677	app.EvalResMeasData_ z_0_Y .Editable = 'off';
1 678	<pre>%% Create EvalResMeasData_z_R_Y_Label</pre>
_	

<pre>1689 app.EvalResMeasData 2 R Y Label.Text = 's\textrm{2}_{\textrm{Ry}}\textrm { in } \mu m5'; app.EvalResMeasData z R, Y_Label.Interpreter = 'latex'; app.EvalResMeasData z R, Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData z R, Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData z R, Y_Label.BackgroundColor_abel; % Create EvalResMeasData z R, Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData z R, Y_Label.BackgroundColor_abel; % Create EvalResMeasData z R, Y_HorizontalAlignment = 'center'; app.EvalResMeasData z R, Y_Hitable = 'off'; % Create EvalResMeasData z R, Y_Hitable = 'off'; % Create EvalResMeasData d. 0, Y_Label = ulabel(app. EvalResMeasDatad_0, Y_Label.Interpreter = 'latex'; app.EvalResMeasData_d.0, Y_Label.Interpreter = 'latex'; app.EvalResMeasData_d.0, Y_Label.Interpreter = 'latex'; app.EvalResMeasData_d.0, Y_Label.BackgroundColor_abel; % Create EvalResMeasData_d.0, Y_Label.Interpreter = 'latex'; app.EvalResMeasData_d.0, Y_Label.BackgroundColor_abel; % Create EvalResMeasData_d.0, Y_Label.BackgroundColor_abel; % Create EvalResMeasData_d.0, Y_Label.BackgroundColor_backGroundColorLabel; % Create EvalResMeasData_d.0, Y_Label.BackgroundColor = backGroundColorLabel; % Create EvalResMeasData_d.0, Y_Label.BackgroundColor_backGroundColorLabel; % Create EvalResMeasData_d.0, Y_Label.BackgroundColor_backGroundColorLabel; % Create EvalResMeasData_d.0, Y_Label, BackgroundColor_backGroundColorLabel; % Create EvalResMeasData_theta_Y_Label = offf; % Create EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}</pre>	1679	<pre>app.EvalResMeasData_z_R_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout);</pre>
<pre>{ in } 'uu m\$'; app.EvalResMeasData.z.R.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.z.R.Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData.z.R.Y = uiediffield(app. EvalResMeasData.z.R.Y = uiediffield(app. EvalResMeasData.d.0.Y.Label = uilabel(app. EvalResMeasData.d.0.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.d.0.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.d.0.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.d.0.Y.Label.Rextpretex(Color=backGroundColorLabel; % Create EvalResMeasData.d.0.Y = uiediffield(app. EvalResMeasData.d.0.Y = uiediffield(app. EvalResMeasData.d.0.Y.HorizontalAlignment = 'center'; app.EvalResMeasData.d.0.Y.HorizontalAlignment = 'center'; app.EvalResMeasData.d.0.Y.HorizontalAlignment = 'center'; app.EvalResMeasData.d.0.Y.HorizontalAlignment = 'center'; app.EvalResMeasData.d.0.Y.HorizontalAlignment = 'center'; app.EvalResMeasData.d.0.Y.HorizontalAlignment = 'center'; app.EvalResMeasData.theta.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.theta.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.theta.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.theta.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.theta.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.theta.Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData.theta.Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData.theta.Y.Label.HorizontalAlignment = 'center'; app.EvalResMeasData.theta.Y.Editable = 'off'; % Create EvalResMeasData.theta.Y.Label.HorizontalAlignment = 'center'; app.EvalResMeasData.theta.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.theta.Y.Label.Interpreter = 'latex'; app.EvalResMeasData.thesA.S.guared.Y.Label</pre>	1680	app.EvalResMeasData_z_R_Y_Label.Text = '\$\textrm{z}_{\textrm{Ry}}\textrm
<pre>nbbi app.EvalkesMeasData Z. R.Y.Label.HorizontalAlignment = 'center'; app.EvalkesMeasData.Z.R.Y.Label.HorizontalAlignment = 'center'; app.EvalkesMeasData.Z.R.Y.Label.HorizontalAlignment = 'center'; app.EvalkesMeasData.Z.R.Y. e uiddiffield(app. EvalkesMeasData.Z.R.Y.HorizontalAlignment = 'center'; app.EvalkesMeasData.Z.R.Y.HorizontalAlignment = 'center'; app.EvalkesMeasData.Z.R.Y.HorizontalAlignment = 'center'; app.EvalkesMeasData.Z.R.Y.HorizontalAlignment = 'center'; app.EvalkesMeasData.d.O.Y.Label set app.EvalkesMeasData.d.O.Y.Label app.EvalkesMeasData.d.O.Y.Label app.EvalkesMeasData.d.O.Y.Label. app.EvalkesMeasData.d.O.Y.Label.Text = '\$\textrm{d}_{\textrm{0}}\\textrm { in } \mu m\$'; app.EvalkesMeasData.d.O.Y.Label.Interpreter = 'latex'; app.EvalkesMeasData.d.O.Y.Label.HorizontalAlignment = 'center'; app.EvalkesMeasData.d.O.Y.Label.HorizontalAlignment = 'center'; app.EvalkesMeasData.d.O.Y.Label.HorizontalAlignment = 'center'; app.EvalkesMeasData.d.O.Y = uiddiffield(app. EvalkesMeasData.d.O.Y = uiddiffield(app. EvalkesMeasData.d.O.Y.HorizontalAlignment = 'center'; app.EvalkesMeasData.d.O.Y.HorizontalAlignment = 'center'; app.EvalkesMeasData.d.O.Y.Label.HorizontalAlignment = 'center'; app.EvalkesMeasData.d.O.Y.Label.text = '\$\Theta {\textrm{y}} in rad}\$'; app.EvalkesMeasData.theta.Y.Label = uilabel(app. EvalkesMeasData.theta.Y.Label = uilabel(app. EvalkesMeasData.theta.Y.Label.Interpreter = 'latex'; app.EvalkesMeasData.theta.Y.Label.BackgroundColor = backGroundColorLabel ; imp.EvalkesMeasData.theta.Y.Label.BackgroundColor = backGroundColorLabel ; imp.EvalkesMeasData.theta.Y.Label.AckgroundColor = backGroundColorLabel ; imp.EvalkesMeasData.theta.Y.Label.AckgroundColor = backGroundColorLabel ; imp.EvalkesMeasData.theta.Y.Label.AckgroundColor = backGroundColorLabel ; imp.EvalkesMeasData.theta.Y.Label.KakgroundColor = backGroundColorLabel ; imp.EvalkesMeasData.theta.Y.Label.AckgroundColor = backGroundColorLabel ; imp.EvalkesMeasData.theta.Y.Label.text = '\$\textrm{M}^{2} {\textrm{y} }5'; app.EvalkesMeasD</pre>	1.001	{ in } \mu m\$';
<pre>app.EvalmesMeasData_Z.R.Y_Label.BackgroundColor = backGroundColorLabel; app.EvalmesMeasData_Z.R.Y_Label.BackgroundColor = backGroundColorLabel; %% Create EvalResMeasData_Z.R.Y = uieditfield(app. EvalResMeasData_Z.R.Y.Editable = 'off'; app.EvalResMeasData_Z.R.Y.Editable = 'off'; %% Create EvalResMeasData_L.R.Y.Editable = 'off'; %% Create EvalResMeasData_d.0.Y.Label = uilabel(app. EvalResMeasData_d.0.Y.Label = uilabel(app. EvalResMeasData_d.0.Y.Label.Interpreter = 'latex'; app.EvalResMeasData_d.0.Y.Label.Interpreter = 'latex'; app.EvalResMeasData_d.0.Y.Label.Interpreter = 'latex'; app.EvalResMeasData_d.0.Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData_d.0.Y.Label.Interpreter = 'latex'; app.EvalResMeasData_d.0.Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData_d.0.Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData_d.0.Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData_d.0.Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData_d.0.Y.Cititable = uilabel(app. EvalResMeasData_d.0.Y.Cititable = uilabel(app. EvalResMeasData_d.0.Y.Cititable = 'off'; 'app.EvalResMeasData_d.0.Y.Cititable = 'off'; 'app.EvalResMeasData_d.0.Y.Cititable = 'off'; 'app.EvalResMeasData_d.0.Y.Label.Text = '\$\Theta_{\textrm(y}}\textrm(in radj\$'; 'app.EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label.Text = '\$\textrm(M)^{2}_{\textrm(Y})</pre>	1081	app.EvalResMeasData_Z_R_Y_Label.Interpreter = 'latex';
<pre>app_EvalkesMeasData_Z_R.Y_LabeL.Backgroundcolor = backGroundcolorLabel; app_EvalkesMeasData_Z_R.Y = uieditfield(app. EvalkesMeasData_Z_R.Y = uieditfield(app. EvalkesMeasData_Z_R.Y.HorizontalAlignment = 'center'; app_EvalkesMeasData_Z_R.Y.HorizontalAlignment = 'center'; app_EvalkesMeasData_L_R.Y.Editable = 'offf'; set app_EvalkesMeasData_L_0_Y_Label app_EvalkesMeasData_L_0_Y_Label = uilabel(app. EvalkesMeasData_L_0_Y_Label.Text = '\$\textrm{d}_{(\textrm{0})}\textrm { in } \mu m\$'; app_EvalkesMeasData_L_0_Y_Label.Text = '\$\textrm{d}_{(\textrm{0})}\textrm { in } \mu m\$'; app_EvalkesMeasData_L_0_Y_Label.HorizontalAlignment = 'center'; app_EvalkesMeasData_L_0_Y_Label.HorizontalAlignment = 'center'; app_EvalkesMeasData_L_0_Y_Label.BackgroundColor = backGroundColorLabel; % Create EvalkesMeasData_L_0_Y = uieditfield(app. EvalkesMeasData_L_0_Y_Label.BackgroundColor = backGroundColorLabel; app_EvalkesMeasData_L_0_Y.Label.Agayout, 'numeric', 'ValueDisplayFormat', '%.If '); app_EvalkesMeasData_Lo_Y.HorizontalAlignment = 'center'; app_EvalkesMeasData_theta_Y_Label = uilabel(app. EvalkesMeasData_theta_Y_Label = uilabel(app. EvalkesMeasData_theta_Y_Label = uilabel(app. EvalkesMeasData_theta_Y_Label = liket(app. EvalkesMeasData_theta_Y_Label = 's\Theta_{\textrm{y}} in rad}s'; app_EvalkesMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; app_EvalkesMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; app_EvalkesMeasData_theta_Y_Label.Gap. EvalkesMeasData_theta_Y_Label.SackgroundColor = backGroundColorLabel ; app_EvalkesMeasData_theta_Y_Label.Gap. Eval</pre>	1682	app.EvalResMeasData_z_R_Y_Label.HorizontalAlignment = 'center';
<pre>1884</pre>	1683	<pre>app.EvalResMeasData_z_R_Y_Label.BackgroundColor = backGroundColorLabel;</pre>
<pre>1885 app.EvalResMeasData_R_RY = uiediffield(app. EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.1f</pre>	1684	%% Create EvalResMeasData_z_R_Y
<pre>EvalResMeasDataParamsGridLayout, 'numeric', 'ValueDisplayFormat','%.1f '); app.EvalResMeasData_z_R_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_Z_R_Y.Editable = 'off'; % Create EvalResMeasData_d_0_Y.Label = uilabel(app. EvalResMeasData_d_0_Y.Label = uilabel(app. EvalResMeasData_d_0_Y.Label.Text = '\$\textrm{d}_{\textrm{0}}\\textrm { in } Nu m\$'; app.EvalResMeasData_d_0_Y.Label.Text = '\$\textrm{d}_{\textrm{0}}\\textrm { in } Nu m\$'; app.EvalResMeasData_d_0_Y.Label.Interpreter = 'latex'; app.EvalResMeasData_d_0_Y.Label.MorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y.Label.BackgroundColor = backGroundColorLabel; % Create EvalResMeasData_d_0_Y.Editable = 'off'; % Create EvalResMeasData_theta_Y.Label app.EvalResMeasData_d_0_Y.Editable = 'off'; % Create EvalResMeasData_theta_Y.Label app.EvalResMeasData_theta_Y.Label.Text = '\$\textrm{y}\\textrm{ in rad}\$'; 1701 app.EvalResMeasData_theta_Y.Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y.Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label.Text = '\$\textrm{y}\\textrm{ in rad}\$'; 1702 app.EvalResMeasData_theta_Y.Label.BackgroundColor = backGroundColorLabel ; 1704 % Create EvalResMeasData_theta_Y.Label.Resdresdresdresdresdresdresdresdresdresdr</pre>	1685	app.EvalResMeasData_z_R_Y = uieditfield(app.
<pre>1686 app.EvalResMeasData_z_R_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_YLabel sopp.EvalResMeasData_d_0_YLabel = uilabel(app. EvalResMeasData_d_0_YLabel.Text = '\$\textrm{d}_{\textrm{0}}\textrm { in } \mu m\$'; app.EvalResMeasData_d_0_YLabel.Text = '\$\textrm{d}_{\textrm{0}}\textrm app.EvalResMeasData_d_0_YLabel.Text = '\$\textrm{d}_{\textrm{0}}\textrm app.EvalResMeasData_d_0_YLabel.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y Label.MorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y Label.MorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y Label.MorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y Label.Text = '\$\textrm{d}_{}\textrm ','%.ff '); app.EvalResMeasData_d_0_Y = uidditfield(app. EvalResMeasData_d_0_Y.Editable = 'off'; app.EvalResMeasData_d_0_Y.Editable = 'off'; app.EvalResMeasData_theta_YLabel.Text = '\$\textrm{y}\ in rad}\$'; app.EvalResMeasData_theta_YLabel.Text = '\$\textrm{y}\ in rad}\$'; app.EvalResMeasData_theta_YLabel.BackgroundColor = backGroundColorLabel ; in rad}\$'; app.EvalResMeasData_theta_YLabel.Text = '\$\textrm{y}\ in rad}\$'; app.EvalResMeasData_theta_YLabel.BackgroundColor = backGroundColorLabel ; in rad}\$'; app.EvalResMeasData_theta_YLabel.BackgroundColor = backGroundColorLabel ; in cad\$'; app.EvalResMeasData_theta_YLabel.BackgroundColor = backGroundColorLabel ; in cad\$'; app.EvalResMeasData_theta_YLabel.BackgroundColor = backGroundColorLabel ; in cad\$'; app.EvalResMeasData_theta_YLabel.BackgroundColor = backGroundColorLabel ; in cadBesMeasData_theta_YLabel.BackgroundColor = backGroundCo</pre>		EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.1f '):
<pre>1687 app.EvalResMeasData_Z_R_Y_Editable = 'off'; 1688 %% Create EvalResMeasData_d_0_Y_Label 1689 app.EvalResMeasData_d_0_Y_Label = uilabel(app. EvalResMeasData_d_0_Y_Label.Text = '\$\textrm{d}_{\textrm{0}}\textrm { in } \mu m\$; 1690 app.EvalResMeasData_d_0_Y_Label.Text = '\$\textrm{d}_{\textrm{0}}\textrm { in } \mu m\$; 1691 app.EvalResMeasData_d_0_Y_Label.Interpreter = 'latex'; 1692 app.EvalResMeasData_d_0_Y_Label.BackgroundColor = backGroundColorLabel; 1693 app.EvalResMeasData_d_0_Y_Label.BackgroundColor = backGroundColorLabel; 1694 %% Create EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y_Label.BackgroundColor = backGroundColorLabel; 1695 app.EvalResMeasData_d_0_Y_HorizontalAlignment = 'center'; 1696 app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; 1697 app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; 1698 %% Create EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; 1699 app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}}\ 1698 in radjS'; 1700 app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; 1701 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel 1702 ; 1703 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel 1704 %% Create EvalResMeasData_theta_Y 1705 app.EvalResMeasData_theta_Y_Label.BackgroundColor = 'center'; 1706 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y_Label.BackgroundColor = 'center'; 1707 app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label.Text = '\$\textrm{M}^{2}_{2}_{\textrm{y}} 1}_{3}; 1710 app.EvalResMeasData_th_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label = uilabel(app.</pre>	1686	app.EvalResMeasData z R Y.HorizontalAlignment = 'center':
<pre>dp.Function for the function of the funct</pre>	1687	app EvalResMeasData z R Y Editable = 'off':
<pre>105</pre>	1688	%% Create EvalResMeasData d A V Label
<pre>app.FrontesHeasDataParamsGridLayout); app.EvalResMeasData_d_0_Y_Label.Text = '\$\textrm{d}_{\textrm{0}}\textrm { in } \mu m\$'; app.EvalResMeasData_d_0_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_d_0_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y_Label.BackgroundColor = backGroundColorLabel; % Create EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y_Label.BackgroundColor= backGroundColorLabel; '% Create EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y.Editable = 'off'; 'pp.EvalResMeasData_theta_Y_Label app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label. for app.EvalResMeasData_theta_Y_Label. for app.EvalResMeasData_theta_Y_Label. in rad}\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; in rad}\$'; app.EvalResMeasData_theta_Y_Label.RextgroundColor = backGroundColorLabel ; in readSumasData_theta_Y_Label.RextgroundColor = backGroundColorLabel ; in readSumasData_theta_Y_Label.RextgroundColor = backGroundColorLabel ; in readSumasData_theta_Y_Label.RextgroundColor = backGroundColorLabel ; in readSumasData_theta_Y_Label.RextgroundColor = backGroundColorLabel ; in readResMeasData_theta_Y_Label.RextgroundColor = backGroundColorLabel ; in readResMeasData_theta_Y_Label.RextgroundColor = backGroundColorLabel ; in readResMeasData_theta_Y_Label.RextgroundColorLabel ; in p.EvalResMeasData_theta_Y_Label = 'off'; in p.EvalResMeasData_theta_Y_Label = 'off'; in p.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; in p.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; in p.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; in p.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; in p</pre>	1680	ann EvalResMeasData d \cap V Label - uilabel(ann
<pre>app.EvalResMeasData_d_0_Y_Label.Text = '\$\textrm{d}_{\textrm{0y}}\textrm { in } \mu m\$'; app.EvalResMeasData_d_0_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_d_0_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y_Label.BackgroundColor = backGroundColorLabel; i694 %% Create EvalResMeasData_d_0_Y app.EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y_Label.BackgroundColor = backGroundColorLabel; i695 app.EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y_Label.BackgroundColor= to the state sta</pre>	1003	$E_{val} = a_v a_v a_v a_v a_v a_v a_v a_v a_v a_v$
<pre>1950 app.EvalResMeasData_d_O_Y_Label.Text = \$\textIm(dy);{\textIm(in), wu m\$; app.EvalResMeasData_d_O_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_d_O_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_d_O_Y = uiediffield(app. EvalResMeasData_d_O_Y = uiediffield(app. EvalResMeasData_d_O_Y = uiediffield(app. EvalResMeasData_d_O_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_O_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_O_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_O_Y.Editable = 'off'; if app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\textIm(y)}\textIm(in rad)\$'; app.EvalResMeasData_theta_Y_Label.Text = '\$\textIm(y)}\textIm(in rad)\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; if 00 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; if 04 % Create EvalResMeasData_theta_Y app.EvalResMeasData_theta_Y_Label.BackgroundColorLabel ; if 05 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.Eistable = 'off'; if 06 app.EvalResMeasData_theta_Y.Eistable = 'off'; if 07 app.EvalResMeasData_theta_Y.Eistable = 'off'; if 08 % Create EvalResMeasData_theta_Y.Eistable = 'off'; if 08 % Create EvalResMeasData_theta_Y.Eistable = 'off'; if 09 app.EvalResMeasData_theta_Y.Eistable = 'off'; if 00 app.EvalResMeasData_theta_Y.Eistable = 'off'; if 00 app.EvalResMeasData_M.squared_Y.Label = uilabel(app. EvalResMeasData_M.squared_Y.Label = uilabel(app. EvalResMeasData_M.squared_Y.Label.Interpreter = 'latex'; app.EvalResMeasData_M.squared_Y.Label.HorizontalAlignment = 'center'; if 05</pre>	1600	Eval Res Meas Data d. A. Y. Label Text = (d) () text rm (d) () text rm (d) () text rm (d)
<pre>(10) {Wu my; app.EvalResMeasData_d_0_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_d_0_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y.Editable = 'off'; app.EvalResMeasData_d_0_Y.Editable = 'off'; if app.EvalResMeasData_d_0_Y.Editable = 'off'; if app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; if 04 %% Create EvalResMeasData_theta_Y_Label.MorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; if 04 %% Create EvalResMeasData_theta_Y app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.Label.andtignment = 'center'; app.EvalResMeasData_theta_Y.Eitable = 'off'; if 06 app.EvalResMeasData_theta_Y.Eitable = 'off'; if 07 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasDataParamsGridLayout, 'numeric', 'ValueDisplayFormat', '%.6f '); if 07 app.EvalResMeasData_theta_Y.Eitable = 'off'; if 08 %% Create EvalResMeasData_Ms.guared_Y.Label = uilabel(app. EvalResMeasDataParamsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); if 08 app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeasDataMarsGridLayout); app.EvalResMeas</pre>	1090	$app.evalkesmeaspala_u_0_f_label.rext = \frac{1}{2} \frac{1}{$
<pre>1091 app.EvalResMeasData_d_0_Y_Label.HorizontalAlignment = 'center'; 1692 app.EvalResMeasData_d_0_Y_Label.HorizontalAlignment = 'center'; 1694 %% Create EvalResMeasData_d_0_Y = uieditfield(app. 1695 app.EvalResMeasData_d_0_Y = uieditfield(app. 1696 app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; 1697 app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; 1697 app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; 1698 %% Create EvalResMeasData_theta_Y_Label = 'off'; 1698 %% Create EvalResMeasData_theta_Y_Label = uilabel(app. 1699 EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} 1700 app.EvalResMeasData_theta_Y_Label.Fext = '\$\Theta_{\textrm{y}} 1701 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1702 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1703 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1704 %% Create EvalResMeasData_theta_Y 1705 app.EvalResMeasData_theta_Y = uieditfield(app. 1706 EvalResMeasData_theta_Y = uieditfield(app. 1707 app.EvalResMeasData_theta_Y = uieditfield(app. 1708 %% Create EvalResMeasData_theta_Y = uieditfield(app. 1707 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1708 %% Create EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1709 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1709 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. 1709 EvalResMeasData_M_squared_Y_Label = uilabel(app. 1709 EvalResMeasData_M_squared_Y_Label = uilabel(app. 1700 EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}} 1710 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; 1711 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1.001	{ IN } \mu m\$";
<pre>app.EvalResMeasData_d_0_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y_Label.BackgroundColor = backGroundColorLabel; %% Create EvalResMeasData_d_0_Y interfease = 'center'; app.EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y.Editable = 'off'; interfease = 'center'; app.EvalResMeasData_d_0_Y.Editable = 'off'; interfease = 'statesMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.BackgroundColorLabel ; in rad}\$'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; in rad\$ in rad\$</pre>	1691	app.EvalResMeasData_d_0_Y_Label.Interpreter = 'latex';
<pre>1693 app.EvalResMeasData_d_0_Y_Label.BackgroundColor = backGroundColorLabel; 1694 % Create EvalResMeasData_d_0_Y 1695 app.EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData@d_0_Y.HorizontalAlignment = 'center'; 1697 app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; 1698 % Create EvalResMeasData_theta_Y_Label 1699 app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; 1700 app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; 1702 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1703 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 % Create EvalResMeasData_theta_Y 1705 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1706 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.Editable = 'off'; 1706 app.EvalResMeasData_theta_Y.Editable = 'off'; 1707 app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 % Create EvalResMeasData_theta_Y.Editable = 'off'; 1709 app.EvalResMeasData_theta_Y.Editable = 'off'; 1709 app.EvalResMeasData_theta_Y.Label.Text = '\$\textrm{M}^{2}_{1}\textrm{y} }}; 1710 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{1}\textrm{y} }}; 1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1692	app.EvalResMeasData_d_0_Y_Label.HorizontalAlignment = 'center';
<pre>1694 1695 1695 app.EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y.Editable = 'off'; 1697 app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 % Create EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y_Label.SackgroundColor = '\atex'; app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.Editable = 'off'; 1705 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label.atex EvalResMeasData_theta_Y.Label.atex i not sploy = 'stalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.Label = 'off'; 1706 app.EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label = 'off'; 1707 app.EvalResMeasData_M_squared_YLabel = uilabel(app. EvalResMeasData_M_squared_YLabel = uilabel(app. EvalResMeasData_M_squared_YLabel.Text = '\$\textrm{M}^{2}_{textrm{y}} }\$'; 1710 app.EvalResMeasData_M_squared_YLabel.Interpreter = 'latex'; app.EvalResMeasData_M_squared_YLabel.Interpreter = 'latex'; app.EvalResMeasData_M_squared_YLabel.HorizontalAlignment = 'center'; 1712 app.EvalResMeasData_M_squared_YLabel.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_YLabel.Interpreter = 'latex'; app.EvalResMeasData_M_squared_YLabel.HorizontalAlignment = 'center'; 1712 app.EvalResMeasData_M_squared_YLabel.HorizontalAlignment = 'center'; 1712 app.EvalResMeasData_M_squared_YLabel.HorizontalAlignment = 'center'; 1712 app.EvalResMeasData_M_squared_YLabel.HorizontalAlignment = 'center'; 1712 app.EvalResMeasData_M_s</pre>	1693	<pre>app.EvalResMeasData_d_0_Y_Label.BackgroundColor = backGroundColorLabel;</pre>
<pre>1695 app.EvalResMeasData_d_0_Y = uieditfield(app. EvalResMeasDataParamsGridLayout, 'numeric', 'ValueDisplayFormat', '%.1f '); 1696 app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; 1697 app.EvalResMeasData_d_0_Y.Editable = 'off'; 1698 % Create EvalResMeasData_theta_Y_Label 1699 app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; 1700 app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1702 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 % Create EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 % Create EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_TaparamsGridLayout); app.EvalResMeasData_M_squared_YLabel = uilabel(app. EvalResMeasData_M_squared_YLabel.Text = '\$\textrm{M}^{2}_{\textrm{y}}; }\$; 1710 app.EvalResMeasData_M_squared_YLabel.Interpreter = 'latex'; app.EvalResMeasData_M_squared_YLabel.Interpreter = 'latex'; app.EvalResMeasData_M_squared_YLabel.HorizontalAlignment = 'center'; 1712 app.EvalResMeasData_M_squared_YLabel.HorizontalAlignment = 'center';</pre>	1694	%% Create EvalResMeasData_d_0_Y
<pre>EvalResMeasDataParamsGridLayout, 'numeric', 'ValueDisplayFormat', '%.lf '); 1696 app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y.Editable = 'off'; 1699 app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}}\textrm{ in rad}\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 %% Create EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y_Lobel.Alignment = 'center'; app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label = 'off'; %% Create EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label = 'off'; %% Create EvalResMeasData_theta_Y.Label = 'off'; %% Create EvalResMeasData_Squared_YLLabel = 'off'; %% Create EvalResMeasData_M_squared_YLLabel = 'off'; %% CreatResMeasData_M_squared_YLLabel.Text = '\$\textrm{M}^{2}_{2-{\textrm{y}}}; %; %; %; %; %; %; %; %; %; %; %; %; %;</pre>	1695	app.EvalResMeasData_d_0_Y = uieditfield(app.
<pre>'); app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y.Editable = 'off'; if app.EvalResMeasData_theta_Y_Label app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; in rad}\$': app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = '\format','%.6f '); app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label = 'off'; infof app.EvalResMeasData_theta_Y.Label = 'off'; infof app.EvalResMeasData_theta_Y.Label = 'uilabel(app. EvalResMeasData_theta_Y.Label = 'uilabel(app. EvalResMeasData_msGridLayout); app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_msGridLayout); infin app.EvalResMeasData_M_squared_Y_Label = '\textrm{M}^{2}_{2}_{textrm{y}}; ifin app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>		EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.1f
<pre>i696 app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_d_0_Y.Editable = 'off'; i698 % Create EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; i704 %% Create EvalResMeasData_theta_Y i705 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; i706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label = 'off'; i708 % Create EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasData_theta_Y.Label = uilabel(app. EvalResMeasDataParamsGridLayout); app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment</pre>		');
<pre>1697 app.EvalResMeasData_d_0_Y.Editable = 'off'; 1698 % Create EvalResMeasData_theta_Y_Label 1699 app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; 1700 app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; 1702 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1703 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 %% Create EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1707 app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 %% Create EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1710 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1710 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}; 1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1696	<pre>app.EvalResMeasData_d_0_Y.HorizontalAlignment = 'center';</pre>
<pre>1698 % Create EvalResMeasData_theta_Y_Label 1699 app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1700 app.EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; 1701 app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; 1702 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1703 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 % Create EvalResMeasData_theta_Y 1705 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1707 app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 % Create EvalResMeasData_theta_Y.Editable = 'off'; 1709 app.EvalResMeasData_M_squared_Y_Label 1709 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}; 1710 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1711 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1697	<pre>app.EvalResMeasData_d_0_Y.Editable = 'off';</pre>
<pre>1699 app.EvalResMeasData_theta_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1700 app.EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; 1701 app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; 1702 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1703 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 %% Create EvalResMeasData_theta_Y 1705 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1707 app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 %% Create EvalResMeasData_M_squared_Y_Label 1709 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1710 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\}\$; 1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1698	<pre>%% Create EvalResMeasData_theta_Y_Label</pre>
EvalResMeasDataParamsGridLayout); app.EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}} in rad}\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 %% Create EvalResMeasData_theta_Y app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Editable = 'offf'; 1708 %% Create EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}; 1710 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; 1711 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';	1699	<pre>app.EvalResMeasData_theta_Y_Label = uilabel(app.</pre>
<pre>1700 app.EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}}</pre>		<pre>EvalResMeasDataParamsGridLayout);</pre>
<pre>in rad}\$'; app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; introd %% Create EvalResMeasData_theta_Y app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.6f '); introd app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Editable = 'off'; %% Create EvalResMeasData_M_squared_Y_Label app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\{\textrm{y}}}; introd app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1700	<pre>app.EvalResMeasData_theta_Y_Label.Text = '\$\Theta_{\textrm{y}}</pre>
<pre>1701 app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex'; 1702 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1703 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 %% Create EvalResMeasData_theta_Y 1705 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.6f '); 1706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1707 app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 % Create EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1710 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}; 1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>		<pre>in rad}\$';</pre>
<pre>1702 app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center'; 1703 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 %% Create EvalResMeasData_theta_Y 1705 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasDataParamsGridLayout, 'numeric', 'ValueDisplayFormat', '%.6f '); 1706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1707 app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 %% Create EvalResMeasData_M_squared_Y_Label 1709 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1710 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}; 1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1701	<pre>app.EvalResMeasData_theta_Y_Label.Interpreter = 'latex';</pre>
<pre>1703 app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel ; 1704 %% Create EvalResMeasData_theta_Y 1705 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasDataParamsGridLayout, 'numeric', 'ValueDisplayFormat', '%.6f '); 1706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1707 app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 %% Create EvalResMeasData_M_squared_Y_Label 1709 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{L}{\textrm{y}}; 1710 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1702	<pre>app.EvalResMeasData_theta_Y_Label.HorizontalAlignment = 'center';</pre>
<pre>int ; if it is it is it is it is it is it it is i</pre>	1703	<pre>app.EvalResMeasData_theta_Y_Label.BackgroundColor = backGroundColorLabel</pre>
<pre>1704 %% Create EvalResMeasData_theta_Y 1705 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasDataParamsGridLayout, 'numeric', 'ValueDisplayFormat', '%.6f '); 1706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; 1707 app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 %% Create EvalResMeasData_M_squared_Y_Label 1709 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1710 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}}; 1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>		;
<pre>21705 app.EvalResMeasData_theta_Y = uieditfield(app. EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.6f '); 21706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 % Create EvalResMeasData_M_squared_Y_Label 1709 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1710 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}}; 1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1704	% Create EvalResMeasData_theta_Y
EvalResMeasDataParamsGridLayout, 'numeric', 'ValueDisplayFormat', '%.6f '); app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Editable = 'off'; %% Create EvalResMeasData_M_squared_Y_Label app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}} }\$'; app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';	1705	app.EvalResMeasData_theta_Y = uieditfield(app.
<pre>'); app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Editable = 'off'; '% Create EvalResMeasData_M_squared_Y_Label app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}; '' '' '''''''''''''''''''''''''''''</pre>		EvalResMeasDataParamsGridLavout.'numeric'.'ValueDisplayFormat'.'%.6f
<pre>x1706 app.EvalResMeasData_theta_Y.HorizontalAlignment = 'center'; app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 %% Create EvalResMeasData_M_squared_Y_Label 1709 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1710 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}} }; 1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>		'):
<pre>1700 1707 app.EvalResMeasData_theta_Y.Editable = 'off'; 1708 %% Create EvalResMeasData_M_squared_Y_Label = uilabel(app. 1709 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\\textrm{y}}; 1710 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1711 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center'; 1712</pre>	1706	ann EvalResMeasData theta Y HorizontalAlignment = 'center':
<pre>1701 dpp:EvalueShedSbata_Hierd_FieldEddet = off, 1708 1709 app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout); 1710 app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}; 1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1707	apprevalues $Particular and Particular and Par$
1700app.EvalResMeasData_M_squared_Y_Label = uilabel(app. EvalResMeasDataParamsGridLayout);1710app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\\textrm{y}};1711app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex';1712app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';	1708	%% Create EvalResMeasData M squared Y Label
InterpretentionEvalResMeasDataParamsGridLayout);1710app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\\textrm{y}}};1711app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex';1712app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';	1700	ann EvalResMeasData M squared Y Label = uilabel(ann
1710app.EvalResMeasData_M_squared_Y_Label.Text = '\$\textrm{M}^{2}_{\textrm{y}}1711app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex';1712app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';	1100	$E_{val} = arcuset - arcuset (app: E_val_Rescaled arcuset arcuset)$
app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex';17111712app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';	1710	Eval (Restrict a subject of the analytic of
<pre>1711 app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex'; 1712 app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';</pre>	1110	$app.cvalkesmeasuala_m_squareu_f_Label.rext = \frac{1}{2} \frac{1}{2}$
1/11app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex';1712app.EvalResMeasData_M_squared_Y_Label.HorizontalAlignment = 'center';	1/711	<pre>}}*;</pre>
app.Evalkesmeasuata_M_squared_Y_Label.HorizontalAlignment = 'center';	1/11	app.EvalResMeasData_M_squared_Y_Label.Interpreter = 'latex';
	1(12	app.cvalkesmeasuala_m_squared_r_Label.HorizontalAlignment = `center';

1719	app EvalDesMeasDate M squared V Label PaskgroundColor -
1110	app.Evaluesheasbala_h_squared_t_labet.backgroundcotor =
1	backGroundLolorLabel;
1714	%% Create EvalResMeasData_M_squared_Y
1715	app.EvalResMeasData_M_squared_Y = uieditfield(app.
	EvalResMeasDataParamsGridLayout,'numeric','ValueDisplayFormat','%.6f
	');
1716	<pre>app.EvalResMeasData_M_squared_Y.HorizontalAlignment = 'center';</pre>
1717	<pre>app.EvalResMeasData_M_squared_Y.Editable = 'off';</pre>
1718	<pre>%% Create EvalResMeasData_M_squared_eff_Label</pre>
1719	<pre>app.EvalResMeasData_M_squared_eff_Label = uilabel(app.</pre>
	<pre>EvalResMeasDataParamsGridLayout);</pre>
1720	<pre>app.EvalResMeasData_M_squared_eff_Label.Text = '\$\textrm{M}^{2}_{\textrm</pre>
	{eff}}\$':
1721	app.EvalResMeasData_M_squared_eff_Label.Interpreter = 'latex':
1722	app.EvalResMeasData M squared eff Label.HorizontalAlignment = 'center':
1723	app EvalResMeasData M squared eff Label BackgroundColor =
1120	hackGroundColorLabel:
1794	%% Create EvalResMassData M squared off
1724	ann EvalPorMoasData M cquared off - uioditfield(ann
	app.EvalResMeasData_r_squared_err = uredriftetu(app.
	EvaluesheasbalararamisoriuLayout, humeric, valuebisplayrormat, %.or
1796);
≥ 1 <i>12</i> 0	app.EvalResMeasData_M_squared_eff.HorizontalAlignment = "center";
$\frac{1}{2}$ 1/2/	app.EvalkesMeasData_M_squared_ett.Editable = 'ott';
g 1728	
1729	<pre>%% Create EvalResMeasDataResultDropDown</pre>
1730	app.EvalResPlotChooserDropDown = uidropdown(app.
חם	<pre>EvalResMeasDataGridLayout);</pre>
1731 g	<pre>app.EvalResPlotChooserDropDown.Layout.Row = 5;</pre>
≥ 1732	<pre>app.EvalResPlotChooserDropDown.Layout.Column = 1;</pre>
<u>^</u> 1733	<pre>app.EvalResPlotChooserDropDown.FontSize = app.fontsize14;</pre>
§ 1734	<pre>app.EvalResPlotChooserDropDown.Items = {'Plot Radius X of all MP'</pre>
1735	,'Plot Radius Y of all MP'
1736	,'Plot 2D Intensity Distribution X of selected MP'
2 1737	,'Plot 2D Intensity Distribution Y of selected MP'
1738	, 'Plot 3D Intensity Distribution of selected MP'
1739	, 'Plot Spatial Spectrum of selected MP'
1740 פ	, 'Plot 2D Spectral Distribution X of selected MP'
1 741	'Plot 2D Spectral Distribution Y of selected MP'
b 1742	'Plot Close Up of Beamprofile'
1743	Plot Diameter X squared Datapoints with fitted curve'
1744	'Plot Diameter Y squared Datapoints with fitted curve'}
v 1745	%% Create EvalResMeasPlotDivYButton
= 1740	ann EvalRecMeasPlotButton $-$ uibutton (ann EvalRecMeasDataGridLavout):
1740	apprevalues received to button $=$ urbatton (apprevalues reasonated) rulayout), ann EvalRecMeasPlotRutton Layout Row = 5.
1740	app. Eval RecMaps D at Puttan Layout, following -3 ,
1748 9 1740	app.EvalResmeasPlotButton.Layout.Column = 2;
	app.EvalResmeasPlotButton.Text = 'Plot Selection';
175U	app.EvalResMeasPlotButton.FontSize = app.fontsize14;
б 1751	app.EvalResMeasPlotButton.ButtonPushedFcn = @(src, event)
Your	<pre>EvalResMeasPlotButton_ButtonPushedFcn(app, src, event);</pre>

1753app.EvalResSaveMeasButton = uibutton(app.EvalResMeasDataGridLayout);1754app.EvalResSaveMeasButton.Layout.Column = 1;1755app.EvalResSaveMeasButton.Text = 'Save';1756app.EvalResSaveMeasButton.FontSize = app.fontsizel4;1757app.EvalResSaveMeasButton.ButtonPushedFcn = @(src.event);1758EvalResSaveMeasButton.ButtonPushedFcn = @(src.event);1759% Create EvalResExportMeasButton.Cayout.Column = 2;1760app.EvalResExportMeasButton.Layout.Column = 2;1761app.EvalResExportMeasButton.Cayout.Column = 2;1763app.EvalResExportMeasButton.Cayout.Column = 2;1764app.EvalResExportMeasButton.Cayout.Column = 1;1765app.EvalResExportMeasButton.Cayout.Column = 1;1766% Create EvalResHoldPlotButton1767app.EvalResHoldPlotButton.Layout.Column = 1;1768app.EvalResHoldPlotButton.Layout.Column = 1;1779app.EvalResHoldPlotButton.Layout.Row = 4;1770app.EvalResHoldPlotButton.Layout.Row = 4;1771app.EvalResSmpDataPanel1773app.EvalResSmpDataFanel1774app.EvalResSmpDataGridLayout = uigridLayout(app.EvalResSmpDataPanel);1775app.EvalResSmpDataGridLayout = columnition = {'li', 'li'};1776app.EvalResSmpDataGridLayout = uigridLayout.Row = 1;1777mapp.EvalResSmpDataDropDownLabel = uilabel(app.EvalResSmpDataGridLayout);1776app.EvalResSmpDataDropDownLabel = uilabel(app.EvalResSmpDataGridLayout);1776app.EvalResSmpDataDropDownLabel.Layout.Row = 1;1777mapp.EvalResSmpDataDropDownLab	1752	%% Create EvalResSaveMeasButton
<pre>1734 app.EvalResSaveMeasButton.Layout.Row = 6; 1735 app.EvalResSaveMeasButton.Layout.column = 1; 1737 app.EvalResSaveMeasButton.FontSize = app.fontSizel4; 1738 app.EvalResSaveMeasButton.ButtonPushedErcn(app.src.event) EvalResSaveMeasButton ButtonPushedErcn(app.src.event); 1740 app.EvalResExportMeasButton = ubiutton(app.EvalResMeasDataGridLayout); 1750 app.EvalResExportMeasButton.Layout.column = 2; 1763 app.EvalResExportMeasButton.Layout.column = 2; 1764 app.EvalResExportMeasButton.FontSize = app.fontSizel4; 1775 app.EvalResExportMeasButton.FontSize = app.fontSizel4; 1776 app.EvalResExportMeasButton.FontSize = app.fontSizel4; 1776 app.EvalResExportMeasButton.Layout.column = 2; 1777 app.EvalResExportMeasButton.Layout.column = 1; 1778 app.EvalResExportMeasButton.Layout.column = 1; 1779 app.EvalResExportMeasButton.Layout.colum = 1; 1770 app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' 1771 spt.EvalResHoldPlotButton.Layout.colum = 1; 1772 ww.Create EvalResSmpDataPanel 1775 app.EvalResHoldPlotButton.FontSize = app.fontSizel4; 1776 app.EvalResSmpDataGridLayout = uigridLayout(app.EvalResSmpDataPanel); 1776 app.EvalResSmpDataGridLayout.columnWidth = ('lx','lx'); 1778 ww.Create EvalResSmpDataGridLayout, ColumnWidth = ('lx','lx'); 1779 app.EvalResSmpDataGridLayout.Label = uilabel(app.EvalResSmpDataGridLayout); 1779 app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout); 1779 app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout); 1780 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1781 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1782 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1783 app.EvalResSmpDataDropDown_Label.FontSize = app.fontSizel4; 1784 app.EvalResSmpDataDropDown_Label.FontSize = app.fontSizel4; 1785 app.EvalResSmpDataDropDown_Label.FontSize = app.fontSizel4; 1786 app.EvalResSmpDataDropDown_Label.FontSize = app.fontSizel4; 1787 app.EvalResSmpDataDropDown_Label.Interpret = 'teater'; 1788 app.EvalResSmpDataDropDown_Label.Interpret = '</pre>	1753	app.EvalResSaveMeasButton = $uibutton(app.EvalResMeasDataGridlavout)$:
<pre>inf35 app.EvalResSaveMeasButton.Layout.column = 1; app.EvalResSaveMeasButton.Text = 'Save'; app.EvalResSaveMeasButton.ButtonPushedFcn = @(src.event) EvalResSaveMeasButton.ButtonPushedFcn = @(src.event) EvalResSaveMeasButton.ButtonPushedFcn(app,src.event); www.creatEvalResExportMeasButton app.EvalResExportMeasButton.Layout.Row = 6; app.EvalResExportMeasButton.Layout.Row = 6; app.EvalResExportMeasButton.Instr = 'Export Report'; app.EvalResExportMeasButton.FontSize = app.fontsizel4; app.EvalResExportMeasButton.FontSize = app.fontsizel4; app.EvalResExportMeasButton.FontSize = app.fontsizel4; app.EvalResExportMeasButton.Layout.Row = 6; app.EvalResExportMeasButton.Instr = 'export Report'; app.EvalResExportMeasButton.Instr = 'app.fontsizel4; app.EvalResExportMeasButton.Instr = 'app.fontsizel4; app.EvalResExportMeasButton.Instr = 'app.fontsizel4; app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' state'); app.EvalResHoldPlotButton.Layout.Column = 1; app.EvalResHoldPlotButton.Layout.Column = 1; app.EvalResHoldPlotButton.Instr = 'Hold Data'; app.EvalResHoldPlotButton.Instr = 'app.fontsizel4; ************************************</pre>	1754	apprendices SaveMeasButton Layout $Row = 6$:
<pre>app.EvalResSaveMeasButton.ExpG:field() app.EvalResSaveMeasButton.FortSize = app.fontSize14; app.EvalResSaveMeasButton.ButtonPushedFcn = @(src.event) EvalResSaveMeasButton.ButtonPushedFcn(app.src.event); % Create EvalResExportMeasButton = uibutton(app.EvalResMeasDataGridLayout); app.EvalResExportMeasButton.Layout.Row = 6; app.EvalResExportMeasButton.Icxt = 'Export Report'; app.EvalResExportMeasButton.Icxt = 'Hold Data'; app.EvalResHoldPlotButton.Icxt = 'Hold Data'; app.EvalResHoldPlotButton.Text = 'Hold Data'; app.EvalResHoldPlotButton.Text = 'Hold Data'; app.EvalResHoldPlotButton.Text = 'Hold Data'; app.EvalResSmpDataGridLayout.RowHeight = (3), 'Lx', '66); app.EvalResSmpDataGridLayout.Lobel.Latter = 'Measuring Point No.'; app.EvalResSmpDataGridLayout.Lobel.Latter = 'Measuring Point No.'; app.EvalResSmpDataGridDorpDown.Label.HortzontaLalignment = 'catter'; app.EvalResSmpDataGridDorpDown.Label.HortzontaLal</pre>	1755	app.EvallesSaveMeasButten Layout Column = 1;
<pre>app.EvalResSaveMeasButton.Fort=Jave , app.EvalResSaveMeasButton.FortSize = app.fortsize14; app.EvalResSaveMeasButton.ButtonPushedFcn = @(src.event) EvalResSaveMeasButton.ButtonPushedFcn(app,src.event); Wereate EvalResExportMeasButton = uibutton(app.EvalResMeasDataGridLayout); app.EvalResExportMeasButton.Layout.Row = 6; app.EvalResExportMeasButton.Instrie = app.fortsize14; app.EvalResExportMeasButton.FortSize = app.fortsize14; app.EvalResExportMeasButton.FortSize = app.fortsize14; app.EvalResExportMeasButton.FortSize = app.fortsize14; app.EvalResExportMeasButton.FortSize = app.fortsize14; app.EvalResExportMeasButton.Visible = 'off'; % Create EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' state'); rom app.EvalResHoldPlotButton.Layout.Column = 1; app.EvalResHoldPlotButton.Layout.Column = 1; app.EvalResHoldPlotButton.Layout.Column = 1; app.EvalResHoldPlotButton.Text = 'Hold Data'; app.EvalResHoldPlotButton.Text = 'Hold Data'; app.EvalResHoldPlotButton.TortSize = app.fortsize14; % Create EvalResSmpDataPanel app.EvalResSmpDataGridLayout.ColummWidth = {1x','1x'}; app.EvalResSmpDataGridLayout.ColummWidth = {1x', '1x'}; app.EvalResSmpDataGridLayout.ColummWidth = {1x', '1x'}; app.EvalResSmpDataGridLayout.Column = 1; app.EvalResSmpDataGridLayout.Column = 1; app.EvalResSmpDataGridLayout.Column = 1; app.EvalResSmpDataGridLayout.Column = 1; app.EvalResSmpDataGridLayout.Label.tayout.Column = 1; app.EvalResSmpDataGridDropDown.Label.ayout.Column = 1; app.EvalResSmpDataGridDropDown.Label.FortSize = app.fortsize14; app.EvalResSmpDataGridDropDown.Label.FortSize = app.fortsize14; app.EvalResSmpDataGridDropDown.Label.FortSize = app.fortsize14; app.EvalResSmpDataBropDown.Label.FortSize = app.fortsize14; app.EvalResSmpDataBropDown.Label.FortSize = app.fortsize14; app.EvalResSmpDataBropDown.Label.FortSize = app.fortsize14; app.EvalResSmpDataBropDown.Label.FortSize = app.fortsize14; app.EvalResSmpDataBropDown.Label.FortSize = app.fortsize14; app.EvalResSmpDataBropDown.Label.fortSize = app.fortsize14; app.EvalResSm</pre>	1756	app.EvalResSaveMeasButton.EayOut.Cotumn = 1,
<pre>100 app.EvalResSaveHeasButton.FundSize = app.fontSize14; app.EvalResSaveHeasButton.ButtonPushedFon = @(src.event) EvalResSaveHeasButton = uibutton(app.src.event); W Create EvalResExportHeasButton = uibutton(app.EvalResMeasDataGridLayout); app.EvalResExportHeasButton.Layout.Row = 6; app.EvalResExportHeasButton.Layout.Row = 6; app.EvalResExportHeasButton.Layout.Column = 2; app.EvalResExportHeasButton.Text = 'Export Report'; app.EvalResExportHeasButton.FontSize = app.fontsize14; app.EvalResExportHeasButton.Visible = 'off'; % Create EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' state'); rstate'); app.EvalResHoldPlotButton.Layout.Column = 1; app.EvalResHoldPlotButton.Layout.Column = 1; app.EvalResHoldPlotButton.FontSize = app.fontsize14; app.EvalResHoldPlotButton.FontSize = app.fontsize14; app.EvalResHoldPlotButton.FontSize = app.fontsize14; app.EvalResHoldPlotButton.FontSize = app.fontsize14; app.EvalResHoldPlotButton.FontSize = app.fontsize14; app.EvalResSmpDataPanel = uipanel(app.EvalResTabGridLayout); app.EvalResSmpDataGridLayout.RowHeight = {30, 'lx',60}; app.EvalResSmpDataGridLayout.ColummWidth = {'lx', 'lx'}; app.EvalResSmpDataGridLayout.ColummVidth = {'lx', 'lx'}; app.EvalResSmpDataGridLayout.ColummVidth = {'lx', 'lx'}; app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout); ; app.EvalResSmpDataDropDown_Label.Layout.Row = 1; app.EvalResSmpDataDropDown_Label.Layout.Row = 1; app.EvalResSmpDataDropDown_Label.Layout.Row = 1; app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; % Create EvalResSmpDataDropDown_Label.Interpreter = 'latex'; % Create EvalResSmpDataDropDown_Label.Interpreter = 'latex'; % Create EvalResSmpDataDropDown_Label.Interpreter = 'latex'; % Create EvalResSmpDataDropDown_Label.Rows = 1; app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; % Create EvalResSmpDataDropDown_Label.Rows = 1; app.EvalResSmpDataDropDown_Label.AngedF(napp, src, event); FvalResSmpDataDropDown_Label.AngedF(napp, src, event); FvalResSmpDataDropDown_Label.AngedF(napp, src, even</pre>	1750	app.EvalResSaveMeasButton.Text - Save ,
1735app.EvalkesSavPreasButton.ButtonPushedFcn(app,src.event);1750FualResSavPortMeasButton1760app.EvalResExportMeasButton.Layout.Row = 6;1761app.EvalResExportMeasButton.Layout.Row = 6;1762app.EvalResExportMeasButton.Layout.Row = 6;1763app.EvalResExportMeasButton.Layout.Column = 2;1764app.EvalResExportMeasButton.Text = 'Export Report';1765app.EvalResExportMeasButton.Visible = 'off';1766% Create EvalResSuptonVisible = 'off';1767app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' state');1768app.EvalResHoldPlotButton.Layout.Column = 1;1770app.EvalResHoldPlotButton.Instize = app.fontsizel4;1771app.EvalResHoldPlotButton.Instize = app.fontsizel4;1772% Create EvalResSmpDataGridLayout, ColumNidth = {'lx','lx','spint'; app.EvalResSmpDataGridLayout.ColumNidth = {'lx','lx','spint'; app.EvalResSmpDataGridLayout.ColumNidth = {'lx','lx','spint'; app.EvalResSmpDataGridLayout.ColumNidth = {'lx','lx','spint'; app.EvalResSmpDataGridLayout.ColumNidth = {'lx','lx','spint'; app.EvalResSmpDataGridDopDown_Label.Layout.Row = 1;1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1781app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1782% Create EvalReSSmpDataDropDown_Label.Interpreter = 'latex';1783app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1774app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1775app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1784app.EvalResSmpDataDropDown_Label.SecondColor = backGroundColorLabel;<	1750	app.EvalResSaveMeasButton.FontSize = app.TontSize14;
<pre>Transmitter is a set of the set of the</pre>	1100	app.EvalResSaveMeasButton.ButtonPushedFor(and and such that a second sec
<pre>1699</pre>	1750	EvalResSaveMeasButton_ButtonPushedFch(app,src,event);
1600app.EvalkesExportMeasButton = ubutton(app.EvalKesMeasDataGridLayout);1761app.EvalkesExportMeasButton.Layout.column = 2;1763app.EvalkesExportMeasButton.FontSize = app.fontsizel4;1764app.EvalkesExportMeasButton.FontSize = app.fontsizel4;1765app.EvalkesExportMeasButton.Visible = 'off';1766%% Create EvalkesHoldPlotButton1767app.EvalkesHoldPlotButton.Layout.Row = 4;1768app.EvalkesHoldPlotButton.Layout.Row = 4;1770app.EvalkesHoldPlotButton.Layout.Row = 4;1770app.EvalkesHoldPlotButton.Text = 'Hold Data';1771app.EvalkesHoldPlotButton.FontSize = app.fontsizel4;1772%% Create EvalkesSmpDataPanel1773app.EvalkesSmpDataGridLayout.Rowledit = {30, 'ix', 60);1774app.EvalkesSmpDataGridLayout.Rowledit = {30, 'ix', 60);1775app.EvalkesSmpDataDropDown_Label1779app.EvalkesSmpDataDropDown_Label1779app.EvalkesSmpDataDropDown_Label.Layout.Rowledit.app.entsizel4;1779app.EvalkesSmpDataDropDown_Label.Layout.Rowledit.app.fontsizel4;1779app.EvalkesSmpDataDropDown_Label.Layout.Rowledit.app.fontsizel4;1781app.EvalkesSmpDataDropDown_Label.Layout.Rowledit.app.fontsizel4;1782app.EvalkesSmpDataDropDown_Label.Layout.Rowledit.app.fontsizel4;1784app.EvalkesSmpDataDropDown_Label.BackgroundColor = backKorundColorLabel;1785app.EvalkesSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalkeSmpDataDropDown_Label.Interpreter = 'latex';1787app.EvalkesSmpDataDropDown_Label.Interpreter =	1759	%% Create EvalResExportMeasButton
<pre>1001 app.EvalResExportMeasButton.Layout.Row = 6; 1762 app.EvalResExportMeasButton.Layout.Column = 2; 1763 app.EvalResExportMeasButton.Text = 'Export Report'; 1764 app.EvalResExportMeasButton.Visible = 'off'; 1765 app.EvalResExportMeasButton.Visible = 'off'; 1766 %* Create EvalResHoldPlotButton 1767 app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' 1770 state'); 1768 app.EvalResHoldPlotButton.Layout.Column = 1; 1770 app.EvalResHoldPlotButton.Text = 'Hold Data'; 1770 app.EvalResHoldPlotButton.Text = 'Hold Data'; 1771 app.EvalResHoldPlotButton.Text = 'Hold Data'; 1772 %* Create EvalResSmpDataPanel 1773 app.EvalResSmpDataGridLayout = uigridLayout(app.EvalResSmpDataPanel); 1775 app.EvalResSmpDataGridLayout = uigridLayout(app.EvalResSmpDataPanel); 1776 app.EvalResSmpDataGridLayout.ColumnWidth = {'lx','lx'}; 1777 %* Create EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout) 1778 app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout) 1779 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1780 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1781 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1782 app.EvalResSmpDataDropDown_Label.ScategroundColorLabel; 1783 app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center'; 1784 app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center'; 1785 app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center'; 1786 %* Create EvaResSmpDataDropDown_Label.Mayout.Column = 2; 1780 app.EvalResSmpDataDropDown_Label.Mayout.Column = 2; 1780 app.EvalResSmpDataDropDown_Label.Scates = app.fontsize14; 1784 app.EvalResSmpDataDropDown_Label.Mayout.Column = 2; 1799 app.EvalResSmpDataDropDown_Label.Mayout.Column = 2; 1790 app.EvalResSmpDataDropDown_Label.Mayout.Column = 2; 1790 app.EvalResSmpDataDropDown_Label.Mayout.Column = 2; 1790 app.EvalResSmpDataDropDown_Label.Mayout.Column = 2; 1790 app.EvalResSmpDataDropDown_Label.Mayout.Column = 2; 1791 app.EvalResSmpDataDropDown_Label.Mayout.Column = 2; 1792 %* Create EvaResSmp</pre>	1760	app.EvalResExportMeasButton = ulbutton(app.EvalResMeasDataGridLayout);
<pre>app.EvalResExportMeasButton.Layout.Column = 2; app.EvalResExportMeasButton.Text = 'Export Report'; app.EvalResExportMeasButton.Text = 'Export Report'; app.EvalResExportMeasButton.Visible = 'off'; %% Create EvalResHoldPlotButton app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' state'); app.EvalResHoldPlotButton.Layout.Row = 4; app.EvalResHoldPlotButton.Layout.Column = 1; app.EvalResHoldPlotButton.Text = 'Hold Data'; app.EvalResHoldPlotButton.Text = 'Hold Data'; app.EvalResHoldPlotButton.FontSize = app.fontSize14; %% Create EvalResSmpDataPanel app.EvalResMpDataPanel = uipanel(app.EvalResSmpDataPanel); app.EvalResSmpDataGridLayout.cowHeight = {30, 'lx',60}; app.EvalResSmpDataGridLayout.ColumnWidth = {'lx', 'lx'}; *% Create EvalResSmpDataDropDown_Label app.EvalResSmpDataGridLayout.ColumNidth = {'lx', 'lx'}; *% Create EvalResSmpDataDropDown_Label = uilabel(app.EvalReSSmpDataGridLayout); app.EvalResSmpDataDropDown_Label.Layout.Column = 1; app.EvalResSmpDataDropDown_Label.Layout.Column = 1; app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel; app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel; app.EvalResSmpDataDropDown_Label.AgoutAcolums = 'center'; app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; *% Create EvalResSmpDataDropDown_Label.RackgroundColor = backGroundColorLabel; app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; *% Create EvalResSmpDataDropDown_Label.RackgroundColor = backGroundColorLabel; app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; *% Create EvalResSmpDataDropDown_Label.RackgroundColor = backGroundColorLabel; app.EvalResSmpDataDropDown_Label.RackgroundColor = backGroundColorLabel; app.EvalResSmpDataDropDown_Label.RackgroundColor = backGroundColorLabel; app.EvalResSmpDataDropDown_Label.RackgroundColor = backGroundColorLabel; app.EvalResSmpDataDropDown_Label.RackgroundColor = backGroundColorLabel; app.EvalResSmpDataDropDown_Label.GaugedFcn = (center'; app.EvalResSmpDataDropDown_Label.GaugedFcn = 'latex'; ************</pre>	1761	<pre>app.EvalResExportMeasButton.Layout.Row = 6;</pre>
1763app.EvalResExportMeasButton.Fext = 'Export Report';1764app.EvalResExportMeasButton.FontSize = app.fontsize14;1765app.EvalResExportMeasButton.Visible = 'off';1766%% Create EvalResHoldPlotButton1767app.EvalResHoldPlotButton.Layout.Row = 4;1769app.EvalResHoldPlotButton.Layout.Column = 1;1770app.EvalResHoldPlotButton.Text = 'Hold Data';1771app.EvalResHoldPlotButton.FontSize = app.fontsize14;1772%% Create EvalResSmpDataPanel1773app.EvalResMpDataPanel1774app.EvalResSmpDataGridLayout, columnWidth = {'lx', 'lx'};1775app.EvalResSmpDataPanel1776app.EvalResSmpDataPoneDayon.Label1777%% Create EvalResSmpDataDropDown_Label.Layout.Column = 1;1778app.EvalResSmpDataDropDown_Label.ayout.Column = 1;1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1778app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel;1783app.EvalResSmpDataDropDown_Label.BackgroundColor = lackGroundColorLabel;1784app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1785app.EvalResSmpDataDropDown_Label.MorizontalAlignment = 'center';1786%% Create EvalResSmpDataDropDown_Label.SeckgroundColor = lackGroundColorLabel;1785app.EvalResSmpDataDropDown_Label.MorizontalAlignment = 'center';1786	1762	<pre>app.EvalResExportMeasButton.Layout.Column = 2;</pre>
1764app.EvalResExportMeasButton.FontSize = app.fontSize14;1765app.EvalResExportMeasButton.Visible = 'off';1766%% Create EvalResHoldPlotButton1767app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' state');1768app.EvalResHoldPlotButton.Layout.Column = 1;1770app.EvalResHoldPlotButton.Text = 'Hold Data';1771app.EvalResHoldPlotButton.FontSize = app.fontsize14;1772%% Create EvalResSmpDataPanel1773app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel);1774app.EvalResSmpDataGridLayout.ColumnWidth = {'1x', '1x'};1775app.EvalResSmpDataGridLayout.ColumnWidth = {'1x', '1x'};1776app.EvalResSmpDataDropDown_Label1777%% Create EvalResSmpDataDropDown_Label1778app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.HorizontalLignment = 'center';1781app.EvalResSmpDataDropDown_Label.HorizontalLignment = 'center';1782app.EvalResSmpDataDropDown_Label.HorizontalLignment = 'center';1784app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786% Create EvalResSmpDataDropDown_Label.Interpreter = 'latex';1787app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1788app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1789app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1789app.EvalResSmpDa	1763	<pre>app.EvalResExportMeasButton.Text = 'Export Report';</pre>
1765app.EvalResExportMeasButton.Visible = 'off';1766%% Create EvalResHoldPlotButton1767app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' state');1768app.EvalResHoldPlotButton.Layout.Row = 4;1769app.EvalResHoldPlotButton.Text = 'Hold Data'; app.EvalResHoldPlotButton.FontSize = app.fontsizel4;1771app.EvalResSmpDataPanel1773app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel); app.EvalResSmpDataGridLayout.RowHeight = {30, 'lx',60}; app.EvalResSmpDataGridLayout.ColumNWidth = {'lx','sc};1776app.EvalResSmpDataGridLayout.columNWidth = {'lx','sc}; i;1777% Create EvalResSmpDataBoropDown_Label1778app.EvalResSmpDataDropDown_Label.Layout.Column = 1; app.EvalResSmpDataDropDown_Label.Layout.Column = 1; app.EvalResSmpDataDropDown_Label.FontSize = app.fontsizel4; app.EvalResSmpDataDropDown_Label.FontSize = app.fontsizel4; app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1784 app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1785 app.EvalResSmpDataDropDown_Label.Interpreter = (latex';1786 app.EvalResSmpDataDropDown_Layout.Column = 2; app.EvalResSmpDataDropDown_ValueChangedFcn = @(src, event) EvalResSmpDataDrop	1764	app.EvalResExportMeasButton.FontSize = app.fontsize14;
1766% Create EvalResHoldPlotButton1767app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' state');1768app.EvalResHoldPlotButton.Layout.Row = 4;1769app.EvalResHoldPlotButton.Layout.Column = 1;1770app.EvalResHoldPlotButton.Text = 'Hold Data';1771app.EvalResHoldPlotButton.FontSize = app.fontsizel4;1772% Create EvalResSmpDataPanel1773app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel);1774app.EvalResSmpDataGridLayout.columnWidth = {'1x','1x'};1775app.EvalResSmpDataGridLayout.columNWidth = {'1x','1x'};1776% Create EvalResSmpDataDropDown_Label1777% Create EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout);1778app.EvalResSmpDataDropDown_Label.Layout.Row = 1;1779app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';app.EvalResSmpDataDropDown_Label.FontSize = app.fontsizel4;1781app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1782app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1784app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786% Create EvalResSmpDataDropDown_Label.Interpreter = 'latex';1787app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1788app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1789app.EvalResSmpDataDropDown.Label.Interpreter = 'latex';1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790 <th>1765</th> <th><pre>app.EvalResExportMeasButton.Visible = 'off';</pre></th>	1765	<pre>app.EvalResExportMeasButton.Visible = 'off';</pre>
<pre>1767 app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,' state'); 1768 app.EvalResHoldPlotButton.Layout.Row = 4; app.EvalResHoldPlotButton.Layout.Column = 1; 1770 app.EvalResHoldPlotButton.Text = 'Hold Data'; 1771 app.EvalResSmpDataPanel 1773 app.EvalResSmpDataPanel 1773 app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel); 1775 app.EvalResSmpDataGridLayout.RowHeight = {30, 'lx', 60}; 1776 app.EvalResSmpDataDropDown_Label 1777 % Create EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout) ; 1777 % Create EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout) ; 1779 app.EvalResSmpDataDropDown_Label.Layout.Row = 1; 1780 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1780 app.EvalResSmpDataDropDown_Label.ScatGroundColor = backGroundColorLabel; 1783 app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center'; app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center'; app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1784 app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1785 app.EvalResSmpDataDropDown_Label.GroundColor = backGroundColorLabel; 1788 app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1786 % Create EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1788 app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1789 app.EvalResSmpDataDropDown_Label.GroundColor = backGroundColorLabel; 1790 app.EvalResSmpDataDropDown_Label.Ground = 2; 1790 app.EvalResSmpDataDropDown.Layout.Row = 1; app.EvalResSmpDataDropDown.Layout.Row = 1; 1790 app.EvalResSmpDataDropDown.ValueChangedFcn(app, src, event); EvalResSmpDataDropDown_ValueChangedFcn(app, src, event); 1792 % Create EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout); 1794 app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2]; 1795 app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];</pre>	1766	<pre>%% Create EvalResHoldPlotButton</pre>
<pre>state'); app.EvalResHoldPlotButton.Layout.Row = 4; app.EvalResHoldPlotButton.Layout.Column = 1; app.EvalResHoldPlotButton.Text = 'Hold Data'; app.EvalResHoldPlotButton.FontSize = app.fontsize14; app.EvalResSmpDataPanel = uipanel(app.EvalResTabGridLayout); app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel); app.EvalResSmpDataGridLayout.columnWidth = {'1x', '00}; app.EvalResSmpDataGridLayout.ColumnWidth = {'1x', '1x'}; rmf app.EvalResSmpDataDropDownLabel = uilabel(app.EvalResSmpDataGridLayout) ; rmf app.EvalResSmpDataDropDownLabel = uilabel(app.EvalResSmpDataGridLayout) ; rmf app.EvalResSmpDataDropDownLabel Layout.Row = 1; app.EvalResSmpDataDropDownLabel.Layout.Row = 1; app.EvalResSmpDataDropDownLabel.Layout.Column = 1; rmf app.EvalResSmpDataDropDownLabel.Layout.Column = 1; rmf app.EvalResSmpDataDropDownLabel.FortSize = app.fontsize14; app.EvalResSmpDataDropDownLabel.FortSize = app.fontsize14; app.EvalResSmpDataDropDownLabel.Interr = 'latex'; app.EvalResSmpDataDropDownLabel.Interrer = 'latex'; app.EvalResSmpDataDropDownLabel.Interrer = 'latex'; app.EvalResSmpDataDropDownLabel.Interrer = 'latex'; app.EvalResSmpDataDropDownLabel.Interrer = 'latex'; app.EvalResSmpDataDropDownLabel.Interrer = 'latex'; app.EvalResSmpDataDropDownLabel.Solumn = 2; app.EvalResSmpDataDropDownLayout.Column = 2; app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataParamsGridLayout.Layout.Row = 2; app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];<!--</th--><th>1767</th><th>app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,'</th></pre>	1767	app.EvalResHoldPlotButton = uibutton(app.EvalResMeasDataGridLayout,'
1768app.EvalResHoldPlotButton.Layout.Row = 4;1769app.EvalResHoldPlotButton.Layout.Column = 1;1770app.EvalResHoldPlotButton.Layout.Column = 1;1771app.EvalResMoldPlotButton.Text = 'Hold Data';1772%% Create EvalResSmpDataPanel1773app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel);1774app.EvalResSmpDataGridLayout.RowHeight = {30, 'Lx',60};1776app.EvalResSmpDataGridLayout.columnWidth = {'Lx','Lx'};1777%% Create EvalResSmpDataDropDownLabel1778app.EvalResSmpDataDropDownLabel = uilabel(app.EvalResSmpDataGridLayout);1779app.EvalResSmpDataDropDownLabel.Layout.Column = 1;1780app.EvalResSmpDataDropDownLabel.BackgroundColor = backGroundColorLabel;1781app.EvalResSmpDataDropDownLabel.News(color = backGroundColorLabel;1782app.EvalResSmpDataDropDownLabel.HorizontalAlignment = 'center';1784app.EvalResSmpDataDropDownLabel.Interpreter = 'latex';1785app.EvalResSmpDataDropDownLabel.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDownLabel.Interpreter = 'latex';1787app.EvalResSmpDataDropDownLabel.Interpreter = 'latex';1788app.EvalResSmpDataDropDown.Layout.Row = 1;1789app.EvalResSmpDataDropDown.Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown1787app.EvalResSmpDataDropDown.Layout.Row = 1;1788app.EvalResSmpDataDropDown.Layout.Row = 2;1790app.EvalResSmpDataDropDown.Layout.Column = 2;1791app.EvalResSmpDataPropDown.ValueChangedFcn (app. src		<pre>state');</pre>
app.EvalResHoldPlotButton.Layout.Column = 1;1770app.EvalResHoldPlotButton.Text = 'Hold Data';1771app.EvalResHoldPlotButton.Text = 'Hold Data';1772app.EvalResHoldPlotButton.FontSize = app.fontsizel4;1773app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel);1774app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel);1775app.EvalResSmpDataGridLayout.RowHeight = {30,'lx',60};1776app.EvalResSmpDataGridLayout.ColummWidth = {'lx','lx'};1777% Create EvalResSmpDataDropDown_Label1778app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout)1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1781app.EvalResSmpDataDropDown_Label.FortSize = app.fontNo.';1782app.EvalResSmpDataDropDown_Label.FortSize = app.fontSizel4;1783app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786% Create EvalResSmpDataDropDown_Label.Interpreter = 'latex';1787app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1788app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Layout.column = 2;1790app.EvalResSmpDataDropDown.Layout.column = 2;1790app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Layout.Column = 2;1791app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataParamsGridLayout = uigridlayout(ap	1768	<pre>app.EvalResHoldPlotButton.Layout.Row = 4;</pre>
1770app.EvalResHoldPlotButton.Text = 'Hold Data';1771app.EvalResHoldPlotButton.FontSize = app.fontsize14;1772%% Create EvalResSmpDataPanel = uipanel(app.EvalResTabGridLayout);1773app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel);1774app.EvalResSmpDataGridLayout.RowHeight = {30, 'lx',60};1775app.EvalResSmpDataGridLayout.ColumnWidth = {'lx', 'lx'};1776app.EvalResSmpDataBopDown_Label1777%% Create EvalResSmpDataDropDown_Label1778app.EvalResSmpDataDropDown_Label.Layout.Row = 1;1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1781app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1782app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1783app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1784app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown_Label.Interpreter = 'latex';1787app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1788app.EvalResSmpDataDropDown_Layout.Column = 2;1790app.EvalResSmpDataDropDown.Layout.Colums = 1;1772app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)1788app.EvalResSmpDataPropDown_Layout1791app.EvalResSmpDataPropDown_ValueChangedFcn = @(src, event);1792%% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayo	1769	<pre>app.EvalResHoldPlotButton.Layout.Column = 1;</pre>
1771app.EvalResHoldPlotButton.FontSize = app.fontsize14;1772%% Create EvalResSmpDataPane11773app.EvalResSmpDataPane1 = uipanel(app.EvalResTabGridLayout);1774app.EvalResSmpDataGridLayout = uigridLayout(app.EvalResSmpDataPanel);1775app.EvalResSmpDataGridLayout.RowHeight = {30,'1x',60};1776app.EvalResSmpDataGridLayout.ColumnWidth = {'1x','1x'};1777%% Create EvalResSmpDataDropDown_Label1778app.EvalResSmpDataDropDown_Label1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1782app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1783app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1784app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown_Label.Interpreter = 'latex';1788app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1789app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown1787app.EvalResSmpDataDropDown_Layout.Row = 1;1790app.EvalResSmpDataDropDown_Layout.Row = 2;1790app.EvalResSmpDataDropDown_Layout.Column = 2;1790app.EvalResSmpDataParamsGridLayout1791app.EvalResSmpDataParamsGridLayout1792%% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayout1794app.EvalResSmpDataParamsGridLa	1770	<pre>app.EvalResHoldPlotButton.Text = 'Hold Data';</pre>
<pre>1772 %* Create EvalResSmpDataPanel 1773 app.EvalResSmpDataPanel = uipanel(app.EvalResTabGridLayout); 1774 app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel); 1775 app.EvalResSmpDataGridLayout.RowHeight = {30, 'lx',60}; 1776 app.EvalResSmpDataGridLayout.ColumnWidth = {'lx','lx'}; 1777 %* Create EvalResSmpDataDropDown_Label 1778 app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout); 1779 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1780 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1781 app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.'; 1782 app.EvalResSmpDataDropDown_Label.Rext = 'Measuring Point No.'; 1783 app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1784 app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1785 app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1789 app.EvalResSmpDataDropDown_Layout.Column = 2; 1790 app.EvalResSmpDataDropDown_ValueChangedFcnt = @(src, event) EvalResSmpDataDropDown_ValueChangedFcnt = @(src, event); 1792 %* Create EvalResSmpDataParamsGridLayout 1793 app.EvalResSmpDataParamsGridLayout 1794 app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2]; </pre>	1771	<pre>app.EvalResHoldPlotButton.FontSize = app.fontsize14;</pre>
1773app.EvalResSmpDataPanel = uipanel(app.EvalResTabGridLayout);1774app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel);1775app.EvalResSmpDataGridLayout.RowHeight = {30, '1x',60};1776app.EvalResSmpDataGridLayout.ColumnWidth = {'1x', '1x'};1777%% Create EvalResSmpDataDropDown_Label1778app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout);1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1781app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1782app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel;1783app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1784app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown_Layout.Column = 2;1790app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event);1791app.EvalResSmpDataDropDown_ValueChangedFcn = @(src, event);1792%% Create EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout);1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout);1794app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	2 1772	<pre>%% Create EvalResSmpDataPanel</pre>
1774app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel);1775app.EvalResSmpDataGridLayout.RowHeight = {30, '1x', 60};1776app.EvalResSmpDataGridLayout.ColumNWidth = {'1x', '1x'};1777%* Create EvalResSmpDataDropDown_Label1778app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout)1779app.EvalResSmpDataDropDown_Label.Layout.Row = 1;1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1782app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1783app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1784app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%* Create EvalResSmpDataDropDown_Label.Interpreter = 'latex';1787app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1788app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Layout.Colum = 2;1791app.EvalResSmpDataDropDown_ValueChangedFcn(app, src, event)1792%* Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayout1794app.EvalResSmpDataParamsGridLayout.Layout.Column = 2;1795app.EvalResSmpDataParamsGridLayout.Layout.Column = 2;1794app.EvalResSmpDataParamsGridLayout1794app.EvalResSmpDataParamsGridLayout.Layout.Column = 12;1795app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	¥ 1773	<pre>app.EvalResSmpDataPanel = uipanel(app.EvalResTabGridLayout);</pre>
1775app.EvalResSmpDataGridLayout.RowHeight = {30, 'lx',60};1776app.EvalResSmpDataGridLayout.ColumnWidth = {'lx','lx'};1777%% Create EvalResSmpDataDropDown_Label1778app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout);1779app.EvalResSmpDataDropDown_Label.Layout.Row = 1;1770app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1781app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1782app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1784app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown1787app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1788app.EvalResSmpDataDropDown1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.ValueChangedFcn(app, src, event)1792%% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayout1794app.EvalResSmpDataParamsGridLayout.Layout.Row = 2;1795app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1774	<pre>app.EvalResSmpDataGridLayout = uigridlayout(app.EvalResSmpDataPanel);</pre>
1776app.EvalResSmpDataGridLayout.ColumnWidth = {'1x','1x'};1777% Create EvalResSmpDataDropDown_Label1778app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout)1779app.EvalResSmpDataDropDown_Label.Layout.Row = 1;1779app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1781app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel;1783app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1784app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786% Create EvalResSmpDataDropDown_Layout.Row = 1;1787app.EvalResSmpDataDropDown.Layout.Column = 2;1788app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)1791app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event);1792% Create EvalResSmpDataPropDown.ValueChangedFcn = @(src, event);1793app.EvalResSmpDataPropDown_ValueChangedFcn = @(src, event);1794app.EvalResSmpDataPramsGridLayout = uigridlayout(app. EvalResSmpDataPramsGridLayout);1794app.EvalResSmpDataPramsGridLayout.Layout.Column = [1 2];	1775	<pre>app.EvalResSmpDataGridLayout.RowHeight = {30, '1x',60};</pre>
<pre>1777 % Create EvalResSmpDataDropDown_Label 1778 app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout) ; 1779 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1780 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1781 app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.'; 1782 app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel; 1783 app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14; 1784 app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center'; 1785 app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1786 % Create EvalResSmpDataDropDown 1787 app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout); 1788 app.EvalResSmpDataDropDown_Layout.Row = 1; 1790 app.EvalResSmpDataDropDown.Layout.Column = 2; 1790 app.EvalResSmpDataDropDown.Items = {}; 1791 app.EvalResSmpDataDropDown_ValueChangedFcn(app, src, event) EvalResSmpDataDropDown_ValueChangedFcn(app, src, event); 1792 % Create EvalResSmpDataParamsGridLayout 1793 app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataParamsGridLayout.Row = 2; 1794 app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];</pre>	1776	<pre>app.EvalResSmpDataGridLayout.ColumnWidth = {'1x','1x'};</pre>
1778app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout) ;1779app.EvalResSmpDataDropDown_Label.Layout.Row = 1;1770app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1780app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1781app.EvalResSmpDataDropDown_Label.BeckgroundColor = backGroundColorLabel;1783app.EvalResSmpDataDropDown_Label.BeckgroundColor = backGroundColorLabel;1784app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1785app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1786%% Create EvalResSmpDataDropDown_Label.Interpreter = 'latex';1787app.EvalResSmpDataDropDown1788app.EvalResSmpDataDropDown1789app.EvalResSmpDataDropDown.Layout.Row = 1;1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)1791app.EvalResSmpDataDropDown_ValueChangedFcn(app, src, event);1792%% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataParamsGridLayout);1794app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1777	<pre>%% Create EvalResSmpDataDropDown_Label</pre>
<pre>; 1779 app.EvalResSmpDataDropDown_Label.Layout.Row = 1; 1780 app.EvalResSmpDataDropDown_Label.Layout.Column = 1; 1781 app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.'; 1782 app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel; 1783 app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14; 1784 app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center'; 1785 app.EvalResSmpDataDropDown_Label.Interpreter = 'latex'; 1786 % Create EvalResSmpDataDropDown 1787 app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout); 1788 app.EvalResSmpDataDropDown.Layout.Row = 1; 1790 app.EvalResSmpDataDropDown.Layout.Column = 2; 1790 app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)</pre>	1778	<pre>app.EvalResSmpDataDropDown_Label = uilabel(app.EvalResSmpDataGridLayout)</pre>
1779app.EvalResSmpDataDropDown_Label.Layout.Row = 1;1780app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1781app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1782app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel;1783app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1784app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown1787app.EvalResSmpDataDropDown1788app.EvalResSmpDataDropDown.Layout.Row = 1;1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Layout.Column = 2;1791app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)1792%% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataParamsGridLayout);1794app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	2	;
1780app.EvalResSmpDataDropDown_Label.Layout.Column = 1;1781app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1782app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel;1783app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1784app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown1787app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout);1788app.EvalResSmpDataDropDown1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)1791evalResSmpDataDropDown_ValueChangedFcn (app, src, event);1792%% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataGridLayout);1794app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	3 1779	<pre>app.EvalResSmpDataDropDown_Label.Layout.Row = 1;</pre>
1781app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';1782app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel;1783app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1784app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown1787app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout);1788app.EvalResSmpDataDropDown.Layout.Row = 1;1790app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Items = {};1791app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)1792%% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout);1794app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1780	<pre>app.EvalResSmpDataDropDown_Label.Layout.Column = 1;</pre>
1782app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel;1783app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1784app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786 %% Create EvalResSmpDataDropDown 1787app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout);1788app.EvalResSmpDataDropDown.Layout.Row = 1;1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Items = {};1791app.EvalResSmpDataDropDown_ValueChangedFcn = @(src, event)1792 %% Create EvalResSmpDataParamsGridLayout 1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app.1794app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1781	<pre>app.EvalResSmpDataDropDown_Label.Text = 'Measuring Point No.';</pre>
1783app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;1784app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%* Create EvalResSmpDataDropDown1787app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout);1788app.EvalResSmpDataDropDown.Layout.Row = 1;1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Items = {};1791app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)1792%* Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app.1794app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1782	app.EvalResSmpDataDropDown_Label.BackgroundColor = backGroundColorLabel;
1784app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786% Create EvalResSmpDataDropDown1787app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout);1788app.EvalResSmpDataDropDown.Layout.Row = 1;1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Layout.Column = 2;1791app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)EvalResSmpDataDropDown_ValueChangedFcn (app, src, event);1792% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataGridLayout = uigridlayout(app. EvalResSmpDataGridLayout);1794app.EvalResSmpDataParamsGridLayout.Layout.Row = 2;1795app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1783	<pre>app.EvalResSmpDataDropDown_Label.FontSize = app.fontsize14;</pre>
1785app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';1786%% Create EvalResSmpDataDropDown1787app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout);1788app.EvalResSmpDataDropDown.Layout.Row = 1;1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Items = {};1791app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)1792%% Create EvalResSmpDataDropDown_ValueChangedFcn(app, src, event);1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout);1794app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	3 1784	<pre>app.EvalResSmpDataDropDown_Label.HorizontalAlignment = 'center';</pre>
<pre>%% Create EvalResSmpDataDropDown app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout); app.EvalResSmpDataDropDown.Layout.Row = 1; app.EvalResSmpDataDropDown.Layout.Column = 2; app.EvalResSmpDataDropDown.Items = {}; app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event) EvalResSmpDataDropDown_ValueChangedFcn(app, src, event); %% Create EvalResSmpDataParamsGridLayout app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout); app.EvalResSmpDataParamsGridLayout.Layout.Row = 2; app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];</pre>	1785	<pre>app.EvalResSmpDataDropDown_Label.Interpreter = 'latex';</pre>
1787app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout);1788app.EvalResSmpDataDropDown.Layout.Row = 1;1789app.EvalResSmpDataDropDown.Layout.Column = 2;1790app.EvalResSmpDataDropDown.Items = {};1791app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event)EvalResSmpDataDropDown_ValueChangedFcn(app, src, event);1792%% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app.EvalResSmpDataGridLayout);app.EvalResSmpDataParamsGridLayout.Layout.Row = 2;1794app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	° 1786	%% Create EvalResSmpDataDropDown
<pre>1788 app.EvalResSmpDataDropDown.Layout.Row = 1; app.EvalResSmpDataDropDown.Layout.Column = 2; app.EvalResSmpDataDropDown.Items = {}; app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event) EvalResSmpDataDropDown_ValueChangedFcn(app, src, event); 1792 %% Create EvalResSmpDataParamsGridLayout 1793 app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout); 1794 app.EvalResSmpDataParamsGridLayout.Layout.Row = 2; app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];</pre>	1787	<pre>app.EvalResSmpDataDropDown = uidropdown(app.EvalResSmpDataGridLayout);</pre>
<pre>app.EvalResSmpDataDropDown.Layout.Column = 2; app.EvalResSmpDataDropDown.Items = {}; app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event) EvalResSmpDataDropDown_ValueChangedFcn(app, src, event); %% Create EvalResSmpDataParamsGridLayout 1793 app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout); app.EvalResSmpDataParamsGridLayout.Layout.Row = 2; app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];</pre>	3 1788	<pre>app.EvalResSmpDataDropDown.Layout.Row = 1;</pre>
<pre>1790 app.EvalResSmpDataDropDown.Items = {}; app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event) EvalResSmpDataDropDown_ValueChangedFcn(app, src, event); 1792 % Create EvalResSmpDataParamsGridLayout 1793 app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout); 1794 app.EvalResSmpDataParamsGridLayout.Layout.Row = 2; app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];</pre>	2 1789	app.EvalResSmpDataDropDown.Lavout.Column = 2:
<pre>1791 app.EvalResSmpDataDropDown.ValueChangedFcn = @(src, event) EvalResSmpDataDropDown_ValueChangedFcn(app, src, event); 1792 %% Create EvalResSmpDataParamsGridLayout 1793 app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout); 1794 app.EvalResSmpDataParamsGridLayout.Layout.Row = 2; 1795 app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];</pre>	x 1790	<pre>app.EvalResSmpDataDropDown.Items = {}:</pre>
EvalResSmpDataDropDown_ValueChangedFcn(app, src, event);1792%% Create EvalResSmpDataParamsGridLayout1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout);1794app.EvalResSmpDataParamsGridLayout.Layout.Row = 2; app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1791	app.EvalResSmpDataDropDown.ValueChangedFcn = $@(src. event)$
<pre>1792 %% Create EvalResSmpDataParamsGridLayout 1793 app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout); 1794 app.EvalResSmpDataParamsGridLayout.Layout.Row = 2; 1795 app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];</pre>		EvalResSmpDataDropDown ValueChangedFcn(app, src. event):
1793app.EvalResSmpDataParamsGridLayout = uigridlayout(app. EvalResSmpDataGridLayout);1794app.EvalResSmpDataParamsGridLayout.Layout.Row = 2; app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1792	% Create EvalResSmpDataParamsGridLavout
EvalResSmpDataGridLayout); app.EvalResSmpDataParamsGridLayout.Layout.Row = 2; app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1793	app.EvalResSmpDataParamsGridLavout = uigridlavout(app.
1794app.EvalResSmpDataParamsGridLayout.Layout.Row = 2;1795app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	EvalResSmpDataGridLavout):
1795app.EvalResSmpDataParamsGridLayout.Layout.Column = [1 2];	1794	app. Eval ResSmpDataParamsGridLavout Lavout Row = 2°
	1795	app.EvalResSmpDataParamsGridLavout.Lavout.Column = [1 2]:

1796	<pre>app.EvalResSmpDataParamsGridLayout.RowHeight =</pre>
1 202	{20,20,20,20,20,20,20,20,20,20,20,20,20,2
1797	<pre>app.EvalResSmpDataParamsGridLayout.ColumnWidth = { '1x', '1x' };</pre>
1798	<pre>app.EvalResSmpDataParamsGridLayout.Scrollable = 'on';</pre>
1799	%% Create EvalResSmpData_W_X_Label
1800	app.EvalResSmpData_W_X_Label = uilabel(app.
1001	EvalResSmpDataParamsGridLayout);
1801	<pre>app.EvalResSmpData_W_X_Label.Text = '\$\langle x \rangle_{\textrm{mean}}\</pre>
1000	<pre>textrm{ in } \mu m\$';</pre>
1802	app.EvalResSmpData_w_X_Label.Interpreter = 'latex';
1803	app.EvalResSmpData_W_X_Label.HorizontalAlignment = 'center';
1804	app.EvalResSmpData_w_X_Label.BackgroundColor = backGroundColorLabel;
1800	%% Create EvalResSmpData_w_X
1800	<pre>'numeric');</pre>
1807	<pre>app.EvalResSmpData_W_X.HorizontalAlignment = 'center';</pre>
1808	<pre>app.EvalResSmpData_W_X.Editable = 'off';</pre>
<u>;</u> 1809	<pre>%% Create EvalResSmpData_W_X_std_Label</pre>
1810	app.EvalResSmpData_W_X_std_Label = uilabel(app.
	<pre>EvalResSmpDataParamsGridLayout);</pre>
1811	<pre>app.EvalResSmpData_W_X_std_Label.Text = '\$\langle x \rangle_{std</pre>
	<pre>}\textrm{ in } \mu m\$';</pre>
2 1812	<pre>app.EvalResSmpData_W_X_std_Label.Interpreter = 'latex';</pre>
f 1813	<pre>app.EvalResSmpData_W_X_std_Label.HorizontalAlignment = 'center';</pre>
1814	<pre>app.EvalResSmpData_W_X_std_Label.BackgroundColor = backGroundColorLabel;</pre>
1815	<pre>%% Create EvalResSmpData_W_X_std</pre>
1816	<pre>app.EvalResSmpData_W_X_std = uieditfield(app.</pre>
alla	<pre>EvalResSmpDataParamsGridLayout, 'numeric');</pre>
g 1817	<pre>app.EvalResSmpData_W_X_std.HorizontalAlignment = 'center';</pre>
<u>n</u> 1818	<pre>app.EvalResSmpData_W_X_std.Editable = 'off';</pre>
<u>c</u> 1819	<pre>%% Create EvalResSmpData_W_Y_Label</pre>
1820	<pre>app.EvalResSmpData_W_Y_Label = uilabel(app.</pre>
	EvalResSmpDataParamsGridLayout);
1821	<pre>app.EvalResSmpData_W_Y_Label.Text = '\$\langle y \rangle_{\textrm{mean}}\</pre>
	textrm{ in } \mu m\$';
5 1822 7 1000	<pre>app.EvalResSmpData_W_Y_Label.Interpreter = 'latex';</pre>
1823	<pre>app.EvalResSmpData_W_Y_Label.HorizontalAlignment = 'center';</pre>
5 1824	app.EvalResSmpData_W_Y_Label.BackgroundColor = backGroundColorLabel;
1000	%% Create EvalResSmpData_W_Y
5 1820	app.EvalResSmpData_w_Y = uleditfield(app.EvalResSmpDataParamsGridLayout,
	'numeric');
1000	app.EvalResSmpData_w_Y.HorizontalAlignment = "center";
1828	app.EvalResSmpData_w_Y.Editable = "off";
1029	33 Create EvalkesSmpData_w_r_std_tabet
1090	app.EvalResSmpDataDaramsGrid avout).
<mark>ธ</mark> 1 & ว 1	ann EvalResSmnData W V std Label Text - (\$)langlo v \rangle ()textrefetd
VICOT 0	app.Evaluesompole.instructions
1820	son EvalResSmoData W Y std Label Interpreter - 'latev'.
TOOT	app. Evacicos importa_w_r_sta_cabet. Interpreter - tatex ,

1833	app EvalPocSmpData W V std Labol HorizontalAlignmont - 'contor':
1000	app.EvalResSmpData_w_1_Std_Label.NoTizontatAtignment = center ,
1834	app.EvalResSmpData_w_Y_std_Label.BackgroundColor = backGroundColorLabel;
1835	%% Create EvalResSmpData_W_Y_std
1836	<pre>app.EvalResSmpData_W_Y_std = uieditfield(app.</pre>
	<pre>EvalResSmpDataParamsGridLayout, 'numeric');</pre>
1837	<pre>app.EvalResSmpData_W_Y_std.HorizontalAlignment = 'center';</pre>
1838	<pre>app.EvalResSmpData_W_Y_std.Editable = 'off';</pre>
1839	%% Create EvalResSmpData_W_X_squared_Label
1840	app.EvalResSmpData W X squared Label = uilabel(app.
	EvalBesSmpDataParamsGridLavout):
18/1	ann EvalRecSmpData W X squared Label Text - $\langle \xi \rangle$ and $\langle \chi \rangle$ (rangle $\langle \chi \rangle$
1041	app: Evalues Simplify and the second secon
1049	cextimitimediff(cextimit in f (mu m (2)),
1842	app.EvalResSmpData_w_X_squared_Label.interpreter = latex ;
1843	app.EvalResSmpData_w_X_squared_Label.HorizontalAlignment = 'center';
1844	app.EvalResSmpData_W_X_squared_Label.BackgroundColor =
	backGroundColorLabel;
1845	<pre>%% Create EvalResSmpData_W_X_squared</pre>
<u>z</u> 1846	<pre>app.EvalResSmpData_W_X_squared = uieditfield(app.</pre>
	<pre>EvalResSmpDataParamsGridLayout, 'numeric');</pre>
1847	<pre>app.EvalResSmpData_W_X_squared.HorizontalAlignment = 'center';</pre>
1848	<pre>app.EvalResSmpData_W_X_squared.Editable = 'off';</pre>
1849	<pre>%% Create EvalResSmpData_W_X_squared_std_Label</pre>
2 1850	<pre>app.EvalResSmpData_W_X_squared_std_Label = uilabel(app.</pre>
al	<pre>EvalResSmpDataParamsGridLayout);</pre>
1851	<pre>app.EvalResSmpData_W_X_squared_std_Label.Text = '\$\langle x^{2} \rangle_</pre>
-	<pre>{\textrm{std}}\textrm{ in } \mu m^{2}\$';</pre>
1852	<pre>app.EvalResSmpData_W_X_squared_std_Label.Interpreter = 'latex';</pre>
1853	<pre>app.EvalResSmpData_W_X_squared_std_Label.HorizontalAlignment = 'center';</pre>
a 1854	<pre>app.EvalResSmpData_W_X_squared_std_Label.BackgroundColor =</pre>
<u>0</u>	backGroundColorLabel;
1855	<pre>%% Create EvalResSmpData_W_X_squared_std</pre>
1856	app.EvalResSmpData_W_X_squared_std = uieditfield(app.
5	EvalResSmpDataParamsGridLavout.'numeric'):
2 1857	app.EvalResSmpData_W_X_squared_std.HorizontalAlignment = 'center':
1858	app.EvalResSmpData W X squared std.Editable = off' :
1859	% Create EvalResSmpData_W_Y_squared_Label
1860	app EvalResSmpData W Y squared Label = uilabel(app
2000	EvalResSmpDataParamsGridLavout):
5	ann EvalResSmpData W Y squared Label Text = $\frac{1}{2}$ and $\frac{1}{2}$ and $\frac{1}{2}$
	$tevtrm{mean} \left(tevtrm{ in } \right) mum^{2} tevtrm{ mean} \right)$
1862	ann EvalRecSmpData W V squared Label Interpreter - 'latex':
5 1863	app.EvalNesSmpData_w_1_squared_Label_HorizontalAlignment = 'contor';
1964	app.EvalResSmpData_W_1_Squared_Label_ReskgroundColor_
1004	app.EvatResshipbata_w_1_squareu_Labet.Backgroundcotor =
106E	
1000 1066	% Create EvalkesSmpData_w_T_Squared
5 T900	app.EvalkesSmpData_w_i_squared = uledittleid(app.
	EvalkesSmpuataParamsGridLayout, 'numeric');
1000	<pre>app.EvalkesSmpData_w_Y_squared.HorizontalAlignment = 'center';</pre>
1808	app.EvalResSmpData_W_Y_squared.Editable = 'off';
2	

1869	<pre>%% Create EvalResSmpData_W_Y_squared_std_Label</pre>
1870	$app.EvalResSmpData_W_Y_squared_std_Label = uilabel(app.$
	<pre>EvalResSmpDataParamsGridLayout);</pre>
1871	<pre>app.EvalResSmpData_W_Y_squared_std_Label.Text = '\$\langle y^{2} \rangle_</pre>
	<pre>{\textrm{std}}\textrm{ in } \mu m^{2}\$';</pre>
1872	<pre>app.EvalResSmpData_W_Y_squared_std_Label.Interpreter = 'latex';</pre>
1873	<pre>app.EvalResSmpData_W_Y_squared_std_Label.HorizontalAlignment = 'center';</pre>
1874	<pre>app.EvalResSmpData_W_Y_squared_std_Label.BackgroundColor =</pre>
	backGroundColorLabel;
1875	<pre>%% Create EvalResSmpData_W_Y_squared_std</pre>
1876	$app.EvalResSmpData_W_Y_squared_std = uieditfield(app.$
	<pre>EvalResSmpDataParamsGridLayout, 'numeric');</pre>
1877	<pre>app.EvalResSmpData_W_Y_squared_std.HorizontalAlignment = 'center';</pre>
1878	<pre>app.EvalResSmpData_W_Y_squared_std.Editable = 'off';</pre>
1879	%% Create EvalResSmpData_W_XY_Label
1880	app.EvalResSmpData_W_XY_Label = uilabel(app.
	EvalResSmpDataParamsGridLavout):
1881	app.EvalResSmpData_W_XY_Label.Text = '\$\langle_xy_\rangle_{mean
	}}\textrm{ in } \mu m^{2}\$':
1882	app.EvalResSmpData W XY Label.Interpreter = 'latex':
1883	app.EvalResSmpData W XY Label.HorizontalAlignment = 'center':
1884	app.EvalResSmpData W XY Label.BackgroundColor = backGroundColorLabel:
885	% Create EvalResSmpData W XY
886	app. EvalResSmpData W XY = uieditfield(app. EvalResSmpDataParamsGridLavout
	'numeric'):
887	app.EvalResSmpData W XY.HorizontalAlignment = 'center':
888	app.EvalResSmpData W XY.Editable = 'off':
889	% Create EvalResSmpData_W_XY_std_Label
890	app.EvalResSmpData W XY std Label = uilabel(app.
	EvalResSmpDataParamsGridLavout):
891	app.EvalResSmpData W XY std Label.Text = '\$\langle xy \rangle {
	<pre>std}}\textrm{ in } \mu m^{2}\$':</pre>
.892	<pre>app.EvalResSmpData_W_XY_std_Label.Interpreter = 'latex';</pre>
1893	<pre>app.EvalResSmpData_W_XY_std_Label.HorizontalAlignment = 'center':</pre>
1894	app.EvalResSmpData_W_XY_std_Label.BackgroundColor = backGroundColorLabel
1895	%% Create EvalResSmpData_W_XY_std
1896	app.EvalResSmpData_W_XY_std = uieditfield(app.
	<pre>EvalResSmpDataParamsGridLayout, 'numeric');</pre>
1897	<pre>app.EvalResSmpData_W_XY_std.HorizontalAlignment = 'center';</pre>
1898	app.EvalResSmpData_W_XY_std.Editable = 'off';
1899	%% Create EvalResSmpData_dWx_Label
1900	app.EvalResSmpData_dWx_Label = uilabel(app.
	EvalResSmpDataParamsGridLavout):
1901	app.EvalResSmpData_dWx_Label.Text = '\$\textrm{Beamdiameter } d_{x\textrm}
	<pre>{ mean}} \textrm{ in } \mu m\$':</pre>
1902	app.EvalResSmpData_dWx_Label.Interpreter = 'latex':
1903	app.EvalResSmpData_dWx_Label.HorizontalAlignment = 'center':
	ann EvalPacSmnData dWy Labal PackgroundCalar - backGroundCalarlabal,
1904	app, Evalnespillpbala_uwx_Label, backurbulluculur = backurbulluculur Label,

1905	%% Create EvalResSmpData_dWx
1906	app.EvalResSmpData_dWx = uieditfield(app.EvalResSmpDataParamsGridLayout,
	'numeric');
1907	<pre>app.EvalResSmpData_dWx.HorizontalAlignment = 'center';</pre>
1908	<pre>app.EvalResSmpData_dWx.Editable = 'off';</pre>
1909	<pre>%% Create EvalResSmpData_dWx_std_Label</pre>
1910	app.EvalResSmpData_dWx_std_Label = uilabel(app.
	<pre>EvalResSmpDataParamsGridLayout);</pre>
911	<pre>app.EvalResSmpData_dWx_std_Label.Text = '\$\textrm{Beamdiameter } d_{x\</pre>
	<pre>textrm{ std}} \textrm{ in } \mu m\$';</pre>
912	<pre>app.EvalResSmpData_dWx_std_Label.Interpreter = 'latex';</pre>
.913	app.EvalResSmpData_dWx_std_Label.HorizontalAlignment = 'center';
914	<pre>app.EvalResSmpData_dWx_std_Label.BackgroundColor = backGroundColorLabel;</pre>
915	<pre>%% Create EvalResSmpData_dWx_std</pre>
916	<pre>app.EvalResSmpData_dWx_std = uieditfield(app.</pre>
	<pre>EvalResSmpDataParamsGridLayout, 'numeric');</pre>
917	app.EvalResSmpData_dWx_std.HorizontalAlignment = 'center';
918	<pre>app.EvalResSmpData_dWx_std.Editable = 'off';</pre>
919	<pre>%% Create EvalResSmpData_dWy_Label</pre>
920	app.EvalResSmpData_dWy_Label = uilabel(app.
	<pre>EvalResSmpDataParamsGridLayout);</pre>
921	<pre>app.EvalResSmpData_dWy_Label.Text = '\$\textrm{Beamdiameter } d_{y\textrm</pre>
	<pre>{ mean}} \textrm{ in } \mu m\$';</pre>
22	<pre>app.EvalResSmpData_dWy_Label.Interpreter = 'latex';</pre>
923	<pre>app.EvalResSmpData_dWy_Label.HorizontalAlignment = 'center';</pre>
924	app.EvalResSmpData_dWy_Label.BackgroundColor = backGroundColorLabel;
925	%% Create EvalResSmpData_dWy
)26	app.EvalResSmpData_dWy = uieditfield(app.EvalResSmpDataParamsGridLayout,
	'numeric');
927	app.EvalResSmpData_dWy.HorizontalAlignment = 'center';
928	<pre>app.EvalResSmpData_dWy.Editable = 'off';</pre>
.929	<pre>%% Create EvalResSmpData_dWy_std_Label</pre>
)30	app.EvalResSmpData_dWy_std_Label = uilabel(app.
	<pre>EvalResSmpDataParamsGridLayout);</pre>
931	<pre>app.EvalResSmpData_dWy_std_Label.Text = '\$\textrm{Beamdiameter } d_{y\</pre>
	<pre>textrm{ std}} \textrm{ in } \mu m\$';</pre>
932	<pre>app.EvalResSmpData_dWy_std_Label.Interpreter = 'latex';</pre>
933	<pre>app.EvalResSmpData_dWy_std_Label.HorizontalAlignment = 'center';</pre>
934	<pre>app.EvalResSmpData_dWy_std_Label.BackgroundColor = backGroundColorLabel;</pre>
.935	%% Create EvalResSmpData_dWy_std
936	app.EvalResSmpData_dWy_std = uieditfield(app.
0.07	<pre>EvalResSmpDataParamsGridLayout, 'numeric');</pre>
1937	<pre>app.EvalResSmpData_dWy_std.HorizontalAlignment = 'center';</pre>
1938	<pre>app.EvalResSmpData_dWy_std.Editable = 'off';</pre>
1939	%% Create EvalResSmpData_phi_Label
940	app.EvalResSmpData_phi_Label = uilabel(app.
0.41	<pre>EvalResSmpDataParamsGridLayout);</pre>
.941	<pre>app.EvalResSmpData_phi_Label.Text = '\$\textrm{Azimutangle } \varphi_{\</pre>
1	<pre>textrm{mean}} \textrm{ in } A°\$';</pre>

1942	<pre>app.EvalResSmpData_phi_Label.Interpreter = 'latex';</pre>
1943	<pre>app.EvalResSmpData_phi_Label.HorizontalAlignment = 'center';</pre>
1944	<pre>app.EvalResSmpData_phi_Label.BackgroundColor = backGroundColorLabel;</pre>
1945	%% Create EvalResSmpData_phi
1946	<pre>app.EvalResSmpData_phi = uieditfield(app.EvalResSmpDataParamsGridLayout,</pre>
	'numeric');
1947	<pre>app.EvalResSmpData_phi.HorizontalAlignment = 'center';</pre>
1948	<pre>app.EvalResSmpData_phi.Editable = 'off';</pre>
1949	<pre>%% Create EvalResSmpData_phi_std_Label</pre>
1950	<pre>app.EvalResSmpData_phi_std_Label = uilabel(app.</pre>
	<pre>EvalResSmpDataParamsGridLayout);</pre>
1951	<pre>app.EvalResSmpData_phi_std_Label.Text = '\$\textrm{Azimutangle } \varphi_</pre>
	<pre>{\textrm{std}} \textrm{ in } °\$';</pre>
1952	<pre>app.EvalResSmpData_phi_std_Label.Interpreter = 'latex';</pre>
1953	<pre>app.EvalResSmpData_phi_std_Label.HorizontalAlignment = 'center';</pre>
1954	<pre>app.EvalResSmpData_phi_std_Label.BackgroundColor = backGroundColorLabel;</pre>
1955	<pre>%% Create EvalResSmpData_phi_std</pre>
¥ 1956	<pre>app.EvalResSmpData_phi_std = uieditfield(app.</pre>
	<pre>EvalResSmpDataParamsGridLayout, 'numeric');</pre>
1957	<pre>app.EvalResSmpData_phi_std.HorizontalAlignment = 'center';</pre>
1958	<pre>app.EvalResSmpData_phi_std.Editable = 'off';</pre>
1959	<pre>%% Create EvalResSmpData_phi_std</pre>
1 960	<pre>app.EvalResSmpInfoTextArea = uitextarea(app.EvalResSmpDataGridLayout);</pre>
g 1961	<pre>app.EvalResSmpInfoTextArea.HorizontalAlignment = 'center';</pre>
1962	<pre>app.EvalResSmpInfoTextArea.Editable = 'off';</pre>
<u>1963</u>	<pre>app.EvalResSmpInfoTextArea.Layout.Row = 3;</pre>
1 964	<pre>app.EvalResSmpInfoTextArea.Layout.Column = [1 2];</pre>
1965	<pre>app.EvalResSmpInfoTextArea.Value ='Two or more measuring points have an</pre>
ave	azimuth angle difference of $10 \hat{A}^\circ$ or more. Could be badly conditioned
<u>n</u>	evaluation or a astigmatic laserbeam. Either way the results cannot
22	be trusted.';
7 1966	<pre>app.EvalResSmpInfoTextArea.Visible = 'off';</pre>
1967	<pre>app.EvalResSmpInfoTextArea.FontSize = 14;</pre>
2 1968	<pre>app.EvalResSmpInfoTextArea.FontColor = 'red';</pre>
2 1969	<pre>app.EvalResSmpInfoTextArea.FontWeight = 'bold';</pre>
\$ 1970	%% Create EvalResImDataPanel
1 971	<pre>app.EvalResImDataPanel = uipanel(app.EvalResTabGridLayout);</pre>
≝ 1972	app.EvalResImDataGridLayout = uigridlayout(app.EvalResImDataPanel);
p 1973	app.EvalResImDataGridLayout.RowHeight = {30,240,'1x',50};
5 1974	<pre>app.EvalResImDataGridLayout.ColumnWidth = {'1x', '1x'};</pre>
1975 I 1975	%% Create EvalResImDataDropDown_Label
1976	<pre>app.EvalResImDataDropDown_Label = uilabel(app.EvalResImDataGridLayout);</pre>
1977	app.EvalResImDataDropDown_Label.Layout.Row = 1;
1978	app.EvalResImDataDropDown_Label.Layout.Column = 1;
1979	<pre>app.EvalResImDataDropDown_Label.Text = 'Image No.';</pre>
2 1980	<pre>app.EvalResImDataDropDown_Label.BackgroundColor = backGroundColorLabel;</pre>
Mag 1981	<pre>app.EvalResImDataDropDown_Label.FontSize = app.fontsize14;</pre>
6 1982 1000	<pre>app.EvalResImDataDropDown_Label.HorizontalAlignment = 'center';</pre>
n 1983	<pre>app.EvalResImDataDropDown_Label.Interpreter = 'latex';</pre>
_	

1984	<pre>%% Create EvalResImDataDropDown</pre>
1985	<pre>app.EvalResImDataDropDown = uidropdown(app.EvalResImDataGridLayout);</pre>
1986	<pre>app.EvalResImDataDropDown.Layout.Row = 1;</pre>
1987	app.EvalResImDataDropDown.Layout.Column = 2;
1988	<pre>app.EvalResImDataDropDown.Items = {};</pre>
1989	app.EvalResImDataDropDown.ValueChangedFcn = @(src, event)
	<pre>EvalResImDataDropDown_ValueChangedFcn(app, src, event);</pre>
1990	
1991	<pre>%% Create EvalResImDataParamsGridLayout</pre>
1992	<pre>app.EvalResImDataParamsGridLayout = uigridlayout(app.</pre>
	<pre>EvalResImDataGridLayout);</pre>
1993	<pre>app.EvalResImDataParamsGridLayout.Layout.Row = 2;</pre>
1994	<pre>app.EvalResImDataParamsGridLayout.Layout.Column = [1 2];</pre>
1995	app.EvalResImDataParamsGridLayout.RowHeight = {20,20,20,20,20,20,20,20};
1996	<pre>app.EvalResImDataParamsGridLayout.ColumnWidth = {'1x','1x'};</pre>
1997	<pre>app.EvalResImDataParamsGridLayout.Scrollable = 'on';</pre>
1998	%% Create EvalResImData_W_X_Label
999	<pre>app.EvalResImData_W_X_Label = uilabel(app.EvalResImDataParamsGridLavout)</pre>
	;
2000	<pre>app.EvalResImData_W_X_Label.Text = '\$\langle x \rangle\textrm{ in } \mu</pre>
	m\$';
2001	<pre>app.EvalResImData_W_X_Label.Interpreter = 'latex';</pre>
2002	<pre>app.EvalResImData_W_X_Label.HorizontalAlignment = 'center';</pre>
003	app.EvalResImData W X Label.BackgroundColor = backGroundColorLabel;
004	%% Create EvalResImData_W_X
005	app.EvalResImData W X = uieditfield(app.EvalResImDataParamsGridLavout.
	numeric'):
006	<pre>app.EvalResImData_W_X.HorizontalAlignment = 'center';</pre>
007	app.EvalResImData_W_X.Editable = 'off';
008	%% Create EvalResImData_W_Y_Label
009	<pre>app.EvalResImData_W_Y_Label = uilabel(app.EvalResImDataParamsGridLayout)</pre>
010	<pre>app.EvalResImData_W_Y_Label.Text = '\$\langle y \rangle\textrm{ in } \mu</pre>
	m\$';
2011	<pre>app.EvalResImData_W_Y_Label.Interpreter = 'latex';</pre>
012	<pre>app.EvalResImData_W_Y_Label.HorizontalAlignment = 'center';</pre>
013	<pre>app.EvalResImData_W_Y_Label.BackgroundColor = backGroundColorLabel;</pre>
014	%% Create EvalResImData_W_Y
2015	<pre>app.EvalResImData_W_Y = uieditfield(app.EvalResImDataParamsGridLayout,'</pre>
	numeric');
2016	<pre>app.EvalResImData_W_Y.HorizontalAlignment = 'center';</pre>
2017	<pre>app.EvalResImData_W_Y.Editable = 'off';</pre>
2018	%% Create EvalResImData_W_X_squared_Label
2019	<pre>app.EvalResImData_W_X_squared_Label = uilabel(app.</pre>
	<pre>EvalResImDataParamsGridLayout);</pre>
020	app.EvalResImData_W_X_squared_Label.Text = $\frac{1}{\sqrt{2} \sqrt{2}}$
	<pre>textrm{ in } \mu m^{2}\$';</pre>
2021	<pre>app.EvalResImData_W_X_squared_Label.Interpreter = 'latex';</pre>
0000	ann EvalResImData W X squared Label HorizontalAlignment = 'center'

2023	<pre>app.EvalResImData_W_X_squared_Label.BackgroundColor = backGroundColorLabel;</pre>
2024	%% Create EvalResImData_W_X_squared
2025	app.EvalResImData_W_X_squared = uieditfield(app.
	EvalResImDataParamsGridLayout, 'numeric');
2026	app.EvalResImData_W_X_squared.HorizontalAlignment = 'center':
2027	app.EvalResImData W_X_squared.Editable = 'off':
2028	% Create EvalResImData_W_Y_squared_Label
2029	app.EvalResImData W Y squared Label = uilabel(app.
	EvalResImDataParamsGridLavout):
2030	app.EvalResImData W Y squared Label.Text = $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$
2000	textrm{ in } \mu m^{2}s':
≥ 2031	app.EvalResImData W Y squared Label.Interpreter = 'latex':
eq 2032	app.EvalResImData W Y squared Label.HorizontalAlignment = 'center':
njua 2033	app.EvalResImData W Y squared Label.BackgroundColor =
× <	backGroundColorLabel:
et 2034	%% Create EvalResImData_W_Y_squared
	app.EvalResImData W Y squared = uieditfield(app.
the the	EvalResImDataParamsGridLavout. 'numeric'):
plio 2036	app.EvalResImData W Y squared.HorizontalAlignment = 'center':
	app.EvalResImData_W_Y_squared.Editable = 'off':
L 19 2038	%% Create EvalResImData_W_XY_Label
⁵ > 2039	<pre>app.EvalResImData_W_XY_Label = uilabel(app.EvalResImDataParamsGridLavout</pre>
at T);
	<pre>app.EvalResImData_W_XY_Label.Text = '\$\langle xy \rangle\textrm{ in } \</pre>
arbe	mu m^{2}\$':
ی سے 2041	app.EvalResImData_W_XY_Label.Interpreter = 'latex':
della 2042	app.EvalResImData_W_XY_Label.HorizontalAlignment = 'center':
2043	<pre>app.EvalResImData_W_XY_Label.BackgroundColor = backGroundColorLabel;</pre>
<u></u>	%% Create EvalResImData_W_XY
	app.EvalResImData_W_XY = uieditfield(app.EvalResImDataParamsGridLayout,'
ersi s the	<pre>numeric');</pre>
2046 th al	<pre>app.EvalResImData_W_XY.HorizontalAlignment = 'center';</pre>
jo 2047	<pre>app.EvalResImData_W_XY.Editable = 'off';</pre>
0.00 2048	<pre>%% Create EvalResImData_dWx_Label</pre>
ty a 2049	<pre>app.EvalResImData_dWx_Label = uilabel(app.EvalResImDataParamsGridLayout)</pre>
nal	;
bu 2050	<pre>app.EvalResImData_dWx_Label.Text = '\$\textrm{Beamdiameter } d_{x} \</pre>
erte ed c	<pre>textrm{ in } \mu m\$';</pre>
g 2051	<pre>app.EvalResImData_dWx_Label.Interpreter = 'latex';</pre>
ad 2052	<pre>app.EvalResImData_dWx_Label.HorizontalAlignment = 'center';</pre>
e e 2053	<pre>app.EvalResImData_dWx_Label.BackgroundColor = backGroundColorLabel;</pre>
∩⊨ ₂₀₅₄	<pre>%% Create EvalResImData_dWx</pre>
2 055	app.EvalResImData_dWx = uieditfield(app.EvalResImDataParamsGridLayout,'
O	numeric');
1 2056	<pre>app.EvalResImData_dWx.HorizontalAlignment = 'center';</pre>
0 # 2057	<pre>app.EvalResImData_dWx.Editable = 'off';</pre>
2058	<pre>%% Create EvalResImData_dWy_Label</pre>
2059	app.EvalResImData_dWy_Label = uilabel(app.EvalResImDataParamsGridLayout)

	;
2060	<pre>app.EvalResImData_dWy_Label.Text = '\$\textrm{Beamdiameter } d_{y} \</pre>
	<pre>textrm{ in } \mu m\$';</pre>
2061	app.EvalResImData_dWv_Label.Interpreter = 'latex':
2062	app.EvalResImData dWv Label.HorizontalAlignment = 'center':
2063	app EvalResImData dWy Label BackgroundColor = backGroundColorLabel:
2064	%% Create EvalResImData dWv
2004	app EvalResImData dWy - uieditfield(app EvalResImDataParamsGrid avout '
2000	app.Evativestimbata_dwy = diedititetd(app.Evativestimbatararamsorideayout,
2066	app EvalPecImData dWv HerizentalAlignment - 'conter's
2000	app.EvalResImbata_dwy.HolizontatAtignment = Center,
2007	app.EvalResImbala_dwy.Eultable = 011 ;
2008	%% Create EvalResImpata_pni_Label
2009	app.EvalResimbata_pni_Label = ullabel(app.EvalResimbataParamsGridLayout)
0.0 = 0	;
2070	app.EvalResImData_phi_Label.Text = '\$\textrm{Azimutangle } \varphi \
	<pre>textrm{ in } A°\$';</pre>
2071	app.EvalResImData_phi_Label.Interpreter = 'latex';
2072	app.EvalResImData_phi_Label.HorizontalAlignment = 'center';
2073	app.EvalResImData_phi_Label.BackgroundColor = backGroundColorLabel;
2074	<pre>%% Create EvalResImData_phi</pre>
2075	app.EvalResImData_phi = uieditfield(app.EvalResImDataParamsGridLayout,'
/	numeric');
2076	app.EvalResImData_phi.HorizontalAlignment = 'center';
2077	app.EvalResImData_phi.Editable = 'off';
2078	<pre>%% Create EvalResImDataUIAxes</pre>
2079	<pre>app.EvalResImDataUIAxes = uiaxes(app.EvalResImDataGridLayout);</pre>
2080	<pre>app.EvalResImDataUIAxes.Layout.Row = 3;</pre>
2081	<pre>app.EvalResImDataUIAxes.Layout.Column = [1 2];</pre>
2082	<pre>%% Create EvalResImDataColorGridLayout</pre>
2083	<pre>app.EvalResImDataColorGridLayout = uigridlayout(app.</pre>
)	<pre>EvalResImDataGridLayout);</pre>
2084	<pre>app.EvalResImDataColorGridLayout.Layout.Row = 4;</pre>
2085	<pre>app.EvalResImDataColorGridLayout.Layout.Column = [1 2];</pre>
2086	<pre>app.EvalResImDataColorGridLayout.RowHeight = {'1x'};</pre>
2087	<pre>app.EvalResImDataColorGridLayout.ColumnWidth = {120, '1x', '1x', '1x', '1x'</pre>
	};
2088	<pre>app.EvalResImDataColorGridLayout.Scrollable = 'on';</pre>
2089	%% Create EvalResImDataChangeColorMapButton
2090	app.EvalResImDataChangeColorMapButton = uibutton(app.
)	EvalResImDataColorGridLayout, 'state');
2091	<pre>app.EvalResImDataChangeColorMapButton.Text = 'Switch Color Map';</pre>
2092	app.EvalResImDataChangeColorMapButton.Lavout.Row = 1:
2093	app.EvalResImDataChangeColorMapButton.Lavout.Column = 1:
2094	app.EvalResImDataChangeColorMapButton.ValueChangedFcn = @(src.event)
	EvalResImDataChangeColorManButton ValueChangedEcn(ann src event)
2095	app.EvalResImDataChangeColorMapButton_FontSize = app_fontsize14.
2000	% Create EvalResColorMinLabel
2000	ann EvalResColorMinLabel = uilabel(ann EvalResImDataColorGridLavout);
2001	apprevalues colorMinlabel Text = $\frac{\min Value}{1}$
2030	apprevatioscotornineabet.rext - nin vatue ,
2099	app. EvalResColorMinlabel. Layout. Row = 1:
---------------	--
2100	app.EvalResColorMinLabel.Layout.Column = 2:
2100	app EvalResColorMinLabel FontSize = app fontsize14:
2101	apprevaluescolorMinLabel Interpreter = 'latex':
2102	app.EvallesColorMinLabel.Enterpreter = tattex,
2100 2104	app.EvalResColorMinLabel HorizontalAlignment - 'center':
2104	% Create EvalBesColorMinEditEield
2100 2106	ann EvalResColorMinEditField - wieditfield(ann
2100	EvalPosImDataColorGridLayout /pumoric/):
2107	and EvalPosColorMinEditEiold Layout Pow = 1;
2107	app.EvalResColorMinEditField Layout Column = 2;
2100	app. EvalPacColorMinEditField Limits $- [0, 255]$
2109	app.EvalResColorMinEditField HarizentalAlignment - /conter/
2110 9111	app.EvalRescolorMinEditField.HorizontalAtignment = center;
2111	app.EvalRescolorMinEditField.FontSize = app.TontSize14;
2112 9119	app.EvalRescolorMinEditEiold ValueChangedErn = @(arg. svent)
2113	<pre>app.EvalRescolormineditfield.valuechangedFCn = @(SrC, event) EvalPacImChangeCalarMan(app</pre>
0114	EvalkesimenangecolorMap(app, src, event);
2114	10. Consta LineMassCale Marticle 1
2115	%% Create Live Live <thlive< th=""> Live Live <thl< th=""></thl<></thlive<>
2110	app.EvalResColorMaxLabel = ullabel(app.EvalResImDataColorGridLayout);
2117	app.EvalResColorMaxLabel.Text = 'Max Value';
2118	app.EvalResColorMaxLabel.Layout.Row = 1;
2119	app.EvalResColorMaxLabel.Layout.Column = 4;
3 2120	<pre>app.EvalResColorMaxLabel.FontSize = app.fontsize14;</pre>
2121	<pre>app.EvalResColorMaxLabel.Interpreter = 'latex';</pre>
2122	<pre>app.EvalResColorMaxLabel.BackgroundColor = backGroundColorLabel;</pre>
2123	<pre>app.EvalResColorMaxLabel.HorizontalAlignment = 'center';</pre>
2124	<pre>%% Create LiveMeasColorMinEditField</pre>
3 2125	<pre>app.EvalResColorMaxEditField = uieditfield(app.</pre>
2	<pre>EvalResImDataColorGridLayout, 'numeric');</pre>
<u>3</u> 2126	<pre>app.EvalResColorMaxEditField.Layout.Row = 1;</pre>
2127	app.EvalResColorMaxEditField.Layout.Column = 5;
2128	<pre>app.EvalResColorMaxEditField.Limits = [0 255];</pre>
2129	<pre>app.EvalResColorMaxEditField.HorizontalAlignment = 'center';</pre>
2130	app.EvalResColorMaxEditField.FontSize = app.fontsize14;
2131	app.EvalResColorMaxEditField.Value = app.EvalResColorMap(2);
2132	app.EvalResColorMaxEditField.ValueChangedFcn = @(src, event)
5	<pre>EvalResImChangeColorMap(app, src, event);</pre>
2133	Show the figure after all components are created
2 134	app.UIFigure.Visible = 'on';
2135	end
2136	end
2137	
2138	% App creation and deletion
2139	<pre>methods (Access = public)</pre>
2140	
2141	% Construct app
2142	<pre>function app = lbcUI</pre>
2143	
- '	

```
2153
                           2154
                           2156
TU Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
VIEN vur knowledge hub
The approved original version of this thesis is available in print at TU Wien Bibliothek.
                           2157
                           2158
                           2159
                           2160
                                         14
                                         18
                                         19
```

214621472148

2150

end

```
% Create UIFigure and components
createComponents(app)
% Register the app with App Designer
registerApp(app, app.UIFigure);
if nargout == 0
    clear app
end
end
% Code that executes before app deletion
function delete(app)
    % Delete UIFigure when app is deleted
imaqreset()
delete(app.UIFigure)
end
end
```

 $7.31: Function AutoMeasISO111146 Part1MethodRBGroup_SelectionChangedFcn$

```
function AutoMeasIS0111146Part1MethodRBGroup_SelectionChangedFcn(app, src, event)
1
2
   %AutoMeasIS0111146Part1MethodRBGroup_SelectionChangedFcn ValueChangedFcn listening
      to changes of the selected Automated
   %Measurement Configuration RadioButtonGroup.
4
       Enables and Disables UI Elements depending on the selected
   %
5
       Configuration Method and triggers an update of the Measuring Point
   %
6
   %
       preview axes.
   newSelection = event.NewValue;
7
8
9
   if strcmp('Configure around centerpoint', newSelection.Text)
       app.AutoMeasIS0111146Part1MeasAreaEditField.Enable = 'off';
       app.AutoMeasIS0111146Part10ffseEditField.Enable = 'on';
       app.AutoMeasIS0111146Part1RayleighEditField.Enable = 'on';
       app.AutoMeasIS0111146Part1TimesRayleighEditField.Enable = 'on';
   else
       app.AutoMeasIS0111146Part1MeasAreaEditField.Enable = 'on';
       app.AutoMeasIS0111146Part10ffseEditField.Enable = 'off';
       app.AutoMeasIS0111146Part1RayleighEditField.Enable = 'off';
       app.AutoMeasIS0111146Part1TimesRayleighEditField.Enable = 'off';
   end
  UpdatePreviewAxes(app, src, event);
   end
```

7.32: Function AutoMeasISO111146Part1StartMeasButton_ButtonPushedFcn

function AutoMeasIS0111146Part1StartMeasButton_ButtonPushedFcn(app,src,event)

2 %AutoMeasIS0111146Part1StartMeasButton_ButtonPushedFcn ButtonPushedFcn listening to the Start Button of the Automated

```
3
   % Measurement.
 4
   %
        Starts the automated measuerement using the parameters given in the
   %
        EditFields of the UI.
 6
   try
 7
        app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'off';
 8
        if app.AutoMeasCancelButton.Value
            app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on';
            app.AutoMeasCancelButton.Value = 0;
            app.AutoMeasIS0111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2];
12
            app.AutoMeasIS0111146Part1StatusLabel.Text = 'Inactive';
            uialert(app.UIFigure,{'Measurement canceled!'},'Measurement Stopped
               Information','Icon','success','CloseFcn','uiresume(gcbf)');
14
            uiwait(gcbf)
            return;
        end
17
        measurement = Measurement(app.AutoMeasIS0111146Part1RayleighEditField.Value,...
18
            app.AutoMeasIS0111146Part1TimesRayleighEditField.Value,...
            app.AutoMeasIS0111146Part1NoMeasPointsEditField.Value,...
            app.AutoMeasIS0111146Part1MeasPerPointEditField.Value);
21
22
        % Reset old Measurementdata
23
        measurement.reset();
24
25
        app.AutoMeasIS0111146Part1StatusLabel.Text = 'Active';
        app.AutoMeasIS0111146Part1StatusLabel.BackgroundColor = [0.2 1 0.3];
27
28
        app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Go to zero';
29
        drawnow;
        moveToZpos(app,0);
        pause(1);
        uialert(app.UIFigure,{'Cover laser!'},'Next Step Information','Icon','info','
           CloseFcn', 'uiresume(gcbf)');
        uiwait(gcbf)
        app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Background
           Measurement';
        for i = 1 : size(app.measurementPoints,2)
            app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Value = '-';
            app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField.Value = '--';
38
            drawnow;
            moveToZpos(app,app.measurementPoints(i));
            pause(1);
41
            app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Value = int2str(i);
42
43
            MeasurementPoint = SingleMeasurementPoint(app.currPos(1));
44
            for k = 1 : app.AutoMeasIS0111146Part1MeasPerPointEditField.Value
                if app.AutoMeasCancelButton.Value
                    app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on';
47
                    app.AutoMeasCancelButton.Value = 0;
                    app.AutoMeasIS0111146Part1StatusLabel.BackgroundColor = [1 0.07
48
```

<pre>49 49 50 50 50 51 52 53 6 54 55 55 55 50 51 51 52 55 50 51 51 52 55 50 51 51 51 52 55 50 51 51 51 52 55 50 51 51 51 52 55 50 51 51 52 55 50 51 51 52 55 50 52 50 52 53 55 55 55 50 50 51 52 55 55 50 50 50 50 50 50 50 50 50 50 50</pre>	<pre>Stopped '); ue = mage(app tPoint); o';</pre>
<pre>50 uialert(app.UIFigure,{'Measurement canceled!'},'Measurement Information','Icon','success','CloseFcn','uiresume(gcbf) 51 uiwait(gcbf) 52 return; 53 end 54 app.AutoMeasISO111146Part1ProgressCurrMeasPointMeasEditField.Val int2str(k); 55 MeasurementPoint.addMeasurement(SingleMeasurement(TriggerCameraI</pre>	<pre>Stopped '); ue = mage(app tPoint); o';</pre>
Information','Icon','success','CloseFcn','uiresume(gcbf)51uiwait(gcbf)52return;53end54app.AutoMeasISO111146Part1ProgressCurrMeasPointMeasEditField.Val int2str(k);55MeasurementPoint.addMeasurement(SingleMeasurement(TriggerCameraI	<pre>'); ue = mage(app tPoint); o';</pre>
<pre>51 uiwait(gcbf) 52 return; 53 end 54 app.AutoMeasISO111146Part1ProgressCurrMeasPointMeasEditField.Val</pre>	ue = mage(app tPoint); o';
<pre>52 return; 53 end 54 app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField.Val</pre>	ue = mage(app tPoint); o';
<pre>53 end 54 app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField.Val</pre>	ue = mage(app tPoint); o';
<pre>54 app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField.Val int2str(k); 55 MeasurementPoint.addMeasurement(SingleMeasurement(TriggerCameraI</pre>	ue = mage(app tPoint); o';
<pre>int2str(k); 55 MeasurementPoint.addMeasurement(SingleMeasurement(TriggerCameraI</pre>	<pre>mage(app tPoint); o';</pre>
55 MeasurementPoint.addMeasurement(SingleMeasurement(TriggerCameraI	<pre>mage(app tPoint); o';</pre>
	tPoint); o';
)));	tPoint); o';
56 pause(1);	tPoint); o';
57 end	<pre>tPoint); o';</pre>
b8 measurement.getBackgroundData().addSingLeMeasurementPoint(Measuremen	o';
59 end 60 end AutoMagetCO111114CDast1DasancesCumActionEditEigld Value - Contension	0';
app.AutoMeasISOIIII46Part1ProgressCurrActionEditField.Value = 'Go to Zer	
app.AutoMeasISOIIII46Part1ProgressCurrMeasPointEditField Value = -;	
app.Automeasisoiiiii4opartiprogresscurrmeaspointmeascuitrietu.vatue = -	;
64 move To Zpos (app. 0):	
65 nause(1):	
66 uialert(app UTEigure {'Remove laser cover!'} 'Next Step Information' 'Ic	on''
info'. 'CloseEcn'. 'uiresume(acbf)'):	o n ,
67 uiwait(gcbf)	
68 app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Laser Mea	surement
69 for i = 1 : size(app.measurementPoints,2)	
70 app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Value = '-'	;
71 app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField.Value =	'-';
72 drawnow;	
<pre>73 moveToZpos(app,app.measurementPoints(i));</pre>	
74 pause(1);	
75 app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Value = int	2str(i);
<pre>76 MeasurementPoint = SingleMeasurementPoint(app.currPos(1));</pre>	
for k = 1 : app.AutoMeasIS0111146Part1MeasPerPointEditField.Value	
78 if app.AutoMeasCancelButton.Value	
app.AutoMeasIS0111146PartIStartMeasButton.Enable = 'on';	
80 app.AutoMeasLancelButton.Value = 0;	07
app.AutoMeasISUIIII46PartIStatusLabel.BackgroundColor = [1 0	.07
U.Z];	
app.Automeds150111140rd(l15tdtusLdbet.Text = 11dctive;	Stoppod
The survey of th	·) .
84 uiwait(achf)	/ ,
85 return:	
86 end	
87 app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField Val	ue =
int2str(k):	
88 MeasurementPoint.addMeasurement(SingleMeasurement(TriggerCameraI	mage(app
	5

<pre>pause(1); end measurement.getMeasurementData().addSingleMeasurementPoint(MeasurementPoint); end app.AutoMeasIS0111146PartIProgressCurrActionEditField.Value = 'Go to zero'; app.AutoMeasIS0111146PartIProgressCurrMeasPointEditField.Value = '-'; drawnow; moveToZpos(app,0); pause(1) app.AutoMeasIS0111146PartIProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasIS0111146PartIProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasIS0111146PartIProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasIS0111146PartIProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasIS0111146PartIStatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasIS0111146PartIStatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIEvalResTab(app); if not(isempty(app.loadedMeas.getResults())) updateGUIEvalResTab(app); end catch ME app.AutoMeasIS0111146PartIStartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146PartIStartMeasButton.Enable = 'on'; end</pre>)));
<pre>end measurement.getMeasurementData().addSingleMeasurementPoint(MeasurementPoint) ; end app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Go to zero'; app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField.Value = '-'; drawnow; moveToZpos(app.0); pause(1) app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasIS0111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasIS0111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successful!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConFEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end</pre>	39	<pre>pause(1);</pre>
<pre>measurement.getMeasurementData().addSingleMeasurementPoint(MeasurementPoint) ; end app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Go to zero'; app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Value = '-'; drawnow; moveToZpos(app,0); pause(1) app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; uisave({'measurement'}); app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; uisave({'measurement'}); app.AutoMeasIS0111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasIS0111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIEvalResTab(app); end catch ME app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end</pre>	0	end
<pre>; end app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Go to zero'; app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Value = ''; drawnow; moveToZpos(app.0); pause(1) app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasIS0111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasIS0111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIEvalTestatusLabel.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end</pre>)1	<pre>measurement.getMeasurementData().addSingleMeasurementPoint(MeasurementPoint)</pre>
<pre>end app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Go to zero'; app.AutoMeasISO111146Part1ProgressCurrMeasPointEditField.Value = ''; app.AutoMeasISO111146Part1ProgressCurrMeasPointMeasEditField.Value = ''; drawnow; moveToZpos(app,0); pause(1) app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave{{'measurement'}}; app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>		;
<pre>app.AutoMeasISO111146PartIProgressCurrActionEditField.Value = 'Go to zero'; app.AutoMeasISO111146PartIProgressCurrMeasPointEditField.Value = '-'; drawnow; moveToZpos(app,0); pause(1) app.AutoMeasISO111146PartIProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave{{'measurement'}}; app.AutoMeasISO111146PartIProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146PartIProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146PartIStatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146PartIStatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146PartIStartMeasButton.Enable = 'on'; rethrow(ME); end</pre>)2	end
<pre>app.AutoMeasISO111146Part1ProgressCurrMeasPointEditField.Value = '-'; app.AutoMeasISO11146Part1ProgressCurrMeasPointMeasEditField.Value = '-'; drawnow; moveToZpos(app,0); pause(1) app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>)3	<pre>app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Go to zero';</pre>
<pre>app.AutoMeasISOIII146PartIProgressCurrMeasPointMeasEditField.Value = '-'; drawnow; moveToZpos(app,0); pause(1) app.AutoMeasISOIII146PartIProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasISOIII146PartIProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISOIII146PartIStatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISOIII146PartIStatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISOIII146PartIStartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISOIII146PartIStartMeasButton.Enable = 'on'; end</pre>)4	<pre>app.AutoMeasIS0111146Part1ProgressCurrMeasPointEditField.Value = '-';</pre>
<pre>drawnow; moveToZpos(app,0); pause(1) app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>)5	app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField.Value = '';
<pre>moveToZpos(app,0); pause(1) app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>	96	drawnow;
<pre>pause(1) app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>)7	moveToZpos(app,0);
<pre>app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Saving Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successful!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>	98	pause(1)
<pre>Measurement'; drawnow; uisave({'measurement'}); app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>	99	app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Saving
<pre>drawnow; uisave({'measurement'}); app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>		Measurement',
<pre>uisave({'measurement'}); app.AutoMeasISO111146Part1ProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>	00	drawnow;
<pre>app.AutoMeasISOIII146PartIProgressCurrActionEditField.Value = 'Finished'; drawnow; app.AutoMeasISOIII146PartIStatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISOIII146PartIStatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISOIII146PartIStartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISOIII146PartIStartMeasButton.Enable = 'on'; end</pre>)1	<pre>uisave({'measurement'});</pre>
<pre>drawnow; app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>)2	<pre>app.AutoMeasIS0111146Part1ProgressCurrActionEditField.Value = 'Finished';</pre>
<pre>4 5 6 app.AutoMeasISO111146Part1StatusLabel.BackgroundColor = [1 0.07 0.2]; 6 app.AutoMeasISO111146Part1StatusLabel.Text = 'Inactive'; 7 8 uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement 6 Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); 9 uiwait(gcbf) 1 app.loadedMeas = measurement; 1 UpdateGUIConfEvalTab(app); 3 4 if not(isempty(app.loadedMeas.getResults())) 5 UpdateGUIEvalResTab(app); 6 end 7 8 catch ME 9 app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; 7 rethrow(ME); 1 end 2 app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; 8 end 9 app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; 9 end 9 app.Aut</pre>)3	drawnow;
<pre>app.AutoMeasISUIII146PartIStatusLabel.BackgroundColor = [1 0.07 0.2]; app.AutoMeasISOIII146PartIStatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISOIII146PartIStartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISOIII146PartIStartMeasButton.Enable = 'on'; end</pre>)4	
<pre>app.AutoMeasISOIIII46PartIStatusLabel.Text = 'Inactive'; uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISOIIII46PartIStartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISOIIII46PartIStartMeasButton.Enable = 'on'; end</pre>)5	app.AutoMeasIS0111146PartIStatusLabel.BackgroundColor = [1 0.07 0.2];
<pre>4 8 uialert(app.UIFigure,{'Measurement finished successfull!'},'Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); 9 uiwait(gcbf) 1 app.loadedMeas = measurement; 1 UpdateGUIConfEvalTab(app); 3 4 if not(isempty(app.loadedMeas.getResults())) 5 UpdateGUIEvalResTab(app); 6 end 7 8 catch ME 9 app.AutoMeasIS0111146PartIStartMeasButton.Enable = 'on'; 7 rethrow(ME); 1 end 2 app.AutoMeasIS0111146PartIStartMeasButton.Enable = 'on'; 8 end 9 app.AutoMeasIS0111146PartIStartMeasButton.Enable = 'on'; 9 end 9 app.Au</pre>	10	app.AutoMeasISUIIII46PartIStatusLabel.Text = 'Inactive';
<pre>Stater(ap).01Figure,{ Measurement finished successfult: }, Measurement Successfull Information','Icon','success','CloseFcn','uiresume(gcbf)'); uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end</pre>)(wielert(err UTFigure ('Measurement finished successfull)) Measurement
<pre>successive information, icon, success, coosered, diresdme(gcbr)), uiwait(gcbf) app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>	0	Successfull Information' 'Icon' 'success' 'ClassEcn' 'uirosumo(achf)');
<pre>app.loadedMeas = measurement; updateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) updateGUIEvalResTab(app); end catch ME app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end</pre>	0	uivoit(achf)
<pre>app.loadedMeas = measurement; UpdateGUIConfEvalTab(app); if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; end</pre>	0	uiwai(gcb)
<pre>1 app.toadedNeas = measurement; 2 UpdateGUIConfEvalTab(app); 3 4 if not(isempty(app.loadedMeas.getResults())) 5 UpdateGUIEvalResTab(app); 6 end 7 8 catch ME 9 app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; 7 rethrow(ME); 1 end 2 app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; 8 end 2 app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; 9 end 1 end 1 app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; 9 end 1 app.AutoMeasISO111146Part1StartMeasButton.Enable = 'on'; 9 end</pre>	1	app loadodMoac - moasuromont.
<pre>if not(isempty(app.loadedMeas.getResults()))</pre>	2	app. toadeuneas – measurement,
<pre>if not(isempty(app.loadedMeas.getResults())) UpdateGUIEvalResTab(app); end catch ME app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end</pre>	3	opualeoorconnevaliab(app);
<pre>UpdateGUIEvalResTab(app); end catch ME app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end</pre>	4	if not(isemnty(app loadedMeas getResults()))
<pre>end catch ME app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end</pre>	5	IndateGIIEvalResTab(ann):
<pre>catch ME app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end</pre>	6	end
<pre>catch ME app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end</pre>	7	
<pre>app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; rethrow(ME); end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end</pre>	8	catch ME
<pre>0 rethrow(ME); 1 end 2 app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; 3 end</pre>	9	<pre>app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on':</pre>
<pre>end app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; end</pre>	20	rethrow(ME);
<pre>2 app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on'; 3 end</pre>	21	end
3 end	22	<pre>app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on';</pre>
	23	end

7.33: Function CalcNumOfSteps

```
function steps = CalcNumOfSteps(app,targetPos)
%CALCNUMOFSTEPS Calculates the number of Steps the Steppermotor has to do.
% Nothing to add here.
steps=cast((targetPos_app.currPos(1))/app.StepInkrement,'int32');
end
```

7.34: Function ClearGUIEvalResTab

1

2

4

```
function ClearGUIEvalResTab(app)
 1
2
   %ClearGUIEvalResTab Clears all Evaluation Result UI elements of their
3
   %content.
   app.EvalResMeasDataResultDropDown.Items = {};
 4
   app.EvalResMeasDataResultDropDown.ItemsData = [];
6
 7
   app.EvalResMeasData_BackCorrMethod.Value = '';
8
   app.EvalResMeasData_Lambda.Value = 0;
9
   app.EvalResMeasData_ConvCrit.Value = 0;
   app.EvalResMeasData_IntegAreaFactor.Value = 0;
11
   app.EvalResMeasData_Nt.Value = 0;
   app.EvalResMeasData_EvalNorm.Value = '';
14
   app.EvalResMeasData_z_0_X.Value = 0;
   app.EvalResMeasData_z_0_Y.Value = 0;
17
    app.EvalResMeasData_z_R_X.Value = 0;
   app.EvalResMeasData_z_R_Y.Value = 0;
18
20
   app.EvalResMeasData_d_0_X.Value = 0;
   app.EvalResMeasData_d_0_Y.Value = 0;
    app.EvalResMeasData_theta_X.Value = 0;
24
    app.EvalResMeasData_theta_Y.Value = 0;
26
    app.EvalResMeasData_M_squared_X.Value = 0;
27
    app.EvalResMeasData_M_squared_Y.Value = 0;
28
    app.EvalResMeasData_M_squared_eff.Value = 0;
   app.EvalResSmpDataDropDown.Items = {};
   app.EvalResSmpDataDropDown.ItemsData = [];
   app.EvalResSmpData_W_X.Value = 0;
34
   app.EvalResSmpData_W_X_std.Value = 0;
   app.EvalResSmpData_W_Y.Value = 0;
   app.EvalResSmpData_W_Y_std.Value = 0;
   app.EvalResSmpData_W_X_squared.Value = 0;
38
   app.EvalResSmpData_W_X_squared_std.Value = 0;
39
   app.EvalResSmpData_W_Y_squared.Value = 0;
   app.EvalResSmpData_W_Y_squared_std.Value = 0;
41
   app.EvalResSmpData_W_XY.Value = 0;
42
   app.EvalResSmpData_W_XY_std.Value = 0;
43
   app.EvalResSmpData_dWx.Value = 0;
44
   app.EvalResSmpData_dWx_std.Value = 0;
   app.EvalResSmpData_dWy.Value = 0;
   app.EvalResSmpData_dWy_std.Value = 0;
47
   app.EvalResSmpData_phi.Value = 0;
48
   app.EvalResSmpData_phi_std.Value = 0;
```

```
app.EvalResImData_W_X.Value = 0;
    app.EvalResImData_W_Y.Value = 0;
52
   app.EvalResImData_W_X_squared.Value = 0;
   app.EvalResImData_W_Y_squared.Value = 0;
54
   app.EvalResImData_W_XY.Value = 0;
57
   app.EvalResImData_dWx.Value = 0;
58
   app.EvalResImData_dWy.Value = 0;
   app.EvalResImData_phi.Value = 0;
61
    cla(app.EvalResImDataUIAxes, 'reset');
62
```

end

7.35: Function ConfEvalBackCorrMethodDropDown_ValueChangedFcn

```
function ConfEvalBackCorrMethodDropDown_ValueChangedFcn(app, src, event)
1
2
   %ConfEvalBackCorrMethodDropDown_ValueChangedFcn ValueChangedFcn listening
3
   %to changes of the BackgroundCorrectionMethod Dropdown Menu
   newCorrMethod = event.Value;
4
   if strcmp('FCBAM', newCorrMethod)
6
       app.ConfEvalKernelSizeSlider.Enable = 'on';
7
       app.ConfEvalKernelSizeEditField.Enable = 'on';
8
       return;
9
   end
   app.ConfEvalKernelSizeSlider.Enable = 'off';
   app.ConfEvalKernelSizeEditField.Enable = 'off';
12
   end
```

7.36: Function ConfEvalConvergeCriterionSlider AllEvents

```
function ConfEvalConvergeCriterionSlider_AllEvents(app, src, event)
1
   %ConfEvalConvergeCriterionSlider_AllEvents listening to all events of the
3
   %ConvergeCriterion UI Elements.
   eventName = event.EventName;
4
   switch(eventName)
6
       case {'ValueChanging'}
7
           app.ConfEvalConvergeCriterionEditField.Value = event.Value;
8
       case {'ValueChanged'}
9
           app.ConfEvalConvergeCriterionSlider.Value = event.Value;
   end
12
   end
```

7.37: Function ConfEvalDrawRectButton ButtonPushedFcn

function ConfEvalDrawRectButton_ButtonPushedFcn(app, src, event) %ConfEvalDrawRectButton_ButtonPushedFcn listening to guessBeamarea button 3 Sto draw the first guess of the beam area and beamcenter.

4	<pre>if not(isa(app.ConfEvalRectRoi,'images.roi.Rectangle')) not(isvalid(app.</pre>
	ConfEvalRectRoi))
5	<pre>app.ConfEvalRectRoi = drawrectangle(app.ConfEvalPreviewImUIAxes,'LabelVisible','</pre>
	hover',
6	<pre>'Label', 'beamareaGuess', 'Position', [1 1 200 200]);</pre>
7	addlistener(app.ConfEvalRectRoi,'MovingROI',@(src, evt)
	<pre>drawConfEvalRectAllevents(app,src,evt));</pre>
8	addlistener(app.ConfEvalRectRoi,'ROIMoved',@(src, evt) drawConfEvalRectAllevents
	(app,src,evt));
9	<pre>centerRectangleX = app.ConfEvalRectRoi.Position(1)+app.ConfEvalRectRoi.Position</pre>
	(3)/2;
10	<pre>centerRectangleY = app.ConfEvalRectRoi.Position(2)+app.ConfEvalRectRoi.Position</pre>
	(4)/2;
11	<pre>app.ConfEvalPointRoi = drawpoint(app.ConfEvalPreviewImUIAxes,</pre>
12	<pre>'Position',[centerRectangleX centerRectangleY],'Color','r',</pre>
13	'InteractionsAllowed','none');
14	end
15	end

7.38: Function ConfEvalEvaluationButton_ButtonPushedFcn

```
function ConfEvalEvaluationButton_ButtonPushedFcn(app, src, event)
%ConfEvalEvaluationButton_ButtonPushedFcn listening to the Evaluate Button
%and starting a evaluation using the specified parameters in the UI.
try
    app.ConfEvalEvaluationButton.Enable = 'off';
    app.ConfEvalEvaluationStatusLabel.BackgroundColor = [0.2 1 0.3];
    app.ConfEvalEvaluationStatusLabel.Text = 'Computing';
    drawnow;
    coarseBackCorrMethod = app.ConfEvalBackCorrMethodDropDown.Value;
    fineBackCorrMethod = app.ConfEvalFineBackCorrMethodDropDown.Value;
    convCrit = app.ConfEvalConvergeCriterionEditField.Value;
    lambda = app.ConfEvalWavelengthEditField.Value;
    nT = app.ConfEvalNtEditField.Value;
    integAreaFactor = app.ConfEvalIntegAreaEditField.Value;
    evalNorm = app.ConfEvalMethodDropDown.Value;
    pixelsize = app.ConfEvalPixelsizeEditField.Value;
    kernelSize = app.ConfEvalKernelSizeEditField.Value;
    centerRectangleX = app.ConfEvalRectRoi.Position(1)+app.ConfEvalRectRoi.Position
       (3)/2;
    centerRectangleY = app.ConfEvalRectRoi.Position(2)+app.ConfEvalRectRoi.Position
       (4)/2;
    sizeRectangleX = app.ConfEvalRectRoi.Position(3);
    sizeRectangleY = app.ConfEvalRectRoi.Position(4);
    manualValue = app.ConfEvalFineCorrValueEditField.Value;
    corrData = app.loadedMeas.getMeasurementData().
       backgroundCorrectionMeasurementData(...
        coarseBackCorrMethod,fineBackCorrMethod,app.loadedMeas.getBackgroundData(),
            . . .
```

1 2

3

4

6

7

8

9

14

18

26	<pre>kernelSize,nT,manualValue);</pre>
27	<pre>switch(evalNorm)</pre>
28	case 'IS011146—1'
29	<pre>resData = corrData.iso11146Part1MeasurementData(centerRectangleX* pixelsize,centerRectangleY*pixelsize,sizeRectangleX*pixelsize, sizeRectangleY*pixelsize,convCrit,integAreaFactor,lambda,pixelsize);</pre>
30	<pre>res = Result(coarseBackCorrMethod,fineBackCorrMethod,evalNorm,lambda, convCrit,nT,integAreaFactor,kernelSize,resData);</pre>
31	app.loadedMeas.addResult(res);
32	UpdateGUIEvalResTab(app);
33	case 'IS011146-2'
34	end
35	uialert(app.UIFigure,{'Evaluation finished!'},'Next Step Information','Icon','
	<pre>success','CloseFcn','uiresume(gcbf)');</pre>
36	uiwait(gcbf)
37	catch ME
38	<pre>uialert(app.UIFigure,{'Evaluation failed! Check Evaluation Parameters!'},'Next Step Information' 'Loop' 'warning' 'CloseEcp' 'uiresume(gchf)');</pre>
39	uiwait(gcbf)
40	app.ConfEvalEvaluationButton.Enable = 'on':
41	app.ConfEvalEvaluationStatusLabel.BackgroundColor = [1 0.07 0.2];
42	app.ConfEvalEvaluationStatusLabel.Text = 'Standby';
43	drawnow;
44	rethrow(ME)
45	end
46	<pre>app.ConfEvalEvaluationButton.Enable = 'on';</pre>
47	
48	<pre>app.ConfEvalEvaluationStatusLabel.BackgroundColor = [1 0.07 0.2];</pre>
49	<pre>app.ConfEvalEvaluationStatusLabel.Text = 'Standby';</pre>
50	drawnow;
51	end

 $7.39: Function \ ConfEvalLoadMeasButton_ButtonPushedFcn$

```
function ConfEvalLoadMeasButton_ButtonPushedFcn(app, src, event)
%ConfEvalLoadMeasButton_ButtonPushedFcn listening to the Load Measurement
%button and loading a already performed Measurement with or without already
%existing results.
[baseFileName, folder] = uigetfile('*.mat');
fullFileName = fullfile(folder, baseFileName);
if exist(fullFileName, 'file')
    % Normal situation — they picked an existing file.
    temp = struct2cell(load(fullFileName));
    app.loadedMeas = temp{1};
    \% Now do something with storedStructure, like extract fields into new variables
       or whatever you want.
else
    % Error: Would only get here if they typed in a name of a non-existant file
    % instead of picking one from the folder.
    warningMessage = sprintf('Warning: mat file does not exist:\n%s', fullFileName);
```

1

2 3

4

6

7

8

9

12

```
uiwait(errordlg(warningMessage));
17
        return;
18
   end
19
   UpdateGUIConfEvalTab(app);
   if not(isempty(app.loadedMeas.getResults()))
        UpdateGUIEvalResTab(app);
23
   else
24
        ClearGUIEvalResTab(app);
25
   end
26
27
28
   end
```

7.40: Function ConfEvalSmpDropDown_ValueChangedFcn

```
function ConfEvalSmpDropDown_ValueChangedFcn(app, src, event)
 1
2
   %ConfEvalSmpDropDown_ValueChangedFcn listengin to changes of the selected
   %Measuring Point and adjusting the shown data in the UI.
   app.ConfEvalEvaluationButton.Enable = 'on';
4
   newValueDataSet = event.Value.getDataSet();
6
   [Useless NoI] = size(newValueDataSet);
 7
   cla(app.ConfEvalPreviewImUIAxes, 'reset');
8
   app.ConfEvalSmpImDropDown.Items = {};
9
   app.ConfEvalSmpImDropDown.ItemsData = [];
   for j=1:NoI
       app.ConfEvalSmpImDropDown.Items = [app.ConfEvalSmpImDropDown.Items sprintf('%i',
           i)];
12
       app.ConfEvalSmpImDropDown.ItemsData = [app.ConfEvalSmpImDropDown.ItemsData
           newValueDataSet(j)];
   end
   % Display image and stretch to fill axes
14
   imshow(newValueDataSet(1).getImageData(),...
        'Parent', app.ConfEvalPreviewImUIAxes);
   app.ConfEvalPreviewImUIAxes.Toolbar.Visible = 'off';
18
   colormap(app.ConfEvalPreviewImUIAxes, jet(256));
19
   app.ConfEvalPreviewImUIAxes.Interactions = [];
20
   if not(isempty(app.ConfEvalSelectedArea))
21
       app.ConfEvalRectRoi = drawrectangle(app.ConfEvalPreviewImUIAxes,'LabelVisible','
           hover','Label','OuterRectangle','Position',app.ConfEvalSelectedArea);
       addlistener(app.ConfEvalRectRoi, 'MovingROI',@(src, evt) drawRectAllevents(app,
           src,evt));
       addlistener(app.ConfEvalRectRoi, 'ROIMoved',@(src, evt) drawRectAllevents(app,src
           ,evt));
       centerRectangleX = app.ConfEvalRectRoi.Position(1)+app.ConfEvalRectRoi.Position
           (3)/2;
       centerRectangleY = app.ConfEvalRectRoi.Position(2)+app.ConfEvalRectRoi.Position
           (4)/2;
       app.ConfEvalPointRoi = drawpoint(app.ConfEvalPreviewImUIAxes,...
```

28	<pre>'Position',[centerRectangleX centerRectangleY],'Color','r',</pre>
29	'InteractionsAllowed','none');
30	end
31	drawnow;
32	drawnow;
33	end

	7.41: Function ConfEvalSmpImDropDown_ValueChangedFcn
1	<pre>function ConfEvalSmpImDropDown_ValueChangedFcn(app, src, event)</pre>
2	%ConfEvalSmpImDropDown_ValueChangedFcn listening to changes to the selected
3	%image and adjusting shown elements in the UI.
4	<pre>cla(app.ConfEvalPreviewImUIAxes, 'reset');</pre>
5	<pre>imshow(event.Value.getImageData(), 'Parent', app.ConfEvalPreviewImUIAxes);</pre>
6	<pre>app.ConfEvalPreviewImUIAxes.Toolbar.Visible = 'off';</pre>
7	<pre>app.ConfEvalPreviewImUIAxes.Interactions = [];</pre>
8	<pre>colormap(app.ConfEvalPreviewImUIAxes, jet(256));</pre>
9	<pre>if not(isempty(app.ConfEvalSelectedArea))</pre>
10	<pre>app.ConfEvalRectRoi = drawrectangle(app.ConfEvalPreviewImUIAxes,'LabelVisible','</pre>
	hover','Label','OuterRectangle','Position',app.ConfEvalSelectedArea);
11	addlistener(app.ConfEvalRectRoi,'MovingROI',@(src, evt)
	drawConfEvalRectAllevents(app,src,evt));
12	addlistener(app.ConfEvalRectRoi,'ROIMoved',@(src, evt) drawConfEvalRectAllevents
	(app,src,evt));
13	
14	<pre>centerRectangleX = app.ConfEvalRectRoi.Position(1)+app.ConfEvalRectRoi.Position</pre>
	(3)/2;
15	centerRectangleY = app.ConfEvalRectRoi.Position(2)+app.ConfEvalRectRoi.Position
1.0	(4)/2;
17	app.confevalPointRol = drawpoint(app.confevalPreviewImulAxes,
10	Transactions (centerRectanglex centerRectangler), (otor, 'r',
10	interactionsActowed , none);
19 20	end drawnow:
$\frac{20}{21}$	drawnow;
22	cla(app.liveMeasResultImUIAxes.'reset'):
23	<pre>imshow(event.Value.getImageData(), 'Parent', app.liveMeasResultImUIAxes):</pre>
24	<pre>% el = drawellipse(app.LiveMeasResultImUTAxes.'Center'</pre>
25	% [1282/2 1026/2]
26	% 'SemiAxes',[100 100],
27	<pre>% 'StripeColor','r','Linewidth',1.75);</pre>
28	<pre>% el.InteractionsAllowed = 'none';</pre>
29	end

7.42: Function drawConfEvalRectAllevents

```
1 function drawConfEvalRectAllevents(app,src,evt)
2 %drawRectAllevents listening to the Rectangle object and resolving all
3 %events happening to it.
4 evname = evt.EventName;
```

```
TU Bibliothek, Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
WIEN vour knowledge hub.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
switch(evname)
6
       case{'MovingROI'}
           disp(['ROI moving previous position: ' mat2str(evt.PreviousPosition)]);
8
           disp(['ROI moving current position: ' mat2str(evt.CurrentPosition)]);
9
       case{'ROIMoved'}
           disp(['ROI moved previous position: ' mat2str(evt.PreviousPosition)]);
           disp(['ROI moved current position: ' mat2str(evt.CurrentPosition)]);
12
   end
14
   app.ConfEvalSelectedArea = evt.CurrentPosition;
   centerRectangleX = app.ConfEvalRectRoi.Position(1)+app.ConfEvalRectRoi.Position(3)
       /2;
   centerRectangleY = app.ConfEvalRectRoi.Position(2)+app.ConfEvalRectRoi.Position(4)
       /2;
   app.ConfEvalPointRoi.Position(1) = centerRectangleX;
18
   app.ConfEvalPointRoi.Position(2) = centerRectangleY;
19
```

end

 $7.43: \ Function \ drawLiveMeasRectAllevents$

```
function drawLiveMeasRectAllevents(app,src,evt)
2
   %drawRectAllevents listening to the Rectangle object and resolving all
3
   %events happening to it.
   evname = evt.EventName;
4
   switch(evname)
6
       case{'MovingROI'}
7
           disp(['ROI moving previous position: ' mat2str(evt.PreviousPosition)]);
8
           disp(['ROI moving current position: ' mat2str(evt.CurrentPosition)]);
9
       case{'ROIMoved'}
           disp(['ROI moved previous position: ' mat2str(evt.PreviousPosition)]);
           disp(['ROI moved current position: ' mat2str(evt.CurrentPosition)]);
   end
14 app.LiveMeasSelectedArea = evt.CurrentPosition;
   centerRectangleX = app.LiveMeasRectRoi.Position(1)+app.LiveMeasRectRoi.Position(3)
       /2;
   centerRectangleY = app.LiveMeasRectRoi.Position(2)+app.LiveMeasRectRoi.Position(4)
       /2;
   app.LiveMeasPointRoi.Position(1) = centerRectangleX;
18
   app.LiveMeasPointRoi.Position(2) = centerRectangleY;
19
   end
```

7.44: Function EvalResImChangeColorMap

```
1 function EvalResImChangeColorMap(app, src, event)
2 %changes the colormap limits of the EvalResImDataUIAxes plot
3 app.EvalResColorMap(1) = app.EvalResColorMinEditField.Value
4 app.EvalResColorMap(2) = app.EvalResColorMaxEditField.Value
```

caxis(app.EvalResImDataUIAxes,app.EvalResColorMap);

end

7.45: Function EvalResImDataChangeColorMapButton ValueChangedFcn

```
function EvalResImDataChangeColorMapButton_ValueChangedFcn(app, src, event)
1
2
   %HSCamConfCamPreviewButton_ValueChangedFcn ValueChangedFcn listening to the
   %state button to enable and disable the preview of the camera.
4
   if src.Value
6
       colormap(app.EvalResImDataUIAxes, gray(256));
7
       caxis(app.EvalResImDataUIAxes,app.EvalResColorMap);
8
       return;
   end
   colormap(app.EvalResImDataUIAxes, jet(256));
11
   caxis(app.EvalResImDataUIAxes,app.EvalResColorMap);
   end
```

7.46: Function EvalResImDataDropDown_ValueChangedFcn

1	<pre>function EvalResImDataDropDown_ValueChangedFcn(app, src, event)</pre>
2	%EvalResImDataDropDown_ValueChangedFcn listening to changes of the selected
3	%image and ajusting data shown in ui to the new selection
4	<pre>SetEvalResImDataParams(app,event.Value);</pre>
5	end

7.47: Function EvalResMeasDataResultDropDown_ValueChangedFcn

<pre>function EvalResMeasDataResultDropDown_ValueChangedFcn(app, src, event)</pre>	
%EvalResMeasDataResultDropDown_ValueChangedFcn listening to the result	
%dropdown menu adjusting shown data in the ui for the new selection.	
<pre>SetEvalResMeasDataParams(app,event.Value);</pre>	
end	

7.48: Function EvalResMeasPlotButton ButtonPushedFcn

```
function EvalResMeasPlotButton_ButtonPushedFcn(app, src, event)
2
   %EvalResMeasPlotButton_ButtonPushedFcn listening to Plot Selection button
3
   %to plot the chosen Plotoption.
4
   fontsizeLabels = 38;
   fontsizeLegend = 20;
   fontsizeArrowLabel = 26;
6
   fontsizeTickLabels = 22;
8
9
   offsetHleft = -5;
   offsetHright = -5;
   offsetArrowLeft = -5;
12
   offsetArrowHeight = -22;
14
   lineWidth = 2.5;
```

 $\mathbf{2}$

4

1

```
TU Bibliothek, Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
VIEN vour knowedge hub.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

45

47

48

17 18

19

20

23

24

25

27

28

```
plotSelection = app.EvalResPlotChooserDropDown.Value;
Mp = app.EvalResSmpDataDropDown.Value;
Mp.calcMeanImage();
im = Mp.getMeanImage().getImageData();
offset = 125;
if app.EvalResHoldPlotButton.Value
     hold on;
     ax=gca;
else
     fig=figure;
     ax=axes;
end
switch plotSelection
     case 'Plot Radius X of all MP'
         f = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData().
            getfxFit();
         dW = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData().
            getdWx();
         NoMP = size(app.EvalResMeasDataResultDropDown.Value.
            getEvaluatedMeasurementData().getMeasurementDataSet(),2);
         temp = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData()
             .getMeasurementDataSet();
         NoI = size(temp(1).getDataSet(),2);
         dWStd = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData
            ().getdWxStd();
         zPos = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData()
             .getzPos();
         corrMethod = app.EvalResMeasDataResultDropDown.Value.getBackgroundCorrMethod
            ();
         lambda = app.EvalResMeasDataResultDropDown.Value.getLambda();
         integAreaFact = app.EvalResMeasDataResultDropDown.Value.getIntegAreaFactor;
         legendText = sprintf('Corr Method: %s, \x03bb = %i nm, IAF = %.1f, NoMP = %i
             , NoI = %i',corrMethod,lambda,integAreaFact,NoMP,NoI);
         %
                   zPos = zPos(1:end-2);
                   dW = dW(1:end-2);
         %
                   dWStd = dWStd(1:end-2);
         %
         dW_min = dW_dWStd;
         dW_max = dW+dWStd;
         hold on;
         stdLowerLimit = line(zPos/le+3,dW_min/2,'Color','black','LineWidth',
            lineWidth);
         stdLowerLimit.Annotation.LegendInformation.IconDisplayStyle = 'off';
         stdUpperLimit = line(zPos/1e+3,dW_max/2,'Color','black','LineWidth',
```

	lineWidth);
56	<pre>stdUpperLimit.Annotation.LegendInformation.IconDisplayStyle = 'off';</pre>
57	
58	zPos2 = [zPos/le+3, fliplr(zPos/le+3)];
59	inBetween = [dW_min/2, fliplr(dW_max/2)];
60	<pre>stdArea = fill(zPos2, inBetween, [0.85 0.85 0.85],'DisplayName','Standard</pre>
	deviation');
61	<pre>% stdArea.Annotation.LegendInformation.IconDisplayStyle = 'off';</pre>
52 63	<pre>nlot(zPos/le+3 dW/2 'b' 'lineWidth' lineWidth 'DisplayName' legendText).</pre>
64	<pre>set(aca 'FontSize' fontsizeTickLabels):</pre>
65	legend('FontSize', fontsizeLegend):
56	
67	<pre>title('\$\textrm{Beamradius } R_{x}\$','Interpreter','latex','FontSize', fontsizeLabels)</pre>
68	<pre>ylabel('\$\textrm{Beamradius } R_{x} \textrm{ in } \mu m\$ ','Interpreter','</pre>
	<pre>latex', 'FontSize', fontsizeLabels);</pre>
69	<pre>xlabel('\$\textrm{z - Coordinate in } mm\$ ','Interpreter','latex','FontSize',</pre>
	<pre>fontsizeLabels);</pre>
70 c	ase 'Plot Radius Y of all MP'
71	<pre>f = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData().</pre>
	getfyFit();
72	<pre>dW = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData(). getdWy();</pre>
73	NoMP = size(app.EvalResMeasDataResultDropDown.Value.
	getEvaluatedMeasurementData().getMeasurementDataSet(),2);
74	<pre>temp = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData() .getMeasurementDataSet();</pre>
75	<pre>NoI = size(temp(1).getDataSet(),2);</pre>
76	<pre>dWStd = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData ().getdWyStd();</pre>
77	<pre>zPos = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData() .getzPos();</pre>
(8 70	
79	<pre>corrMethod = app.EvalResMeasDataResultDropDown.Value.getBackgroundCorrMethod ():</pre>
80	lambda = app.EvalResMeasDataResultDropDown.Value.getLambda():
81	integAreaFact = app.EvalResMeasDataResultDropDown.Value.getIntegAreaFactor:
82	<pre>legendText = sprintf('Corr Method: %s. \x03bb = %i nm. IAF = %.1f. NoMP = %i</pre>
	<pre>NoI = %i',corrMethod,lambda,integAreaFact,NoMP,NoI);</pre>
83	
84	$dW_min = dW_dWStd;$
85	$dW_max = dW+dWStd;$
86	hold on;
87	
88	<pre>stdLowerLimit = line(zPos/le+3,dW_min/2,'Color','black','LineWidth',</pre>
89	<pre>stdLowerLimit.Annotation.LegendInformation.IconDisplayStyle = 'off';</pre>
90	<pre>stdUpperLimit = line(zPos/le+3,dW_max/2,'Color','black','LineWidth',</pre>

```
lineWidth);
             stdUpperLimit.Annotation.LegendInformation.IconDisplayStyle = 'off';
             zPos2 = [zPos/1e+3, fliplr(zPos/1e+3)];
             inBetween = [dW_min/2, fliplr(dW_max/2)];
             stdArea = fill(zPos2, inBetween, [0.85 0.85 0.85], 'DisplayName', 'Standard
                deviation');
 96
             %
                       stdArea.Annotation.LegendInformation.IconDisplayStyle = 'off';
             plot(zPos/1e+3,dW/2,'b','LineWidth', lineWidth,'DisplayName',legendText);
             set(gca, 'FontSize', fontsizeTickLabels);
100
             legend('FontSize',fontsizeLegend);
             title('$\textrm{Beamradius } R_{y}$', 'Interpreter', 'latex', 'FontSize',
                fontsizeLabels)
             ylabel('$\textrm{Beamradius } R_{y} \textrm{ in } \mu m$ ','Interpreter','
                latex', 'FontSize', fontsizeLabels);
             xlabel('$\textrm{z - Coordinate in } mm$ ','Interpreter','latex','FontSize',
                fontsizeLabels);
106
         case 'Plot 2D Intensity Distribution X of selected MP'
             %Get Data from current MP
108
             xCenterInPixel = Mp.getBeamCenterX_mean/app.ConfEvalPixelsizeEditField.Value
                 ():
             yCenterInPixel = Mp.getBeamCenterY_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
             zPos = Mp.getZPos()/1e+3;
111
             dWxInPixel = Mp.getdWx_mean/app.ConfEvalPixelsizeEditField.Value()+offset;
             dWyInPixel= Mp.getdWy_mean/app.ConfEvalPixelsizeEditField.Value()+offset;
             x=xCenterInPixel—round(dWxInPixel/2):1:xCenterInPixel+round(dWxInPixel/2);
114
             y=ones(size(x))*yCenterInPixel;
             v=[x;y];
             %Rotate line into Mainaxis using the azimuth angle
             x_center = xCenterInPixel;
118
             y_center = yCenterInPixel;
             center = repmat([x_center; y_center], 1, length(x));
             theta = Mp.getAzimutPhi_mean();
121
             R = [cosd(theta) - sind(theta); sind(theta) cosd(theta)];
                                 % shift points in the plane so that the center of
             s = v - center;
                rotation is at the origin
             so = R*s;
                                 % apply the rotation about the origin
124
                                 % shift again so the origin goes back to the desired
             vo = so + center;
                center of rotation
             % this can be done in one line as:
             % vo = R*(v - center) + center
             % pick out the vectors of rotated x— and y—data
128
             x_rotated = vo(1,:);
             y_rotated = vo(2,:);
             % plot(x, y, 'k—', x_rotated, y_rotated, 'r—', x_center, y_center, 'bo');
```

```
Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
vour knowledge hub
The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
rounded_rotated_x=floor(x_rotated);
rounded_rotated_y=floor(y_rotated);
%creating data vektor on rotated line
z=[];
for i=1:size(rounded_rotated_x,2)
    z = [z,im(rounded_rotated_y(i),rounded_rotated_x(i))];
end
rounded_rotated_x = (rounded_rotated_x—xCenterInPixel)*app.
   ConfEvalPixelsizeEditField.Value;
%fitting gaus to data
f = fit(rounded_rotated_x.',z.','gauss1');
% plotting gaus plus data
plot(ax,rounded_rotated_x,z,'b','LineWidth',lineWidth);
set(gca, 'FontSize', fontsizeTickLabels);
hold on;
h=plot(f, 'r---');
set(h, 'LineWidth',lineWidth);
legend('Measurement Data','Gauss1 fit','FontSize',fontsizeLegend);
% hold on because we need to draw quite some things
hold on;
%Draw 1/e^2 Diameter lines and centerline
hLeft = xline(-(dWxInPixel-offset)*app.ConfEvalPixelsizeEditField.Value/2-
   offsetHleft, 'k—', 'LineWidth', lineWidth);
hLeft.Annotation.LegendInformation.IconDisplayStyle = 'off';
hRight = xline((dWxInPixel-offset)*app.ConfEvalPixelsizeEditField.Value/2-
   offsetHright, 'k—', 'LineWidth', lineWidth);
hRight.Annotation.LegendInformation.IconDisplayStyle = 'off';
%
          hCenter = xline(yCenterInPixel-yCenterInPixel, 'g---', 'LineWidth',
    1);
          hCenter.Annotation.LegendInformation.IconDisplayStyle = 'off';
%
xLeft = ones(1,size(z,2))*(-(dWxInPixel-offset)*app.
   ConfEvalPixelsizeEditField.Value/2);
yLeft = 0:\max(z)/(size(z,2)-1):\max(z);
[xiLeft,yiLeft] = polyxpoly(rounded_rotated_x,z,xLeft,yLeft);
xRight = ones(1,size(z,2))*((dWxInPixel-offset)*app.
   ConfEvalPixelsizeEditField.Value/2);
yRight = 0:\max(z)/(size(z,2)-1):\max(z);
[xiRight,yiRight] = polyxpoly(rounded_rotated_x,z,xRight,yRight);
% Draw Doublearrow plus textbox for 1/e^2 Diameter
xArrow = [xiLeft-offsetArrowLeft xiRight-offsetArrowLeft];
yArrow = [yiLeft-offsetArrowHeight yiLeft-offsetArrowHeight];
```

```
Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
vour knowledge hub
The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

148

```
174
             annoArrow=annotation('doublearrow');
             annoArrow.Parent = ax;
             annoArrow.X = xArrow;
             annoArrow.Y = yArrow;
178
             annoArrow.LineWidth = 1;
             annoText=annotation('textbox');
181
             annoText.Parent = ax;
182
             annoText.Position = [(xArrow(1))/2 yArrow(1)+20 1 1];
183
             annoText.FitBoxToText = 'on';
184
             annoText.LineWidth = 1;
             annoText.String = '$D4\sigma$ Diameter';
             annoText.FontSize = fontsizeArrowLabel;
187
             annoText.Interpreter = 'latex';
             annoText.Position(1) = annoText.Position(1)—annoText.Position(3)/2;
189
             annoText.LineStyle = 'none';
190
             % Calc FWHM Diameter location
             xFHWM = min(rounded_rotated_x):0.001:max(rounded_rotated_x);
             gausFitFunction = feval(f,xFHWM);
             indexOfFWHM=find(abs(gausFitFunction-max(z)/2) < 0.01);</pre>
             indexOfFWHMLeft=indexOfFWHM(1);
196
             indexOfFWHMRight=indexOfFWHM(end);
             %Draw FWHM Diameter lines
200
             hLeftFWHM = xline(xFHWM(indexOfFWHMLeft), 'm-', 'LineWidth', lineWidth);
             hLeftFWHM.Annotation.LegendInformation.IconDisplayStyle = 'off';
             hRightFHWM = xline((xFHWM(indexOfFWHMRight)), 'm—', 'LineWidth', lineWidth)
                 :
203
             hRightFHWM.Annotation.LegendInformation.IconDisplayStyle = 'off';
204
205
206
             xiRight = xFHWM(indexOfFWHMRight);
207
             xiLeft = xFHWM(indexOfFWHMLeft);
208
209
             % Draw Doublearrow plus textbox for FWHM Diameter
210
             yiLeft = gausFitFunction(indexOfFWHMLeft);
211
             xArrow = [xiLeft xiRight];
212
             yArrow = [yiLeft yiLeft];
213
             annoArrow=annotation('doublearrow');
214
             annoArrow.Parent = ax;
215
             annoArrow.X = xArrow;
216
             annoArrow.Y = yArrow;
             annoArrow.LineWidth = 1;
218
             annoText=annotation('textbox');
             annoText.Parent = ax;
             annoText.Position = [(xArrow(1))/2 yArrow(1)+20 1 1];
```

```
annoText.FitBoxToText = 'on';
             annoText.LineWidth = 1;
224
             annoText.String = 'FWHM Diameter';
             annoText.FontSize = fontsizeArrowLabel;
             annoText.Interpreter = 'latex';
             annoText.Position(1) = annoText.Position(1)—annoText.Position(3)/2;
             annoText.LineStyle = 'none';
             titleText = sprintf('2D Intensity Distribution X (z = %.3f mm)',zPos);
             title(titleText, 'Interpreter', 'latex', 'FontSize', fontsizeLabels)
             ylabel('Intensity 0 - 255','Interpreter','latex','FontSize',fontsizeLabels);
             xlabel('$\textrm{Beamradius } R_{x} \textrm{ in } \mu m$ ','Interpreter','
                latex', 'FontSize', fontsizeLabels);
233
234
             hold off;
         case 'Plot 2D Intensity Distribution Y of selected MP'
             %Get Data from current MP
237
             xCenterInPixel = Mp.getBeamCenterX_mean/app.ConfEvalPixelsizeEditField.Value
                ();
             yCenterInPixel = Mp.getBeamCenterY_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
             zPos = Mp.getZPos()/1e+3;
             dWxInPixel = Mp.getdWx_mean/app.ConfEvalPixelsizeEditField.Value()+offset;
241
             dWyInPixel= Mp.getdWy_mean/app.ConfEvalPixelsizeEditField.Value()+offset;
242
             x=xCenterInPixel-round(dWxInPixel/2):1:xCenterInPixel+round(dWxInPixel/2);
243
             y=ones(size(x))*yCenterInPixel;
244
             v=[x;y];
             %Rotate line into Mainaxis using the azimuth angle
             x_center = xCenterInPixel;
             y_center = yCenterInPixel;
248
             center = repmat([x_center; y_center], 1, length(x));
249
             theta = Mp.getAzimutPhi_mean()+90;
250
             R = [cosd(theta) - sind(theta); sind(theta) cosd(theta)];
             s = v - center;
                                 % shift points in the plane so that the center of
                rotation is at the origin
252
                                 % apply the rotation about the origin
             so = R*s;
                                 % shift again so the origin goes back to the desired
             vo = so + center;
                center of rotation
254
             % this can be done in one line as:
             % vo = R*(v - center) + center
255
256
             % pick out the vectors of rotated x— and y—data
257
             x_rotated = vo(1,:);
258
             y_rotated = vo(2,:);
259
             % plot(x, y, 'k-', x_rotated, y_rotated, 'r-', x_center, y_center, 'bo');
             rounded_rotated_x=floor(x_rotated);
             rounded_rotated_y=floor(y_rotated);
             %creating data vektor on rotated line
             z=[];
             for i=1:size(rounded_rotated_x,2)
```

```
z = [z,im(rounded_rotated_y(i),rounded_rotated_x(i))];
             end
270
             rounded_rotated_y = (rounded_rotated_y-yCenterInPixel)*app.
                ConfEvalPixelsizeEditField.Value;
271
             %fitting gaus to data
272
             f = fit(rounded_rotated_y.',z.','gauss1');
             % plotting gaus plus data
             plot(ax,rounded_rotated_y,z,'b','LineWidth',lineWidth);
274
             set(gca, 'FontSize', fontsizeTickLabels);
276
             hold on;
             h=plot(f, 'r—');
277
278
             set(h, 'LineWidth',lineWidth);
             legend('Measurement Data','Gauss1 fit','FontSize',fontsizeLegend);
             % hold on because we need to draw quite some things
             hold on;
             %Draw 1/e^2 Diameter lines and centerline
             hLeft = xline(-(dWyInPixel-offset)*app.ConfEvalPixelsizeEditField.Value/2-
                offsetHleft, 'k-', 'LineWidth', lineWidth);
             hLeft.Annotation.LegendInformation.IconDisplayStyle = 'off';
             hRight = xline((dWyInPixel-offset)*app.ConfEvalPixelsizeEditField.Value/2-
                offsetHright, 'k-', 'LineWidth', lineWidth);
             hRight.Annotation.LegendInformation.IconDisplayStyle = 'off';
288
             %
                       hCenter = xline(yCenterInPixel—yCenterInPixel, 'g—', 'LineWidth',
                 1);
                       hCenter.Annotation.LegendInformation.IconDisplayStyle = 'off';
             %
290
291
             xLeft = ones(1,size(z,2))*(-(dWyInPixel-offset)*app.
                ConfEvalPixelsizeEditField.Value/2);
292
             yLeft = 0:\max(z)/(size(z,2)-1):\max(z);
             [xiLeft,yiLeft] = polyxpoly(rounded_rotated_y,z,xLeft,yLeft);
             xRight = ones(1,size(z,2))*((dWyInPixel-offset)*app.
                ConfEvalPixelsizeEditField.Value/2);
295
             yRight = 0:\max(z)/(size(z,2)-1):\max(z);
296
             [xiRight,yiRight] = polyxpoly(rounded_rotated_y,z,xRight,yRight);
297
             % Draw Doublearrow plus textbox for 1/e^2 Diameter
             xArrow = [xiLeft-offsetArrowLeft xiRight-offsetArrowLeft];
             yArrow = [yiLeft-offsetArrowHeight yiLeft-offsetArrowHeight];
             annoArrow=annotation('doublearrow');
             annoArrow.Parent = ax;
             annoArrow.X = xArrow;
             annoArrow.Y = yArrow;
             annoArrow.LineWidth = 1;
306
             annoText=annotation('textbox');
             annoText.Parent = ax;
```

```
Bibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar <sup>Vour knowledge hub</sup> The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
annoText.Position = [(xArrow(1))/2 yArrow(1)+20 1 1];
             annoText.FitBoxToText = 'on';
             annoText.LineWidth = 1;
             annoText.String = '$D4\sigma$ Diameter';
             annoText.FontSize = fontsizeArrowLabel;
314
             annoText.Interpreter = 'latex';
             annoText.Position(1) = annoText.Position(1)—annoText.Position(3)/2;
             annoText.LineStyle = 'none';
             % Calc FWHM Diameter location
318
             xFHWM = min(rounded_rotated_y):0.001:max(rounded_rotated_y);
             gausFitFunction = feval(f,xFHWM);
             indexOfFWHM=find(abs(gausFitFunction-max(z)/2) < 0.01);</pre>
             indexOfFWHMLeft=indexOfFWHM(1);
             indexOfFWHMRight=indexOfFWHM(end);
             %Draw FWHM Diameter lines
             hLeftFWHM = xline(xFHWM(indexOfFWHMLeft), 'm-', 'LineWidth', lineWidth);
             hLeftFWHM.Annotation.LegendInformation.IconDisplayStyle = 'off';
             hRightFHWM = xline((xFHWM(indexOfFWHMRight)), 'm-', 'LineWidth', lineWidth)
             hRightFHWM.Annotation.LegendInformation.IconDisplayStyle = 'off';
             xiRight = xFHWM(indexOfFWHMRight);
             xiLeft = xFHWM(index0fFWHMLeft);
             % Draw Doublearrow plus textbox for FWHM Diameter
             yiLeft = gausFitFunction(indexOfFWHMLeft);
             xArrow = [xiLeft xiRight];
             yArrow = [yiLeft yiLeft];
             annoArrow=annotation('doublearrow');
             annoArrow.Parent = ax;
             annoArrow.X = xArrow;
             annoArrow.Y = yArrow;
             annoArrow.LineWidth = 1;
             annoText=annotation('textbox');
             annoText.Parent = ax;
             annoText.Position = [(xArrow(1))/2 yArrow(1)+20 1 1];
             annoText.FitBoxToText = 'on';
             annoText.LineWidth = 1;
             annoText.String = 'FWHM Diameter';
             annoText.FontSize = fontsizeArrowLabel;
             annoText.Interpreter = 'latex';
             annoText.Position(1) = annoText.Position(1)—annoText.Position(3)/2;
             annoText.LineStyle = 'none';
             titleText = sprintf('2D Intensity Distribution Y (z = %.3f mm)',zPos);
```

```
title(titleText, 'Interpreter', 'latex', 'FontSize', fontsizeLabels)
             ylabel('Intensity 0 - 255','Interpreter','latex','FontSize',fontsizeLabels);
             xlabel('$\textrm{Beamradius } R_{y} \textrm{ in } \mu m$ ','Interpreter','
                latex', 'FontSize', fontsizeLabels);
             hold off:
         case 'Plot 3D Intensity Distribution of selected MP'
             xCenterInPixel = Mp.getBeamCenterX_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
             yCenterInPixel = Mp.getBeamCenterY_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
             zPos = Mp.getZPos()/1e+3;
             dWxInPixel = Mp.getdWx_mean/app.ConfEvalPixelsizeEditField.Value()+10;
             dWyInPixel= Mp.getdWy_mean/app.ConfEvalPixelsizeEditField.Value()+10;
             titleText = sprintf('3D Intensity Distribution (z = %.3f mm)',zPos);
             if dWxInPixel > dWyInPixel
                 dWsym = dWxInPixel;
             else
                 dWsym = dWyInPixel;
             end
             offset=200;
             im = im+offset;
             centerY = (round(yCenterInPixel+dWsym)-round(yCenterInPixel-dWsym))/2;
378
             centerX = (round(xCenterInPixel+dWsym)-round(xCenterInPixel-dWsym))/2;
             s = surf(ax,im(round(yCenterInPixel-dWsym):round(yCenterInPixel+dWsym),...
                 round(xCenterInPixel-dWsym):round(xCenterInPixel+dWsym)));
381
             s.XData = (s.XData-centerX)*app.ConfEvalPixelsizeEditField.Value;
             s.YData = (s.YData-centerY)*app.ConfEvalPixelsizeEditField.Value;
             set(gca,'xlim',[s.XData(1) s.XData(end)],'ylim',[s.YData(1) s.YData(end)]);
             s.EdgeColor = 'none';
             set(gca, 'FontSize', fontsizeTickLabels);
             X = s.XData';
             zFFT = s.YData;
             ZZ = s.ZData;
             [XX YY] = meshgrid(X,zFFT);
             hold on
             ims = imagesc(im(round(yCenterInPixel-dWsym):round(yCenterInPixel+dWsym),...
                 round(xCenterInPixel-dWsym):round(xCenterInPixel+dWsym)));
             ims.XData = (ims.XData-centerX)*app.ConfEvalPixelsizeEditField.Value;
             ims.YData = (ims.YData-centerY)*app.ConfEvalPixelsizeEditField.Value;
             spacing = 10; % play around so it fits the size of your data set
396
             for i = 1 : spacing : length(XX(:,1))
                 plot3(XX(:,i), YY(:,i), ZZ(:,i), '-k');
                 plot3(XX(i,:), YY(i,:), ZZ(i,:), '-k');
             end
400
             set(gca,'ZTickLabel',[]);
             title(titleText, 'Interpreter', 'latex', 'FontSize', fontsizeLabels)
             %
                       zlabel('$\textrm{Intensity}$ ','Interpreter','latex','FontSize
```

```
',16);
             xlabel('$\textrm{x - Coordinate in } \mu m$', 'Interpreter', 'latex', 'FontSize
                 ',25);
             ylabel('$\textrm{y - Coordinate in } \mu m$','Interpreter','latex','FontSize
                 ',25);
406
             xh = get(gca, 'XLabel'); % Handle of the x label
             set(xh, 'Units', 'Normalized')
408
             pos = get(xh, 'Position');
             set(xh, 'Position',pos.*[1.2,-0.3,1],'Rotation',21)
410
             yh = get(gca, 'YLabel'); % Handle of the y label
             set(yh, 'Units', 'Normalized')
411
412
             pos = get(yh, 'Position');
             set(yh, 'Position',pos.*[0.6,-0.2,1],'Rotation',-33)
413
414
             axis image;
415
             a = colorbar('FontSize',20);
             a.Ticks = linspace(200, 455, 6);
418
             a.TickLabels = linspace(0, 255, 6);
             ylabel(a,'Intensity 0-255 (false color)','Interpreter','latex','FontSize',
                 fontsizeLabels);
             colormap(jet(256));
             caxis([0+offset 255+offset]);
         case 'Plot Spatial Spectrum of selected MP'
             dx=app.ConfEvalPixelsizeEditField.Value;
             dy=app.ConfEvalPixelsizeEditField.Value;
             zPos = Mp.getZPos()/1e+3;
428
                              % centimeters per pixel
             Fs_x = 1/dx;
             Fs_y = 1/dy;
             [M, N, \sim] = size(im);
                                           % pixels
             x = dx * (0:N-1)';
                                             % centimeters
             y = dy * (0:M-1)';
             dFx = Fs_x/N;
                                         % cycles per centimeter
             dFy = Fs_y/M;
             Fx = (-Fs_x/2:dFx:Fs_x/2-dFx)';
                                                  % cycles per centimeter
             Fy = (-Fs_y/2:dFy:Fs_y/2-dFy)';
             zFFT = fft2(im);
440
             absY=abs(fftshift(zFFT));
441
             imagesc(Fx,Fy,absY);
             set(gca, 'FontSize', fontsizeTickLabels);
             colormap(gray);
             c = colorbar('FontSize',fontsizeTickLabels);
445
             ylabel(c,'$\textrm{Intensity in Greyscale}$ ','Interpreter','latex','
                 FontSize',fontsizeLabels);
```

```
title('Spatial Spectrum Linear Amplitude', 'Interpreter', 'latex', 'FontSize',
                 fontsizeLabels)
             xlabel('$\textrm{Spatial frequency } f_{x} \textrm{ in } 1/\mu m$ ','
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
             ylabel('$\textrm{Spatial frequency } f_{y} \textrm{ in } 1/\mu m$ ','
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
             figure;
             absYlog=10*log10(absY./max(max(absY)));
             imagesc(Fx,Fy,absYlog);
454
             set(gca, 'FontSize', fontsizeTickLabels);
             colormap(gray);
             c = colorbar('FontSize',fontsizeTickLabels);
             ylabel(c,'$\textrm{Intensity in }dB$ ','Interpreter','latex','FontSize',
                 fontsizeLabels);
             title('Spatial Spectrum Logarithmic Amplitude', 'Interpreter', 'latex', '
                 FontSize',fontsizeLabels)
             xlabel('$\textrm{Spatial frequency } f_{x} \textrm{ in } 1/\mu m$ ','
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
             ylabel('$\textrm{Spatial frequency } f_{y} \textrm{ in } 1/\mu m$ ','
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
             k=0.0001;
464
             scaledP1 = absY.*k;
             c=255/log10(1+max(max(scaledP1)));
             absYlog2 = c*log10(1+scaledP1);
             figure;
             imagesc(Fx,Fy,absYlog2);
             set(gca, 'FontSize', fontsizeTickLabels);
             colormap(gray);
472
             c = colorbar('FontSize',fontsizeTickLabels);
473
             ylabel(c,'$\textrm{Intensity in }dB$ ','Interpreter','latex','FontSize',
                 fontsizeLabels);
474
             titleText = sprintf('Spatial Spectrum (z = %.3f mm)',zPos);
             title(titleText,'Interpreter','latex','FontSize',fontsizeLabels)
476
             xlabel('$\textrm{Spatial frequency } f_{x} \textrm{ in } 1/\mu m$ ','
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
477
             ylabel('$\textrm{Spatial frequency } f_{y} \textrm{ in } 1/\mu m$ ','
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
478
             grid on;
         case 'Plot 2D Spectral Distribution X of selected MP'
479
             %Get Data from current MP
481
             xCenterInPixel = Mp.getBeamCenterX_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
             yCenterInPixel = Mp.getBeamCenterY_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
             zPos = Mp.getZPos()/1e+3;
```

```
Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar Vour knowledge hub. The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
484
             dWxInPixel = Mp.getdWx_mean/app.ConfEvalPixelsizeEditField.Value()+offset;
             dWyInPixel= Mp.getdWy_mean/app.ConfEvalPixelsizeEditField.Value()+offset;
             x=xCenterInPixel-round(dWxInPixel/2):1:xCenterInPixel+round(dWxInPixel/2);
487
             y=ones(size(x))*yCenterInPixel;
             v=[x;y];
             %Rotate line into Mainaxis using the azimuth angle
490
             x_center = xCenterInPixel;
             y_center = yCenterInPixel;
             center = repmat([x_center; y_center], 1, length(x));
             theta = Mp.getAzimutPhi_mean();
             R = [cosd(theta) - sind(theta); sind(theta) cosd(theta)];
             s = v - center;
                                 % shift points in the plane so that the center of
                 rotation is at the origin
496
             so = R*s;
                                  % apply the rotation about the origin
             vo = so + center;
                                 % shift again so the origin goes back to the desired
                 center of rotation
             % this can be done in one line as:
             % vo = R*(v - center) + center
500
             % pick out the vectors of rotated x- and y-data
             x_rotated = vo(1,:);
             y_rotated = vo(2,:);
             % plot(x, y, 'k—', x_rotated, y_rotated, 'r—', x_center, y_center, 'bo');
             rounded_rotated_x=floor(x_rotated);
             rounded_rotated_y=floor(y_rotated);
506
             %creating data vektor on rotated line
             z=[];
             for i=1:size(rounded_rotated_x,2)
                 z = [z,im(rounded_rotated_y(i),rounded_rotated_x(i))];
             end
             dx=app.ConfEvalPixelsizeEditField.Value;
             dy=app.ConfEvalPixelsizeEditField.Value;
514
             Fs_x = 1/dx;
                              % centimeters per pixel
             Fs_y = 1/dy;
518
             [M, N] = size(z);
                                    % pixels
             x = dx * (0:N-1)';
                                             % centimeters
             dFx = Fs_x/N;
                                         % cycles per centimeter
             Fx = (-Fs_x/2:dFx:Fs_x/2-dFx)';
                                                  % cycles per centimeter
             zFFT = fft(z);
             P2 = abs(zFFT/N);
             P1 = P2(1:floor(N/2)+1);
528
             P1(2:end-1) = 2*P1(2:end-1);
             f = Fs_x * (0:(N/2))/N;
               plot(gca,f,P1,'LineWidth',lineWidth);
    %
```

```
Jibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
<sup>EN</sup> ver knowledge hub
The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
set(gca, 'FontSize', fontsizeTickLabels);
    %
     %
               titleText = sprintf('2D Spectral Distribution X (z = %.3f mm)',zPos);
     %
               title(titleText,'Interpreter','latex','FontSize',fontsizeLabels)
               ylabel('$\textrm{Intensity}\textrm{ in Greyscale}$ ','Interpreter','latex
     %
        ', 'FontSize', fontsizeLabels);
     %
               xlabel('$\textrm{Spatial frequency } f_{x} \textrm{ in } 1/\mu m$ ','
        Interpreter', 'latex', 'FontSize', fontsizeLabels);
    %
               grid on;
    %
538
    %
               figure;
    %
               absP1log=10*log10(P1./max(P1));
    %
               plot(gca,f,absP1log,'LineWidth',lineWidth);
     %
               set(gca, 'FontSize', fontsizeTickLabels);
542
    %
               titleText = sprintf('2D Spectral Distribution X (z = %.3f mm)',zPos);
               title(titleText,'Interpreter','latex','FontSize',fontsizeLabels)
    %
544
               ylabel('$\textrm{Intensity}\textrm{ in }dB$ ','Interpreter','latex','
     %
        FontSize',fontsizeLabels);
     %
               xlabel('$\textrm{Spatial frequency } f_{x} \textrm{ in } 1/\mu m$ ','
        Interpreter', 'latex', 'FontSize', fontsizeLabels);
    %
               grid on;
    %
548
    %
               figure;
             k=0.0001;
             scaledP1 = P1.*k;
             c=255/log10(1+max(max(scaledP1)));
             absPllog = c*log10(1+scaledP1);
             plot(ax,f,absPllog,'LineWidth',lineWidth);
             set(gca, 'FontSize', fontsizeTickLabels);
             titleText = sprintf('2D Spectral Distribution X (z = %.3f mm)',zPos);
             title(titleText, 'Interpreter', 'latex', 'FontSize', fontsizeLabels)
558
             ylabel('$\textrm{Intensity}\textrm{ in }dB$ ','Interpreter','latex','
                 FontSize',fontsizeLabels);
             xlabel('$\textrm{Spatial frequency } f_{x} \textrm{ in } 1/\mu m$ ','
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
             grid on;
             hold off;
         case 'Plot 2D Spectral Distribution Y of selected MP'
             %Get Data from current MP
             xCenterInPixel = Mp.getBeamCenterX_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
             yCenterInPixel = Mp.getBeamCenterY_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
             zPos = Mp.getZPos()/1e+3;
             dWxInPixel = Mp.getdWx_mean/app.ConfEvalPixelsizeEditField.Value()+offset;
             dWyInPixel= Mp.getdWy_mean/app.ConfEvalPixelsizeEditField.Value()+offset;
             x=xCenterInPixel-round(dWxInPixel/2):1:xCenterInPixel+round(dWxInPixel/2);
             y=ones(size(x))*yCenterInPixel;
```

```
572
             v=[x;y];
             %Rotate line into Mainaxis using the azimuth angle
             x_center = xCenterInPixel;
             y_center = yCenterInPixel;
             center = repmat([x_center; y_center], 1, length(x));
             theta = Mp.getAzimutPhi_mean()+90;
             R = [cosd(theta) - sind(theta); sind(theta) cosd(theta)];
             s = v - center;
                                 % shift points in the plane so that the center of
                 rotation is at the origin
             so = R*s;
                                 % apply the rotation about the origin
             vo = so + center;
                                 % shift again so the origin goes back to the desired
                 center of rotation
             % this can be done in one line as:
             % vo = R*(v - center) + center
             % pick out the vectors of rotated x— and y—data
             x_rotated = vo(1,:);
             y_rotated = vo(2,:);
             % plot(x, y, 'k—', x_rotated, y_rotated, 'r—', x_center, y_center, 'bo');
             rounded_rotated_x=floor(x_rotated);
             rounded_rotated_y=floor(y_rotated);
590
             %creating data vektor on rotated line
             z=[];
             for i=1:size(rounded_rotated_x,2)
                 z = [z,im(rounded_rotated_y(i),rounded_rotated_x(i))];
             end
596
             dx=app.ConfEvalPixelsizeEditField.Value;
             dy=app.ConfEvalPixelsizeEditField.Value;
                              % centimeters per pixel
600
             Fs_x = 1/dy;
             [M, N] = size(z);
                                     % pixels
             x = dx * (0:N-1)';
                                             % centimeters
             dFx = Fs_x/N;
                                         % cycles per centimeter
606
             Fx = (-Fs_x/2:dFx:Fs_x/2-dFx)';
                                                  % cycles per centimeter
             zFFT = fft(z);
610
             P2 = abs(zFFT/N);
             P1 = P2(1:floor(N/2)+1);
612
             P1(2:end-1) = 2*P1(2:end-1);
613
             f = Fs_x*(0:(N/2))/N;
614
               plot(gca,f,P1,'LineWidth',lineWidth);
     %
               set(gca, 'FontSize', fontsizeTickLabels);
615
     %
    %
               titleText = sprintf('2D Spectral Distribution Y (z = %.3f mm)',zPos);
    %
               title(titleText,'Interpreter','latex','FontSize',fontsizeLabels)
    %
618
               ylabel('$\textrm{Intensity}\textrm{ in Greyscale}$ ','Interpreter','latex
```

```
', 'FontSize', fontsizeLabels);
               xlabel('$\textrm{Spatial frequency } f_{y} \textrm{ in } 1/\mu m$ ','
    %
        Interpreter', 'latex', 'FontSize', fontsizeLabels);
    %
               grid on;
621
    %
    %
               figure;
    %
               absP1log=10*log10(P1./max(P1));
    %
               plot(gca,f,absP1log,'LineWidth',lineWidth);
625
               set(gca, 'FontSize', fontsizeTickLabels);
    %
               titleText = sprintf('2D Spectral Distribution Y (z = %.3f mm)',zPos);
    %
627
     %
               title(titleText, 'Interpreter', 'latex', 'FontSize', fontsizeLabels)
628
     %
               ylabel('$\textrm{Intensity}\textrm{ in }dB$ ','Interpreter','latex','
        FontSize',fontsizeLabels);
     %
               xlabel('$\textrm{Spatial frequency } f_{y} \textrm{ in } 1/\mu m$ ','
        Interpreter', 'latex', 'FontSize', fontsizeLabels);
     %
               grid on;
     %
               figure;
             k=0.0001;
             scaledP1 = P1.*k;
             c=255/log10(1+max(max(scaledP1)));
636
             absPllog = c*log10(1+scaledP1);
             plot(gca,f,absP1log,'LineWidth',lineWidth);
             set(gca, 'FontSize', fontsizeTickLabels);
             titleText = sprintf('2D Spectral Distribution Y (z = %.3f mm)',zPos);
640
             title(titleText, 'Interpreter', 'latex', 'FontSize', fontsizeLabels)
642
             ylabel('$\textrm{Intensity}\textrm{ in }dB$ ','Interpreter','latex','
                 FontSize',fontsizeLabels);
             xlabel('$\textrm{Spatial frequency } f_{y} \textrm{ in } 1/\mu m$ ','
643
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
             hold off;
         case 'Plot Close Up of Beamprofile'
             xCenterInPixel = Mp.getBeamCenterX_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
             yCenterInPixel = Mp.getBeamCenterY_mean/app.ConfEvalPixelsizeEditField.Value
                 ();
648
             zPos = Mp.getZPos()/1e+3;
             dWxInPixel = Mp.getdWx_mean/app.ConfEvalPixelsizeEditField.Value()+0;
649
             dWyInPixel= Mp.getdWy_mean/app.ConfEvalPixelsizeEditField.Value()+0;
             titleText = sprintf('Beamprofile Close - Up (z = %.3f mm)',zPos);
             if dWxInPixel > dWyInPixel
                 dWsym = dWxInPixel;
             else
                 dWsym = dWyInPixel;
             end
658
             centerY = (round(yCenterInPixel+dWsym)-round(yCenterInPixel-dWsym))/2;
             centerX = (round(xCenterInPixel+dWsym)-round(xCenterInPixel-dWsym))/2;
```

```
set(gca, 'FontSize', fontsizeTickLabels);
             ims = image(im(round(yCenterInPixel-dWsym):round(yCenterInPixel+dWsym),...
                 round(xCenterInPixel-dWsym):round(xCenterInPixel+dWsym)));
             axis image;
             ims.XData = (ims.XData-centerX)*app.ConfEvalPixelsizeEditField.Value;
664
             ims.YData = (ims.YData-centerY)*app.ConfEvalPixelsizeEditField.Value;
             title(titleText, 'Interpreter', 'latex', 'FontSize', fontsizeLabels)
             set(gca, 'FontSize', fontsizeTickLabels, 'GridLineStyle', '--', 'GridAlpha', 1, '
                GridColor','k');
                       zlabel('$\textrm{Intensity}$ ','Interpreter','latex','FontSize
             %
                 ',16);
             xlabel('$\textrm{x - Coordinate in } \mu m$','Interpreter','latex','FontSize
                 ',fontsizeLabels);
             ylabel('$\textrm{y - Coordinate in } \mu m$', 'Interpreter', 'latex', 'FontSize
                 ',fontsizeLabels);
             a = colorbar('FontSize',20);
672
             a.Ticks = linspace(0, 255, 6);
             a.TickLabels = linspace(0, 255, 6);
             ylabel(a,'Intensity 0 - 255 (false color)','Interpreter','latex','FontSize',
                fontsizeLabels);
             colormap(jet(256));
         case 'Plot Diameter X squared Datapoints with fitted curve'
             f = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData().
                getfxFit();
             dW = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData().
                getdWx();
             NoMP = size(app.EvalResMeasDataResultDropDown.Value.
                getEvaluatedMeasurementData().getMeasurementDataSet(),2);
             temp = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData()
                 .getMeasurementDataSet();
             NoI = size(temp(1).getDataSet(),2);
             dWStd = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData
                 ().getdWxStd();
             zPos = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData()
                 .getzPos();
             corrMethod = app.EvalResMeasDataResultDropDown.Value.getBackgroundCorrMethod
                 ():
             lambda = app.EvalResMeasDataResultDropDown.Value.getLambda();
             integAreaFact = app.EvalResMeasDataResultDropDown.Value.getIntegAreaFactor;
             legendText = sprintf('Corr Method: %s, \x03bb = %i nm, IAF = %.1f, NoMP = %i
                 , NoI = %i',corrMethod,lambda,integAreaFact,NoMP,NoI);
             hold on;
690
             j = plot(zPos,dW.^2, 'bx');
             set(j,'LineWidth', 0.75*lineWidth,'MarkerSize',15);
             h = plot(f, 'r-', zPos, dW.^2, 'bx');
             set(h,'LineWidth', lineWidth,'MarkerSize',15);
             uistack(j,'top')
             set(gca, 'FontSize', fontsizeTickLabels);
```

```
Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar vour knowledge hub.

Vour knowledge hub
```

```
696
             legend('FontSize',fontsizeLegend);
             title('','Interpreter','latex','FontSize',fontsizeLabels)
             ylabel('$\textrm{Beamdiameter } D_{x}^{2} \textrm{ in } \mu m$ ','
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
700
             xlabel('$\textrm{z - Coordinate in }\mu m$ ','Interpreter','latex','FontSize
                 ',fontsizeLabels);
         case 'Plot Diameter Y squared Datapoints with fitted curve'
             f = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData().
                 getfyFit();
             dW = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData().
                 getdWy();
             NoMP = size(app.EvalResMeasDataResultDropDown.Value.
                 getEvaluatedMeasurementData().getMeasurementDataSet(),2);
             temp = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData()
                 .getMeasurementDataSet();
706
             NoI = size(temp(1).getDataSet(),2);
             dWStd = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData
                 ().getdWyStd();
             zPos = app.EvalResMeasDataResultDropDown.Value.getEvaluatedMeasurementData()
                 .getzPos();
             corrMethod = app.EvalResMeasDataResultDropDown.Value.getBackgroundCorrMethod
                 ();
             lambda = app.EvalResMeasDataResultDropDown.Value.getLambda();
             integAreaFact = app.EvalResMeasDataResultDropDown.Value.getIntegAreaFactor;
             legendText = sprintf('Corr Method: %s, \x03bb = %i nm, IAF = %.1f, NoMP = %i
                 , NoI = %i',corrMethod,lambda,integAreaFact,NoMP,NoI);
             hold on;
             j = plot(zPos,dW.^2, bx');
             set(j,'LineWidth', 0.75*lineWidth,'MarkerSize',15);
718
             h = plot(f, 'r-', zPos, dW.^2, 'bx');
             set(h,'LineWidth', lineWidth,'MarkerSize',15);
             uistack(j,'top')
             set(gca, 'FontSize', fontsizeTickLabels);
             legend('FontSize',fontsizeLegend);
724
             title('','Interpreter','latex','FontSize',fontsizeLabels)
725
             ylabel('$\textrm{Beamdiameter } D_{y}^{2} \textrm{ in } \mu m$ ','
                 Interpreter', 'latex', 'FontSize', fontsizeLabels);
             xlabel('$\textrm{z - Coordinate in }\mu m$ ','Interpreter','latex','FontSize
                 ',fontsizeLabels);
727
    end
728
    ax.XGrid = 'on';
    ax.YGrid = 'on';
    end
```

Bibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar ^N vur knowledge hub

```
1 function EvalResSaveMeasButton_ButtonPushedFcn(app, src, event)
2 %EvalResSaveMeasButton_ButtonPushedFcn ButtonPushedFcn listening to the
3 %Save Button.
4 loadedMeas = app.loadedMeas;
5 uisave({'loadedMeas'});
```

```
6 end
```

2

3

4

6

7.50: Function EvalResSmpDataDropDown_ValueChangedFcn

```
1 function EvalResSmpDataDropDown_ValueChangedFcn(app, src, event)
2 %EvalResSmpDataDropDown_ValueChangedFcn ValueChangedFcn listening to
3 %changes to the currently shown Measuring Point in the Evaluation Results
4 %Tab and triggering the function to update Measuring Point data.
5 SetEvalResSmpDataParams(app,event.Value);
6 end
```

7.51: Function HSCamConfApplySettingButton_ButtonPushedFcn

```
function HSCamConfApplySettingButton_ButtonPushedFcn(app,src,event)
%HSCamConfApplySettingButton_ButtonPushedFcn ButtonPushedFcn listening to
%the Apply Button to adjust the ExposureTime of the Camera.
src = getselectedsource(app.vid);
src.ExposureTimeAbs = app.HSCamConfExposureTimeEditField.Value;
```

```
src.ExposureTimeRaw = app.HSCamConfExposureTimeEditField.Value;
```

 $7.52: Function \ HSC amConfCamPreview Button_Value Changed Fcn$

```
1
   function HSCamConfCamPreviewButton_ValueChangedFcn(app, src, event)
2
   %HSCamConfCamPreviewButton_ValueChangedFcn ValueChangedFcn listening to the
   %state button to enable and disable the preview of the camera.
4
   if src.Value
   app.HSCamConfCamUIAxes.Visible = 'on';
6
   app.vid.FramesPerTrigger = 1;
7
   vidsrc = getselectedsource(app.vid);
   delay = CalculatePacketDelay(app.vid, 30);
8
9
   vidsrc.PacketDelay = delay;
10 vidRes = app.vid.VideoResolution;
   nBands = app.vid.NumberOfBands;
12
   hImage = image( zeros(vidRes(2), vidRes(1),nBands),'Parent', app.HSCamConfCamUIAxes)
       ;
   drawnow;
   preview(app.vid,hImage);
14
   wait(app.vid);
16 else
17
   stoppreview(app.vid);
18
   wait(app.vid);
19
   app.HSCamConfCamUIAxes.Visible = 'off';
20 vidsrc = getselectedsource(app.vid);
21 delay = CalculatePacketDelay(app.vid, 1);
22
  vidsrc.PacketDelay = delay;
23 drawnow;
```

- 24end
- end

7

11

21

1

7.53: Function HSCamConfCamTestButton ButtonPushedFcn

```
function HSCamConfCamTestButton_ButtonPushedFcn(app, src, event)
 2
   %HSCamConfCamTestButton_ButtonPushedFcn ButtonPushedFcn listening to the
   %Trigger Image button to create Testimages with the current ExposureTime
4
   %Setting.
6
   % Initiate the acquisition.
   start(app.vid);
8
9
   % Trigger the acquisition.
   trigger(app.vid)
   % Wait for the acquisition to end.
   wait(app.vid, 10);
14
   % Determine the number frames acquired.
   frameslogged = app.vid.FramesAcquired;
18
   data = getdata(app.vid);
19
   imshow(data);
   colormap(gca, jet(256));
   uisave('data');
23
   end
```

7.54: Function HSCamConfConnectButton_ButtonPushedFcn

```
function HSCamConfConnectButton_ButtonPushedFcn(app, src, event)
2
   %HSCamConfConnectButton_ButtonPushedFcn listening to the Connect Button for
   %the camera.
4
   % Creates the camera connection and sets a few properties.
   imaqreset();
6
   pause(2);
   app.vid = videoinput('gige', 1, 'Mono8');
8
   src = getselectedsource(app.vid);
9
   app.vid.FramesPerTrigger = 1;
   % framesPerSecond = CalculateFrameRate(app.vid, app.vid.FramesPerTrigger)
12
   delay = CalculatePacketDelay(app.vid, 1)
   src.PacketDelay = delay;
14
   src.ExposureTimeAbs = 130;
   src.ExposureTimeRaw = 130;
   % Configure the trigger type.
18
   triggerconfig(app.vid, 'manual');
19
```

20	<pre>app.HSCamConfExposureTimeEditField.Value = src.ExposureTimeAbs;</pre>
$\frac{21}{22}$	vidRes = app.vid.VideoResolution;
23	app.ConfEvalResWidthEditField.Value = vidRes(1);
24	app.ConfEvalResHeightEditField.Value = vidRes(2);
25	app.LiveMeasResWidthEditField.Value = vidRes(1);
26	app.LiveMeasResHeightEditField.Value = vidRes(2);
27	
28	app.ConfEvalResWidthEditField.Enable = 'off';
29	app.ConfEvalResHeightEditField.Enable = 'off';
30	app.LiveMeasResWidthEditField.Enable = 'off';
31	app.LiveMeasResHeightEditField.Enable = 'off';
32	
33	<pre>app.HSCamConfConnectionLabel.BackgroundColor = [0.2 1 0.3];</pre>
34	<pre>app.HSCamConfConnectionLabel.Text = 'Connected';</pre>
35	drawnow;
36	
37	<pre>app.HSCamConfConnectButton.Enable = 'off';</pre>
38	app.HSCamConfDisconnectButton.Enable = 'on';
39	app.HSCamConfCamTestButton.Enable = 'on';
40	app.HSCamConfExposureTimeEditField.Enable = 'on';
41	app.HSCamConfApplySettingButton.Enable = 'on';
42	app.HSCamConfCamPreviewButton.Enable = 'on';
43	end

 $7.55: \ Function \ HSCamConfDisconnectButton_ButtonPushedFcn$

1	<pre>function HSCamConfDisconnectButton_ButtonPushedFcn(app, src, event)</pre>
2	%HSCamConfDisconnectButton_ButtonPushedFcn listening to the Disconnect Button for
3	%the camera.
4	% Deletes the camera connection and reseset the connection.
5	<pre>imaqreset();</pre>
6	
7	<pre>app.HSCamConfExposureTimeEditField.Enable = 'off';</pre>
8	
9	<pre>app.HSCamConfConnectionLabel.BackgroundColor = [1 0.07 0.2];</pre>
10	<pre>app.HSCamConfConnectionLabel.Text = 'Diconnected';</pre>
11	drawnow;
12	
13	<pre>app.HSCamConfConnectButton.Enable = 'on';</pre>
14	<pre>app.HSCamConfDisconnectButton.Enable = 'off';</pre>
15	<pre>app.HSCamConfCamTestButton.Enable = 'off';</pre>
16	<pre>app.HSCamConfExposureTimeEditField.Enable = 'off';</pre>
17	<pre>app.HSCamConfApplySettingButton.Enable = 'off';</pre>
18	<pre>app.HSCamConfCamPreviewButton.Enable = 'off';</pre>
19	end

 $7.56: Function \ HSS tepmotor ConfApply Settings Button_ButtonPushed Fcn$

1 function HSStepmotorConfApplySettingsButton_ButtonPushedFcn(app,src,event)

2	%HSStepmotorConfApplySettingsButton_ButtonPushedFcn listening to the Aplly Button
3	%for the Stepmotor Settings.
4	% Creates the Stepper object and applies the stepper properties.
5	<pre>if isempty(app.arduinoUNOShieldStepmotor)</pre>
6	<pre>app.arduinoUNOShieldStepmotor = stepper(app.arduinoUNOShield,1,app.</pre>
	HSStepmotorConfStepsPerRevolutionEditField.Value);
7	<pre>app.HSStepmotorConfStepsPerRevolutionEditField.Enable = 'off';</pre>
8	end
9	<pre>app.arduinoUNOShieldStepmotor.RPM = app.HSStepmotorConfRpmEditField.Value;</pre>
0	end

 $7.57: Function \ HSS tepmotor ConfComPort Refresh Button_ButtonPushedFcn$

```
function HSStepmotorConfComPortRefreshButton_ButtonPushedFcn(app,src,event)
%HSStepmotorConfComPortRefreshButton_ButtonPushedFcn listening to refresh
%button to check for avaiable COM Ports.
app.HSStepmotorConfComPortDropDown.Items = seriallist;
app.HSStepmotorConfComPortDropDown.ItemsData = seriallist;
end
```

 $7.58: Function \ HSS tepmotor ConfConnectButton_ButtonPushedFcn$

```
1
    function HSStepmotorConfConnectButton_ButtonPushedFcn(app,src,event)
2
   %HSStepmotorConfConnectButton_ButtonPushedFcn listening to the connect
3
   %button of the Steppermotor and creating a arduino connnectiom element.
4
   selComPort=app.HSStepmotorConfComPortDropDown.Value;
6
   if isempty(app.arduinoUNO)
7
   app.arduinoUNO = arduino(selComPort, 'Uno', 'Libraries', 'Adafruit\MotorShieldV2');
8
   configurePin(app.arduinoUNO, 'D3', 'pullup');
9
   app.arduinoUNOShield = addon(app.arduinoUNO,'Adafruit\MotorShieldV2');
   app.currPos = 0;
   app.HSStepmotorConfConnectionLabel.BackgroundColor = [0.2 1 0.3];
14
   app.HSStepmotorConfConnectionLabel.Text = 'Connected';
   drawnow;
17
   app.HSStepmotorConfConnectButton.Enable = 'off';
18
   app.HSStepmotorConfDisconnectButton.Enable = 'on';
19
   app.HSStepmotorConfRpmEditField.Enable = 'on';
   app.HSStepmotorConfApplySettingsButton.Enable = 'on';
21
   app.HSStepmotorConfStepsPerRevolutionEditField.Enable = 'on';
22
   end
   end
```

 $7.59: Function\ HST est Hardware Go To Position Button_ButtonPushed Fcn$

1function HSTestHardwareGoToPositionButton_ButtonPushedFcn(app,src,event)2%HSTestHardwareGoToPositionButton_ButtonPushedFcn listening to the Go

2

4

6

```
%Button of the HardwareSettingTest area. Makes the Steppermotor go to a
   %specified position
4
   if app.HSTestHardwareGoToPositionEditField.Value < 0</pre>
6
       uialert(app.UIFigure,{'YOU FOOL! One does not simply walk into mordor ... ahm i
           mean zero!'},'You Fool Information','Icon','warning','CloseFcn','uiresume(
           qcbf)');
7
       uiwait(gcbf)
8
       return;
9
   end
       moveToZpos(app,app.HSTestHardwareGoToPositionEditField.Value);
11
   end
```

7.60: Function HSTestHardwareGoToZeroButton ButtonPushedFcn

```
function HSTestHardwareGoToZeroButton_ButtonPushedFcn(app,src,event)
2
  %HSTestHardwareGoToZeroButton_ButtonPushedFcn listening to the GoToZero
  %button and making the stage go to the zero position.
  moveToZpos(app,0);
  end
```

7.61: Function HSTestHardwareReferenceStageButton_ButtonPushedFcn

```
function HSTestHardwareReferenceStageButton_ButtonPushedFcn(app,src,event)
%HSTestHardwareReferenceStageButton_ButtonPushedFcn listening to the
%Reference Stage button starting the referencing process of the stage.
disp('Started Stage referencing.');
app.HSTestHardwareMotorStatusLabel.BackgroundColor = [0.2 1 0.3];
app.HSTestHardwareMotorStatusLabel.Text = 'Motor On';
drawnow;
limitSwitchStatus = readDigitalPin(app.arduinoUNO, 'D3');
while limitSwitchStatus
    move(app.arduinoUNOShieldStepmotor,-1);
    limitSwitchStatus = readDigitalPin(app.arduinoUNO, 'D3');
end
app.currPos(1) = 0;
disp('Stage referencing is finished.');
app.AutoMeasIS0111146Part1CurrPosEditField.Value = app.currPos(1);
app.HSTestHardwareCurrPosEditField.Value = app.currPos(1);
app.currPosRoi.Position = [app.currPos(1) 0];
app.currPosRoiAutoMeas.Position = [app.currPos(1) 0];
app.HSTestHardwareMotorStatusLabel.BackgroundColor = [1 0.07 0.2];
app.HSTestHardwareMotorStatusLabel.Text = 'Motor Off';
drawnow;
end
```

7.62: Function LiveMeasBackgroundMeasButton ButtonPushedFcn

function LiveMeasBackgroundMeasButton_ButtonPushedFcn(app, src, event) |%LiveMeasBackgroundMeasButton_ButtonPushedFcn listening to the Measure $3 \mid$ %Background button and starting background measurement at the current

```
%position.
 4
   app.LiveMeasStartStopButton.Enable = 'off';
6
   app.LiveMeasBackgroundMeasButton.Enable = 'off';
   app.LiveMeas = Measurement(999999,...
8
        999999,...
9
        999999,...
        999999);
12
   % Reset old Measurementdata
   app.LiveMeas.reset();
14
   MeasurementPoint = SingleMeasurementPoint(app.currPos(1));
   for k = 1 : app.LiveMeasNoOfImEditField.Value
        app.AutoMeasIS0111146Part1ProgressCurrMeasPointMeasEditField.Value = int2str(k);
18
       MeasurementPoint.addMeasurement(SingleMeasurement(TriggerCameraImage(app)));
19
        pause(1);
   end
21
   app.LiveMeas.getBackgroundData().addSingleMeasurementPoint(MeasurementPoint);
   app.LiveMeasStartStopButton.Enable = 'on';
23
   app.LiveMeasBackgroundMeasButton.Enable = 'on';
24
   end
```

7.63: Function LiveMeasDrawRectButton_ButtonPushedFcn

```
1
   function LiveMeasDrawRectButton_ButtonPushedFcn(app, src, event)
2
   %LiveMeasDrawRectButton_ButtonPushedFcn listening to guessBeamarea button
   %to draw the first guess of the beam area and beamcenter.
4
   if not(isa(app.LiveMeasRectRoi,'images.roi.Rectangle')) || not(isvalid(app.
       LiveMeasRectRoi))
       app.LiveMeasRectRoi = drawrectangle(app.LiveMeasPreviewImUIAxes, 'LabelVisible', '
           hover',...
6
            'Label', 'beamareaGuess', 'Position', [1 1 200 200]);
       addlistener(app.LiveMeasRectRoi, 'MovingROI',@(src, evt)
           drawLiveMeasRectAllevents(app,src,evt));
8
       addlistener(app.LiveMeasRectRoi, 'ROIMoved',@(src, evt) drawLiveMeasRectAllevents
           (app,src,evt));
       LiveMeascenterRectangleX = app.LiveMeasRectRoi.Position(1)+app.LiveMeasRectRoi.
           Position(3)/2;
       LiveMeascenterRectangleY = app.LiveMeasRectRoi.Position(2)+app.LiveMeasRectRoi.
           Position(4)/2;
       app.LiveMeasPointRoi = drawpoint(app.LiveMeasPreviewImUIAxes,...
            'Position',[LiveMeascenterRectangleX LiveMeascenterRectangleY],'Color','r',
                . . .
            'InteractionsAllowed', 'none');
14
   end
   end
```

 $7.64: Function\ LiveMeasStartStopButton_ValueChangedFcn$

function LiveMeasStartStopButton_ValueChangedFcn(app, src, event)
```
TU Bibliothek, Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar

The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

```
2
   %LiveMeasStartStopButton_ValueChangedFcn ValueChangedFcn listening to the
   %state button to start and stop the livemeasurement.
 4
   if src.Value
       app.LiveMeasStatusLabel.Text = 'Active';
6
       app.LiveMeasStatusLabel.BackgroundColor = [0.2 1 0.3];
 7
       app.LiveMeasBackgroundMeasButton.Enable = 'off';
8
   else
9
       app.LiveMeasStatusLabel.BackgroundColor = [1 0.07 0.2];
       app.LiveMeasStatusLabel.Text = 'Inactive';
       app.LiveMeasBackgroundMeasButton.Enable = 'on';
12
   end
   coarseMethod = app.LiveMeasBackCorrMethodDropDown.Value;
14
   fineMethod = app.LiveMeasFineBackCorrMethodDropDown.Value;
   background = app.LiveMeas.getBackgroundData();
   kernelSize = app.LiveMeasKernelSizeEditField.Value;
   ntFactor = app.LiveMeasNtEditField.Value;
18
   %Messpunkt Nummer hier immer 1
19
   k = 1;
   centerRectangleX = app.LiveMeasRectRoi.Position(1)+app.LiveMeasRectRoi.Position(3)
       /2;
21
   centerRectangleY = app.LiveMeasRectRoi.Position(2)+app.LiveMeasRectRoi.Position(4)
       /2;
   sizeRectangleX = app.LiveMeasRectRoi.Position(3);
   sizeRectangleY = app.LiveMeasRectRoi.Position(4);
24
   convergeKriterium = app.LiveMeasConvergeCriterionEditField.Value;
25
   faktor = app.LiveMeasIntegAreaEditField.Value;
   pixelsize = app.LiveMeasPixelsizeEditField.Value;
27
   manualValue=app.LiveMeasFineCorrValueEditField.Value;
28
   app.vid.FramesPerTrigger = 1;
29
   vidsrc = getselectedsource(app.vid);
   delay = CalculatePacketDelay(app.vid, 1);
   vidsrc.PacketDelay = delay;
   while src.Value
34
       app.LiveMeasColorMap(2) = app.LiveMeasColorMaxEditField.Value;
       app.LiveMeasColorMap(1) = app.LiveMeasColorMinEditField.Value;
       singleMeasurement = SingleMeasurement(TriggerCameraImage(app));
       corrMeas = singleMeasurement.backgroundCorrectionSingleMeasurement(coarseMethod,
           fineMethod,background,kernelSize,ntFactor,k,manualValue);
38
       processedMeas = corrMeas.iso11146Part1SingleMeasurement(centerRectangleX*
           pixelsize.centerRectangleY*pixelsize.sizeRectangleX*pixelsize.sizeRectangleY
           *pixelsize,convergeKriterium,faktor,pixelsize);
41
       cla(app.LiveMeasResultImUIAxes, 'reset');
       imshow(processedMeas.getImageData(), 'Parent', app.LiveMeasResultImUIAxes);
       if app.LiveMeasChangeColorMapButton.Value
44
           colormap(app.LiveMeasResultImUIAxes, gray(256));
           caxis(app.LiveMeasResultImUIAxes,app.LiveMeasColorMap);
```

```
else
47
            colormap(app.LiveMeasResultImUIAxes, jet(256));
48
            caxis(app.LiveMeasResultImUIAxes,app.LiveMeasColorMap);
        end
        colorbar(app.LiveMeasResultImUIAxes, 'FontSize', 20);
        el = drawellipse(app.LiveMeasResultImUIAxes, 'Center',...
            [processedMeas.getbeamCenterX()/pixelsize processedMeas.getbeamCenterY()/
               pixelsize],...
            'SemiAxes',[processedMeas.getdWx()/2/pixelsize processedMeas.getdWy()/2/
               pixelsize],...
54
            'StripeColor', 'r', 'Linewidth', 1.75);
        el.InteractionsAllowed = 'none';
        app.LiveMeasImData_dWx.Value = processedMeas.getdWx();
        app.LiveMeasImData_dWy.Value = processedMeas.getdWy();
58
        app.LiveMeasImData_phi.Value = processedMeas.getAzimutPhi();
   end
   end
```

7.65: Function LiveMeasTriggerImageButton_ButtonPushedFcn

```
function LiveMeasTriggerImageButton_ButtonPushedFcn(app, src, event)
1
2
   %LiveMeasTriggerImageButton_ButtonPushedFcn ButtonPushedFcn listening to the
   %Trigger Image button to create Testimages with the current ExposureTime
3
4
   %Setting.
6
   % Initiate the acquisition.
7
   app.vid.FramesPerTrigger = 1;
8
   vidsrc = getselectedsource(app.vid);
9
   delay = CalculatePacketDelay(app.vid, 1);
   vidsrc.PacketDelay = delay;
   start(app.vid);
   % Trigger the acquisition.
14
   trigger(app.vid)
   % Wait for the acquisition to end.
   wait(app.vid, 10);
18
19
   % Determine the number frames acquired.
20
   frameslogged = app.vid.FramesAcquired;
   hImage = getdata(app.vid);
23
24
   cla(app.LiveMeasPreviewImUIAxes,'reset');
25
   imshow(hImage, 'Parent', app.LiveMeasPreviewImUIAxes);
26
   colormap(app.LiveMeasPreviewImUIAxes, jet(256));
27
28
   end
```

TU Bibliothek, Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar VIEN vour knowledge hub The approved original version of this thesis is available in print at TU Wien Bibliothek.

7.66: Function moveToZpos

```
function moveToZpos(app,zPos)
2
   %moveToZpos moves the stage to a specified location
3
   steps = CalcNumOfSteps(app,zPos);
4
   app.HSTestHardwareMotorStatusLabel.BackgroundColor = [0.2 1 0.3];
6
   app.HSTestHardwareMotorStatusLabel.Text = 'Motor On';
   drawnow;
8
9
   move(app.arduinoUNOShieldStepmotor,steps);
   app.currPos(1) = app.currPos(1)+steps*app.StepInkrement;
12
   app.AutoMeasIS0111146Part1CurrPosEditField.Value = app.currPos(1);
   app.HSTestHardwareCurrPosEditField.Value = app.currPos(1);
14
   app.currPosRoi.Position = [app.currPos(1) 0];
   app.currPosRoiAutoMeas.Position = [app.currPos(1) 0];
   app.HSTestHardwareMotorStatusLabel.BackgroundColor = [1 0.07 0.2];
18
   app.HSTestHardwareMotorStatusLabel.Text = 'Motor Off';
19
   drawnow;
```

20 **end**

$7.67: Function\ SetEvalResImDataParams$

```
function SetEvalResImDataParams(app,Mp)
1
2
   %SetEvalResImDataParams sets the Image Data Parameters in the Evaluation
3
   %Result Tab.
4
   app.EvalResImData_W_X.Value = Mp.getW_X();
6
   app.EvalResImData_W_Y.Value = Mp.getW_Y();
7
8
   app.EvalResImData_W_X_squared.Value = Mp.getW_X_squared();
9
   app.EvalResImData_W_Y_squared.Value = Mp.getW_Y_squared();
   app.EvalResImData_W_XY.Value = Mp.getW_XY();
12
   app.EvalResImData_dWx.Value = Mp.getdWx();
   app.EvalResImData_dWy.Value = Mp.getdWy();
14
   app.EvalResImData_phi.Value = Mp.getAzimutPhi();
   cla(app.EvalResImDataUIAxes, 'reset');
   imshow(Mp.getImageData(), 'Parent', app.EvalResImDataUIAxes);
17
18
   if app.EvalResImDataChangeColorMapButton.Value
19
   colormap(app.EvalResImDataUIAxes,gray(256));
20
   caxis(app.EvalResImDataUIAxes,app.EvalResColorMap);
   else
       colormap(app.EvalResImDataUIAxes,jet(256));
   caxis(app.EvalResImDataUIAxes,app.EvalResColorMap);
24
   end
25
   colorbar(app.EvalResImDataUIAxes, 'FontSize', 12);
  el = drawellipse(app.EvalResImDataUIAxes, 'Center',...
```

```
27 [Mp.getbeamCenterX()/app.ConfEvalPixelsizeEditField.Value Mp.getbeamCenterY()/
app.ConfEvalPixelsizeEditField.Value],...
28 'SemiAxes',[Mp.getdWx()/2/app.ConfEvalPixelsizeEditField.Value Mp.getdWy()/2/app
.ConfEvalPixelsizeEditField.Value],...
29 'StripeColor','r','Linewidth',1.75);
30 el.InteractionsAllowed = 'none';
31 end
```

```
7.68: Function SetEvalResMeasDataParams
```

```
1
   function SetEvalResMeasDataParams(app, result)
2
   %SetEvalResMeasDataParams sets the Result Parameters in the Evaluation
   %Result Tab.
   app.EvalResSmpDataDropDown.Items = {};
4
   app.EvalResSmpDataDropDown.ItemsData = [];
6
 7
8
9
   app.EvalResMeasData_BackCorrMethod.Value = result.getBackgroundCorrMethod();
   app.EvalResMeasData_Lambda.Value = result.getLambda();
11
   app.EvalResMeasData_ConvCrit.Value = result.getConvCriterion();
12
   app.EvalResMeasData_IntegAreaFactor.Value = result.getIntegAreaFactor();
   app.EvalResMeasData_Nt.Value = result.getNt();
14
   app.EvalResMeasData_EvalNorm.Value = result.getEvalNorm();
   app.EvalResMeasData_KernelSizeInPercent.Value = result.getKernelSizeInPercent();
17
   measData=result.getEvaluatedMeasurementData();
18
19
   app.EvalResMeasData_z_0_X.Value = measData.getZ0X();
   app.EvalResMeasData_z_0_Y.Value = measData.getZ0Y();
22
   if isreal(measData.getZrX())
       app.EvalResMeasData_z_R_X.Value = measData.getZrX();
24
   else
       app.EvalResMeasData_z_R_X.Value = 99999999;
   end
28
   if isreal(measData.getZrY())
       app.EvalResMeasData_z_R_Y.Value = measData.getZrY();
   else
       app.EvalResMeasData_z_R_Y.Value = 99999999;
   end
34
   if isreal(measData.getDSigma0x())
      app.EvalResMeasData_d_0_X.Value = measData.getDSigma0x();
   else
       app.EvalResMeasData_d_0_X.Value = 99999999;
38
   end
  if isreal(measData.getDSigma0y())
```

```
41
                           app.EvalResMeasData_d_0_Y.Value = measData.getDSigma0y();
             42
                    else
             43
                    end
             44
             47
             48
                    else
                    end
TU Bibliothek, Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar

The approved original version of this thesis is available in print at TU Wien Bibliothek.
             54
                    else
                    end
             58
                    else
                    end
             64
                    else
                    end
            68
            69
             70
                     for k = 1:NOMP
             74
                                     1:
                                     ];
             78
                    end
                     for k = 1:NOMP-1
             80
             81
                             end
             82
             83
             84
             85
                             else
```

```
app.EvalResMeasData_d_0_Y.Value = 99999999;
if isreal(measData.getThetaSigmaX())
    app.EvalResMeasData_theta_X.Value = measData.getThetaSigmaX();
    app.EvalResMeasData_theta_X.Value = 99999999;
if isreal(measData.getThetaSigmaY())
    app.EvalResMeasData_theta_Y.Value = measData.getThetaSigmaY();
    app.EvalResMeasData_theta_Y.Value = 99999999;
if isreal(measData.getMsquaredX())
    app.EvalResMeasData_M_squared_X.Value = measData.getMsquaredX();
    app.EvalResMeasData_M_squared_X.Value = 99999999;
if isreal(measData.getMsquaredY())
    app.EvalResMeasData_M_squared_Y.Value = measData.getMsquaredY();
    app.EvalResMeasData_M_squared_Y.Value = 99999999;
app.EvalResMeasData_M_squared_eff.Value = sqrt(app.EvalResMeasData_M_squared_X.Value
    *app.EvalResMeasData_M_squared_Y.Value);
dataSet=result.getEvaluatedMeasurementData().getMeasurementDataSet();
[Useless NoMP] = size(dataSet);
    SmP = dataSet(k);
    tempString=sprintf('Measuring Point %i / z-Pos: %.4f mm',k,SmP.getZPos()/1000);
    app.EvalResSmpDataDropDown.Items = [app.EvalResSmpDataDropDown.Items tempString
    app.EvalResSmpDataDropDown.ItemsData = [app.EvalResSmpDataDropDown.ItemsData SmP
    for j = k+1:NOMP
    SmP = dataSet(k);
    SmP2 = dataSet(j);
    if abs(SmP.getAzimutPhi_mean() - SmP2.getAzimutPhi_mean()) > 10
        app.EvalResSmpInfoTextArea.Visible = 'on';
```

```
87 app.EvalResSmpInfoTextArea.Visible = 'off';
88 end
90 end
90 SetEvalResSmpDataParams(app,dataSet(1))
92 end
```

7.69: Function SetEvalResSmpDataParams

```
1
   function SetEvalResSmpDataParams(app,SmP)
2
   %SetEvalResSmpDataParams sets the MeasuringPoint Parameters in the Evaluation
   %Result Tab.
   app.EvalResImDataDropDown.Items = {};
4
   app.EvalResImDataDropDown.ItemsData = [];
6
 7
   app.EvalResSmpData_W_X.Value = SmP.getW_X_mean();
8
   app.EvalResSmpData_W_X_std.Value = SmP.getW_X_std();
9
   app.EvalResSmpData_W_Y.Value = SmP.getW_Y_mean();
   app.EvalResSmpData_W_Y_std.Value = SmP.getW_Y_std();
11
   app.EvalResSmpData_W_X_squared.Value = SmP.getW_X_squared_mean();
12
   app.EvalResSmpData_W_X_squared_std.Value = SmP.getW_X_squared_std();
   app.EvalResSmpData_W_Y_squared.Value = SmP.getW_Y_squared_mean();
   app.EvalResSmpData_W_Y_squared_std.Value = SmP.getW_Y_squared_std();
14
   app.EvalResSmpData_W_XY.Value = SmP.getW_XY_mean();
   app.EvalResSmpData_W_XY_std.Value = SmP.getW_XY_std();
17
   app.EvalResSmpData_dWx.Value = SmP.getdWx_mean();
18
   app.EvalResSmpData_dWx_std.Value = SmP.getdWx_std();
19
   app.EvalResSmpData_dWy.Value = SmP.getdWy_mean();
   app.EvalResSmpData_dWy_std.Value = SmP.getdWy_std();
   app.EvalResSmpData_phi.Value = SmP.getAzimutPhi_mean();
22
   app.EvalResSmpData_phi_std.Value = SmP.getAzimutPhi_std();
   SmPData = SmP.getDataSet();
24
   [Useless NoI] = size(SmPData);
   for j=1:NoI
       app.EvalResImDataDropDown.Items = [app.EvalResImDataDropDown.Items sprintf('%i',
           i)];
28
       app.EvalResImDataDropDown.ItemsData = [app.EvalResImDataDropDown.ItemsData
           SmPData(j)];
29
   end
   SetEvalResImDataParams(app,SmPData(1));
   end
```

7.70: Function TriggerCameraImage

```
1 function image = TriggerCameraImage(app)
2 %triggerCameraImage Triggers image aquisition of the camera
3 for i=1 : 10
```

```
TU Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar

wie N. Vourknowledge hub

The approved original version of this thesis is available in print at TU Wien Bibliothek.
```

4

6

7 8

9

14

18

21

23

24

26

27

29

36

38

41

42

43

44

45

47

48

```
try
        start(app.vid);
        % Trigger the acquisition.
        trigger(app.vid)
        % Wait for the acquisition to end.
        wait(app.vid, 10);
        % Determine the number frames acquired.
        frameslogged = app.vid.FramesAcquired;
        image = getdata(app.vid);
        if ismatrix(image)
            break;
        end
    catch ME
        disp('HAD TO RESET CAM');
        imagreset();
        pause(2);
        app.vid = videoinput('gige', 1, 'Mono8');
        src = getselectedsource(app.vid);
        app.vid.FramesPerTrigger = 1;
        src.ExposureTimeAbs = 130;
        src.ExposureTimeRaw = 130;
        % Configure the trigger type.
        triggerconfig(app.vid, 'manual');
        app.HSCamConfCamTestButton.Enable = 'on';
%
          start(app.vid);
%
          % Trigger the acquisition.
%
%
          trigger(app.vid)
%
%
          % Wait for the acquisition to end.
%
          wait(app.vid, 2);
%
          % Determine the number frames acquired.
%
%
          frameslogged = app.vid.FramesAcquired;
%
%
          image = getdata(app.vid);
    end
end
end
```

1

2

4

6

7

8

9

7.71: Function UpdateGUIConfEvalTab function UpdateGUIConfEvalTab(app) 2 %UpdateGUIConfEvalTab Updates the whole ConfigureEvaluation Tab and its 3 %elements. 4 dataSet=app.loadedMeas.getMeasurementData().getMeasurementDataSet(); [Useless NoMP] = size(dataSet); 6 app.ConfEvalSmpDropDown.Items = {}; app.ConfEvalSmpDropDown.ItemsData = []; 8 app.ConfEvalSmpImDropDown.Items = {}; 9 app.ConfEvalSmpImDropDown.ItemsData = []; for i = 1:NoMP SmP = dataSet(i); 12tempString=sprintf('Measuring Point %i / z-Pos: %.4f mm',i,SmP.getZPos()/1000); 14 app.ConfEvalSmpDropDown.Items = [app.ConfEvalSmpDropDown.Items tempString]; app.ConfEvalSmpDropDown.ItemsData = [app.ConfEvalSmpDropDown.ItemsData SmP]; end SmP = dataSet(1);18 SmPData = SmP.getDataSet(); 19 [Useless NoI] = size(SmPData); 20 for j=1:NoI app.ConfEvalSmpImDropDown.Items = [app.ConfEvalSmpImDropDown.Items sprintf('%i', j)]; app.ConfEvalSmpImDropDown.ItemsData = [app.ConfEvalSmpImDropDown.ItemsData SmPData(j)]; end 24app.ConfEvalPreviewImUIAxes.Visible = 'on'; 25app.ConfEvalDrawRectButton.Enable = 'on'; app.ConfEvalSmpDropDown.Enable = 'on'; 27 app.ConfEvalSmpImDropDown.Enable = 'on'; 28imshow(SmPData(1).getImageData(), 'Parent', app.ConfEvalPreviewImUIAxes); 29 colormap(app.ConfEvalPreviewImUIAxes, jet(256)); app.ConfEvalPreviewImUIAxes.Toolbar.Visible = 'off'; app.ConfEvalPreviewImUIAxes.Interactions = []; end

7.72: Function UpdateGUIEvalResTab

```
function UpdateGUIEvalResTab(app)
%UpdateGUIEvalResTab Updates the whole EvaluationResult Tab and its
%elements.
resultDataSet=app.loadedMeas.getResults();
[Useless NoR] = size(resultDataSet);
app.EvalResMeasDataResultDropDown.ItemsData = {};
app.EvalResMeasDataResultDropDown.ItemsData = [];
for i = 1:NoR
    result = resultDataSet(i);
    tempString=sprintf('Result No.%i',i);
    app.EvalResMeasDataResultDropDown.Items = [app.EvalResMeasDataResultDropDown.
    Items tempString];
```

12	<pre>app.EvalResMeasDataResultDropDown.ItemsData = [app.EvalResMeasDataResultDropDown</pre>
	.ItemsData result];
13	end
14	<pre>SetEvalResMeasDataParams(app,resultDataSet(1));</pre>
15	
16	end

7.73: Function UpdatePreviewAxes

```
function UpdatePreviewAxes(app, src, event)
 2
   %UpdatePreviewAxes Updates the Measuring Point Axes in the Automated Measurement Tab
4
   if strcmp('on',app.AutoMeasIS0111146Part10ffseEditField.Enable)
        offset = app.AutoMeasIS0111146Part10ffseEditField.Value;
6
       while mod(offset,app.StepInkrement) ~= 0
 7
            offset = offset+1;
8
        end
        app.AutoMeasIS0111146Part10ffseEditField.Value = offset;
9
11
        timesRayLength = app.AutoMeasIS0111146Part1TimesRayleighEditField.Value;
12
        rayLength = app.AutoMeasIS0111146Part1RayleighEditField.Value;
        while mod(rayLength,app.StepInkrement) ~= 0
14
            rayLength = rayLength+1;
        end
        app.AutoMeasIS0111146Part1RayleighEditField.Value = rayLength;
17
        NoMP = app.AutoMeasIS0111146Part1NoMeasPointsEditField.Value;
18
19
        xTicks = rayLength*2*timesRayLength/(NoMP-1);
       while mod(xTicks,app.StepInkrement) ~= 0
            NOMP = NOMP + 1;
            xTicks = rayLength*2*timesRayLength/(NoMP-1);
        end
24
        app.AutoMeasIS0111146Part1NoMeasPointsEditField.Value = NoMP;
        app.measurementPoints = (offset-xTicks*(NoMP-1)/2:xTicks:offset+xTicks*(NoMP-1)
           /2);
        if isempty(app.centerMarker) || not(ishandle(app.centerMarker))
            app.centerMarker = line(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,
               offset,0,'Color','blue','Marker','o','MarkerSize',10,'LineStyle','none',
               'LineWidth',2);
        else
            app.centerMarker.XData = offset;
            app.centerMarker.YData = 0;
32
        end
        app.centerMarker.Visible = 'on';
34
   else
        area = app.AutoMeasIS0111146Part1MeasAreaEditField.Value;
       while mod(area,app.StepInkrement) ~= 0
            area = area+1;
```

```
38
                end
                app.AutoMeasIS0111146Part1MeasAreaEditField.Value = area;
                NoMP = app.AutoMeasIS0111146Part1NoMeasPointsEditField.Value;
       41
                xTicks = area/(NoMP-1);
                while mod(xTicks,app.StepInkrement) ~= 0
                    NOMP = NOMP + 1;
                    xTicks = area/(NoMP-1);
                end
                app.AutoMeasIS0111146Part1NoMeasPointsEditField.Value = NoMP;
       47
       48
                app.measurementPoints = (0:xTicks:xTicks*(NoMP-1));
                app.centerMarker.Visible = 'off';
Bibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar Vour knowledge hub The approved original version of this thesis is available in print at TU Wien Bibliothek.
           end
            y = zeros(1,size(app.measurementPoints,2));
       54
                app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XLim = [0-600 app.
                    maxValueStage+600];
                XTICK = unique([0,app.measurementPoints,app.maxValueStage]);
                app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XTick = XTICK;
                app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes.XTickLabel = XTICK;
       58
                hold(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes, 'on')
                if isempty(app.previewMarkers)
       62
                    app.previewMarkers = line(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes,
                         app.measurementPoints,y,'Color','red','Marker','x','MarkerSize',15,'
                         LineStyle', 'none', 'LineWidth',1);
                else
       64
                    app.previewMarkers.XData = app.measurementPoints;
                    app.previewMarkers.YData = y;
                end
                if isempty(app.endPointMarkers)
                    app.endPointMarkers = line(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes
                         ,[app.measurementPoints(1),app.measurementPoints(end)],[0,0],'Color','
                         green', 'Marker', 'o', 'MarkerSize', 10, 'LineStyle', 'none', 'LineWidth', 2);
                else
                    app.endPointMarkers.XData = [app.measurementPoints(1),app.measurementPoints(
                         end)];
                    app.endPointMarkers.YData = [0,0];
       72
                end
       74
                hold(app.AutoMeasIS0111146Part1PreviewMeasPointsUIAxes, 'off')
           if app.measurementPoints(1)<0 || app.measurementPoints(end)>app.maxValueStage
       78
                app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'off';
                return;
          end
```

```
81 app.AutoMeasIS0111146Part1StartMeasButton.Enable = 'on';
82
83 end
```

Anhang 2: Installationsanleitung

Damit das Messprogramm verwendbar ist, müssen Hardware und Software korrekt eingerichtet werden. Die notwendigen Schritte sind:

- 1. Installieren des Arduino Board Treibers CH341SER.EXE (notwendig damit der Rechner das Arduino Board im Geräte Manager erkennt)
- 2. Installieren von MATLAB Version R2020b
- 3. Installieren von Basler Pylon Viewer Version 6.1.1.19832
- 4. Verbinden der Basler Kamera über Ethernet
- 5. Den Schritten in Anhang 3 folgen um den Ethernet Adapter richtig zu konfigurieren
- 6. Anhang 4 folgen um MATLAB für die Verwendung der Basler Kamera einzurichten
- 7. Installieren der Arduino Library für MATLAB, damit direkt mit dem Board kommuniziert werden kann
- 8. Installieren der notwendigen MATLAB Packages:
 - Curve Fitting Toolbox Version 3.5.12 (R2020b)
 - Image Acquisition Toolbox Version 6.3 (R2020b)
 - Image Processing Toolbox Version 11.2 (R2020b)
 - Image Acquisition Toolbox Support Package for GigE Vision Hardware Version 20.2.1 (R2020b)
 - MATLAB Support Package for Arduino Hardware Version 20.2.0 (R2020b)

Sind alle Schritte abgearbeitet sollte es keine Probleme beim öffnen des Programms geben. Geöffnet wird das Programm über *lbcUI_Launcher.m.* Der Startbildschirm der geöffnet werden sollte ist in Abb. 7.1 dargestellt. Die weitere Bedienung ist bereits in der Arbeit selbst beschrieben.



Abbildung 7.1: Startbildschirm des Messprogramms

GigE Vision Quick Start Configuration Guide

1

Configure GigE Vision Image Acquisition on Windows	1-2
Recommended Hardware Setup	1-3
Gigabit Ethernet Network Adapter Configuration	1-4 1-4
Network Settings for the Camera's Network Connection	1-9
Firewalls and the GigE Vision Camera Network Connection	1-14
Configure Camera GigE Vision Streaming Parameters in MATLAB	1-15
"Configure GigE Vision Image Acquisition on Windows" on page 1-2	
"Recommended Hardware Setup" on page 1-3	
"Gigabit Ethernet Network Adapter Configuration" on page 1-4	

- "Firewalls and the GigE Vision Camera Network Connection" on page 1-14
- "Configure Camera GigE Vision Streaming Parameters in MATLAB" on page 1-15

Configure GigE Vision Image Acquisition on Windows

Acquiring images from a GigE Vision camera in Image Acquisition Toolbox[™] using the GigE Vision Hardware support package functionality (either the videoinput object using the gige adaptor or the gigecam object) requires specific configuration and setup for the Ethernet network adapter and network connection outside of MATLAB[®]. This guide describes the configuration steps to help you get started using a GigE Vision camera with the Image Acquisition Toolbox on a Windows[®] computer.

The configuration instructions and screen shots included are for Windows 7, but similar configuration steps are applicable for Windows 8 or Windows 10.

TU Bibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar wien wurknowledge hub The approved original version of this thesis is available in print at TU Wien Bibliothek.

Recommended Hardware Setup

The recommended getting-started configuration when using the Image Acquisition Toolbox to acquire images from a single GigE Vision camera consists of a system with:

- GigE Vision compliant camera.
- Computer with a camera-dedicated Gigabit Ethernet network adapter (for example a PCI Express Gigabit Ethernet network interface card), which supports jumbo frames (4k or 9k bytes are common jumbo frame sizes).
- Direct connection between the camera and dedicated Gigabit Ethernet adapter.
- For any additional computer network connection (such as an Internet connection or your organization's local area network) it is recommended to use a separate network adapter.

Although other hardware setups are possible, the above configuration ensures that:

- Maximum bandwidth is available for streaming images from the camera.
- The camera-dedicated private network connection can have network/firewall settings different than the Internet or domain network connection.

1-3

Gigabit Ethernet Network Adapter Configuration

- · "Gigabit Ethernet Adapter Driver" on page 1-4
- * "Network Settings for the Camera's Network Connection" on page 1-9

The following Ethernet adapter configuration settings are recommended for proper operation and optimum performance when acquiring images from a GigE Vision camera.

Gigabit Ethernet Adapter Driver

Verify that an appropriate Ethernet network adapter driver (provided by the network adapter manufacturer) is installed and working properly. Custom high-performance drivers installed for use with a third-party imaging application will not work with the Image Acquisition Toolbox videoinput and gigecam interfaces.

- 1 Open Windows Device Manager, and click on Network adapters.
- 2 Right-click on the Gigabit Ethernet adapter entry dedicated to the camera's network connection under **Network adapters**, and click **Properties** in the context menu.

1-4



3 In the Adapter Properties, click the **Driver** tab to verify the driver.

Intel(R) Gigabit CT Desktop Adapter Properties						
General Advanced Dr	iver Details Resources Power Management					
Intel(R) Gigabit	CT Desktop Adapter					
Driver Provider	: Intel					
Driver Date:	10/13/2011					
Driver Version:	11.14.48.0					
Digital Signer:	Microsoft Windows Hardware Compatibility Publisher					
Driver Details	To view details about the driver files.					
Update Driver	To update the driver software for this device.					
Roll Back Driver	If the device fails after updating the driver, roll back to the previously installed driver.					
<u>D</u> isable	Disables the selected device.					
<u>U</u> ninstall	To uninstall the driver (Advanced).					
	OK Cancel					

For optimum GigE Vision streaming performance, i.e. reduced CPU load and smaller likelihood of dropped frames, the following settings for the Gigabit Ethernet adapter jumbo packet and receive buffers are recommended:

- 1 In the network Adapter Properties (described in the previous section), click the **Advanced** tab.
- 2 Set **Jumbo Frame** (or **Jumbo Packet**) to the maximum supported value (for example 9014 bytes). Gigabit Ethernet controllers that support jumbo frames can transfer packet sizes larger than the standard Ethernet frame size (1500 bytes).

1-6



3 Set **Receive Buffers** (or **Receive Descriptors**) to the maximum supported value (for example 2048). For some Ethernet controller drivers this setting is grouped under **Performance Options**.



General	Advanced	Driver	Details	Resources	Power Manager	nent
The follo the prop on the ri	owing propert erty you wan ght.	ies are a t to chan	vailable fo nge on the	or this network e left, and the	k adapter. Click n select its value	
Property	:			<u>V</u> alue:		
Interrup IPv4 Cf Jumbo Large S Link Sp Locally Log Lin Priority Receiv Reduce TCP Cf	t Moderation hecksum Offl Packet Send Offload eed & Duple Administered k State Ever & VLAN e Buffers e Side Scalin e Side Scalin hecksum Offl	Rate oad (IPv4) (IPv6) x I Address It Ig Queue Power Do oad (IPv4	s E ss pwn 4) T	2048		÷

4 Confirm that Link Speed & Duplex is set to Auto Negotiation (or Auto Detect).

1-8

ntel(R) Gigabit CT Desktop Ada	apter Properties
General Advanced Driver I The following properties are avait the property you want to chang on the right. Property: Interrupt Moderation Rate IPv4 Checksum Offload Jumbo Packet Large Send Offload (IPv4) Large Send Offload (IPv6) Link Speed & Duplex Locally Administered Address Log Link State Event Priority & VLAN Receive Buffers Receive Side Scaling Receive Side Scaling Queues Reduce Speed On Power Dov TCP Checksum Offload (IPv4)	Details Resources Power Management ailable for this network adapter. Click e on the left, and then select its value Value:
	OK Cancel

Network Settings for the Camera's Network Connection

The following settings for the camera's dedicated network connection are recommended for proper operation and optimum performance when acquiring images from a GigE Vision camera.

- 1 In Windows Control Panel, open the Network and Sharing Center.
- 2 Click on Change adapter settings.

	Network and I	 Network an
	Control Panel Home Change adapter settings	View yo
	Change advanced sharing settings	MYC (This View your
		4
		Change y
		tininge y
		*
	See also HomeGroup	ન્સ્ટ્રે
	Internet Options	_
	Vector Hardware Windows Firewall	
	3 Right-click on	the came
1-10		
	1-10	See also HomeGroup Internet Options Vector Hardware Windows Firewall 3 Right-click on f

🔾 🗸 🦉 « Network and I	Network and Sharing Center
Control Panel Home	View your basic network information and set up connections
Change adapter settings	👰 💐 🍪 See full map
Change advanced sharing settings	MYCOMPUTER Multiple networks Internet (This computer)
	View your active networks Connect or disconnect
	Access type: Internet
	Domain network Connections: Connection
	Unidentified network Access type: No Internet access Work network Connections: Image: GigE
	Change your networking settings
	Set up a new connection or network
	Set up a wireless, broadband, dial-up, ad hoc, or VPN connection; or set up a router or access point.
	Connect to a network
	Connect or reconnect to a wireless, wired, dial-up, or VPN network connection.
	Choose homegroup and sharing options
See also	Access files and printers located on other network computers, or change sharing settings.
HomeGroup	showing sectings.
Vector Hardware	Troubleshoot problems
ctor riardware	Diagnose and repair network problems, or get troubleshooting information.

Right-click on the camera's network connection and click **Properties**.



4 In the **Networking** tab, confirm that **Internet Protocol Version 4 (TCP/IPv4)** is enabled/checked, and uncheck all other options.

When a dedicated camera network connection is used, other protocols, clients, or services can be disabled/unchecked, as they are not necessary for GigE Vision control and streaming.

GigE Properties
Networking Sharing
Connect using:
Intel(R) Gigabit CT Desktop Adapter
Configure
This connection uses the following items:
VMware Bridge Protocol
QoS Packet Scheduler
Listenst Protocol Version 6 (TCP (IPv6)
Internet Protocol Version 6 (TCP/IPV6)
Link-Layer Topology Discovery Mapper I/O Driver
🗆 🔺 Link-Layer Topology Discovery Responder 🔹
Install Uninstall Properties
Description
Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.
OK Cancel

5 To configure the IP address, select **Internet Protocol Version 4 (TCP/IPv4)** and right-click **Properties**.

To configure automatic IP address assignment for the camera network connection, on the **General** tab, select **Obtain an IP address automatically** and **Obtain DNS server address automatically**, and click **OK**.

TU Sibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar With Wurknowleden hub The approved original version of this thesis is available in print at TU Wien Bibliothek.

General	Alternate Cor	nfiguration					
You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.							
0	btain an IP add	ress automa	atically				
_© U	l <u>s</u> e the following	IP address:					
<u>I</u> P a	ddress:				1.		
S <u>u</u> b	net mask:						
Def	ault gateway:						
0) <u>b</u> tain DNS serve	er address a	utomati	cally			
_© U	ls <u>e</u> the following	DNS server	addres	ses:			
Pref	erred DNS serv	er:					
<u>A</u> lte	rnate DNS serv	er:			1		
	Validate settings	s upon exit				Ad <u>v</u> a	anced

Firewalls and the GigE Vision Camera Network Connection

Firewalls can block UDP packets used for image data transfer by the GigE Vision Streaming Protocol. The ports used for image data transfer by the camera and Gigabit network adapter are dynamic, and setting up the required firewall rules is an advanced operation. Firewall processing of the image data stream can also reduce performance.

A convenient getting-started configuration is to turn off the firewall for the camera's dedicated network connection, while keeping the firewall enabled for the other network connections (such as the Internet connection or your organization's local area network connection).

Windows uses network location profiles to group firewall settings for different types of connections, and turning off Windows firewall selectively for an individual network connection is not directly possible. Refer to the following online article for possible network and firewall configurations: http://www.mathworks.com/matlabcentral/answers/232356.

TU Bibliotheks Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar wien verknowedge hub

Configure Camera GigE Vision Streaming Parameters in MATLAB

Each image acquired by the camera is transferred to the computer as a data block of UDP packets, which are processed by MATLAB. For a lower CPU load during image acquisition and to prevent dropped frames, you can configure the camera GigE Vision streaming parameters (packet size and packet delay) in MATLAB.

Packet Size

- The toolbox configures the packet size automatically once the connection between MATLAB and the camera is established.
- The packet size value can also be configured manually by setting the PacketSize property of the videoinput source or gigecam objects.
- The packet size value should be set to the largest size that the network adapter and camera can handle (not larger than the Ethernet adapter jumbo packet size configured in step 1).

Packet Delay

- To prevent dropped frames, you can configure the camera to introduce a time delay between the image data packets by setting the PacketDelay property of the videoinput source or gigecam objects.
- Optimum values for the packet delay depend on the packet size, acquired frame size (image height and width), specified pixel format (for example 'Mono8'), camera frame rate, and other camera-specific settings.

The following online article provides details about determining the recommended packet delay value, and setting the packet size and packet delay values in MATLAB: http://www.mathworks.com/matlabcentral/answers/91834.

Basler pylon

// Create an instant camera object with the firs Camera_t camera(CT1Factory::Gettostance().Creat

// Register an image event handler that accesses camera.RegisterImageEventHandler(_new CSampleIma Ownership_TakeOwnership);

// Open the damer camera.Open();

APPLICATION NOTE

How to use the Basler pylon GenTL Producers for Basler GigE and USB 3.0 Cameras with MathWorks MATLAB

Document Number: AW001343 Version: 03 Language: 000 (English) Release Date: 6 November 2017



Contacting Basler Support Worldwide

Europe, Middle East, Africa

Basler AG An der Strusbek 60–62 22926 Ahrensburg Germany

Tel. +49 4102 463 515 Fax +49 4102 463 599

support.europe@baslerweb.com

The Americas

Basler, Inc. 855 Springdale Drive, Suite 203 Exton, PA 19341 USA

Tel. +1 610 280 0171 Fax +1 610 280 7608

support.usa@baslerweb.com

Asia-Pacific

Basler Asia Pte. Ltd. 35 Marsiling Industrial Estate Road 3 #05–06 Singapore 739257

Tel. +65 6367 1355 Fax +65 6367 1255

support.asia@baslerweb.com

www.baslerweb.com

All material in this publication is subject to change without notice and is copyright Basler AG.

Table of Contents

1	Introd	duction		2
2	Requ	irements		2
3	Instal	llation		3
	3.1	Installing	g the Basler pylon GigE and USB GenTL Producers	3
	3.2	Installing	g and Configuring MATLAB	5
4	Opera	ating a Ba	sler USB3 Vision Camera with MATLAB	9
5	Opera	ating a Ba	sler GigE Camera with MATLAB	11
6	Trout	oleshootin	g	13
	6.1	Trouble	shooting USB 3.0 Cameras	13
		6.1.1	Troubleshooting Using MATLAB	13
		6.1.2	Troubleshooting Using the Basler pylon Camera Software Suite	15
	6.2	Trouble	shooting GigE Vision Cameras	16
		6.2.1	Troubleshooting Using MATLAB	16
		6.2.2	Troubleshooting Using the Basler pylon Camera Software Suite	18

1 Introduction

The Basler pylon GigE and USB GenTL producers enable you to operate Basler GigE Vision and USB3 Vision cameras with MATLAB (www.mathworks.com) or any software libraries that support the GenICam GenTL standard (http://www.emva.org) and implement their own GenTL consumer.

This document describes how to configure MATLAB so that Basler GigE and USB 3.0 cameras are detected in MATLAB. Once the cameras are displayed in MATLAB, you can configure them as desired and acquire images.

The Basler pylon GigE and U3V GenTL producers are included in the pylon Camera Software Suite 5.0.11 or higher which is available for Windows, Linux x86, and Linux ARM. For more information about the pylon Camera Software Suite and to download the latest version go to www.baslerweb.com.

2 Requirements

The procedures described in this document assume that the following software and hardware is present:

- MATLAB R2017a (9.2.0.556344) 64-bit with the Image Acquisition Toolbox version 5.2 (R2017a)
- Image Acquisition Toolbox Support Package for GenICam Interface version 17.2.0.0, which enables you to acquire video and images from GenTL-compliant cameras
- Image Acquisition Toolbox Support Package for GigE Vision Hardware version 17.2.0.0, which is required for advanced IP address configuration and troubleshooting of GigE cameras
- Basler pylon Camera Software Suite 5.0.11 including the pylon GigE and USB GenTL 64-bit producers
- Basler GigE and/or Basler USB 3.0 cameras
- GigE network card and/or USB 3.0 host controller card recommended by Basler
- GigE and or USB 3.0 cables recommended by Basler



The Basler pylon GenTL producers are compliant with GenTL version 1.5.

For more information about recommended accessories for Basler GigE and Basler USB 3.0 cameras, visit to the Basler website (www.baslerweb.com) or contact your local Basler Customer Service team.

3 Installation

3.1 Installing the Basler pylon GigE and USB GenTL Producers

To install the Basler pylon GigE and USB GenTL producers:

- 1. Run Basler pylon 5.0.11.xxx.exe.
- 2. On the **Profiles** page, choose either the **Camera User** or the **Developer** profile. If you intend to develop applications based on the pylon Camera Software Suite APIs, choose the **Developer** profile.

Δ		×
pylon 5		BASLER?
	Profiles	
	Choose the profile that best describe	es your tasks.
A	Camera User Developer Gus	torn
	You can change the profile any time again.	by running the setup
	Back Next	Cancel
Copyrig	nt © 2007-2017 Basler All Rights Reserved www	v.baslerweb.com

- 3. Click Next.
- 4. On the **Interfaces** page, select the **Interfaces** for which you want to install the necessary drivers, runtime environment, and GenTL producers, e.g., USB and GigE.

	BASLER
Interfaces	
 Select how your camera(s) is/a computer.	re connected to the
US8 Gigf	FreWire Corpera Link
You can change the interface setup again.	s any time by running the
Back Next	Cancel

- 5. Click Next.
- 6. Click Install to install the selected components now.

D. L. L. L.		BASLER?
Ready to Install		
 Profile selected:	✓ Camera User	
Interface(s) selected	✓ USB ✓ GigE	
During installation and network connections unavailable. Close all connection before pro	d uninstallation of py will temporarily beco applications needing occeeding with the ins	lon, me a network tallation!

7. After the installation has completed, log off from your computer and then log in again. This is necessary for the changed system environment variables to take effect.

3.2 Installing and Configuring MATLAB

```
To install and configure MATLAB:
```

1. In MATLAB, go to the **APPS** tab and make sure that the **Image Acquisition** app is installed.

HOM	B.	PLOTS	APP	S
Get More Apps	Install App	Package App	Image Acquisition	Color Thresholder

If the Image Acquisition app has not been installed yet, install it by running the Support Package Installer in one of the following ways:

• On the HOME tab, click Add-Ons > Get Hardware Support Packages.



In the MATLAB Command Window, enter:

supportPackageInstaller



a. In the Add-On Explorer, remove the **Clear Filter** field and enter '**Image Acquisition Toolbox**'.



- b. Select the Image Acquisition Toolbox and install it.
- Check if the MATLAB GenTL consumer is available. In the MATLAB Command Window, enter:

imaqhwinfo

```
Command Window
Sponsored Third Party Support License -- for use
MathWorks software under terms specified in your
>> imaqhwinfo
ans =
struct with fields:
InstalledAdaptors: {'gentl' 'gige'}
MATLABVersion: '9.2 (R2017a)'
ToolboxName: 'Image Acquisition Toolbox'
ToolboxVersion: '5.2 (R2017a)'
```

If 'gentl' is not listed under **InstalledAdaptors**, install the MATLAB GenTL Consumer in one of the following ways:

- On the HOME tab, click Add-Ons > Get Hardware Support Packages.
- In the MATLAB Command Window, enter:

supportPackageInstaller

3. In the Add-On Explorer enter 'gentl' in the search field.



4. Select the Image Acquisition Toolbox Support Package for GenICam Interface.

a m RZ01/6 now a	vailable	Cisar Fithers × genti	Q
Filter by Source	1 I RESULT		
Filter by Category			
Applications Image Processing and Computer Vibion Herdware interfacing and IoT	1	Image Acquisition Toolbox Support Package for GeniCam Interface version 1728 0 by MatWorks Image Acquisition Toolbox Team	 Bowmloods ID Updated 14 Jun 2017
Filter by Type		Acquire video and images from	
Hardware Support Packages	T	GenTL compliant cameras.	
Filter by Hardware Type		Hardware Support	

5. Click Install to install the Support Package for GenICam Interface.



A window with licensing information will be displayed.

6. Accept the MathWorks Auxiliary Software License Agreement and the Genicam License Agreement.

7. Click Next to install the Support Package for GenICam Interface.



8. When the installation is complete, click Finish.

📣 Add-On Explorer	– 🗆 X
	Manage Add-Ons
4 📸 R20	×
Installation Complete	D Ratings 83 Downloads 10 Updated 14 Jun 2017
Qverview	Manage
Image Acquisit from GenlCam	ges and video
This support p:	Finish
Comments	
Data dita dista di secondaria di secondaria	
4 Operating a Basler USB3 Vision Camera with MATLAB

This section explains how to access a Basler ace USB 3.0 camera in MATLAB and how to configure the camera's features.

To access a Basler ace USB 3.0 camera in MATLAB:

- 1. Connect your Basler ace USB 3.0 camera to a port of a USB 3.0 host controller card recommended by Basler.
- 2. In MATLAB, click **APPS** > Image Acquisition.

The **Image Acquisition Tool** opens in a new window. In this example, a Basler ace acA4024-29uc camera is used. It is listed in the **Hardware Browser** pane.

📣 Image Acquisition Tool	-				
<u>F</u> ile <u>T</u> ools <u>D</u> esktop <u>W</u> indow <u>H</u> elp					
Hardware Browser 🗠 🗆 👼 🗙	Preview - Select a device format.	s ⊡ +			
Mage Acquisition Toolbox					
BGR8 BayerRG12	Select a device format.				
BayerRG8 BayerRG8 RGB8 RGB8	Preview Acquire				
_	Start Preview Stop Pre Start Acquisition Trigger Stop Acquisition	Export Data			
Information ± □ ₹ ×	Acquisition Parameters	± □ ≉ ×			
Image Acquisition Toolbox					
Toolbox version: 5.2 (R2017a)					
MATLAB version: 9.2 (R2017a)					
Installed adaptors: gentl, gige, winvide					
~					
< >>					

- 3. When you select the camera in the **Hardware Browser** pane, the available pixel formats will be listed in the **Acquisition Parameters** pane.
- 4. Select the desired pixel format and click **OK**.

The camera has been opened and is ready for use now.

📣 Image Acquisition Tool	-	D X			
<u>File Tools D</u> esktop <u>W</u> indow <u>H</u> elp					
Hardware Browser	Preview - Select a device format.	->- □ ₹	•		
Image Acquisition Toolbox					
BGR8	Colort a device format				
BayerRG12	Select a device format.				
Mono8 (default) RGB8	Preview Acquire Start Preview Stop Pre Start Acquisition Trigger Stop Acquisition	Export Data	-		
Information ± □ ? ×	Acquisition Parameters	× s 🗆 ż	:		
acA4024-29uc (22223214) (gentl-1)	Select an available format: Mono8 (default)				
Device: acA4024-29uc (22223214)	BGR8 BruerPC12				
Adaptor: gentl	BayerRG8				
Device ID: 1	RGB8				
< >					

5. In **Acquisition Parameters** pane, go to the **Device Properties** tab to access all currently supported camera features.

Hardware Browser	* 0 / X	Preview - acA4024-2luc (22223214) (gentl-1): BayerRG8			-	
Image Acquisition Toolbox Image Acquisition Image Acquisition		Gick Start Preview or Start Acquisition to begin.				
		Preview Start Preview Stop Provine	eady to start preview or as Acquire Start Acquisition	Tingger Stop Acquisite	in Export Dita	
information	1 0 * ×	Acquisition Parameters			: 0	
BoyerRGB Device Resolution: Selected source:	ecA4024-29uc (4024x3036 Stream0	General Device Properties Logging Triggerin Selected source Dever0	Region of Interest		Reset to defaults	Î
Number of frames to acquire: 1 Logging mode: memory Trigger type: immediate Adaptor/Driver Description: Image Acquisit Adaptor/Driver Version: 1.0.LIBRARY_BU	Acquisition Burst Frame Count: 1 Acquisition Frame Rate: 28.99 Acquisition Frame Rate Enable: False AcquisitionStatusSelector AcquisitionStatusSelector Fra	8 V neTriggerWait V	1	255	Ī	
		Auto Exposure Time Lower Limit: 50 Auto Exposure Time Upper Limit: 1000	0			

6. In the **Preview** pane, click **Start Preview** in order to get a live image from the camera.



5 Operating a Basler GigE Camera with MATLAB

This section explains how to access a Basler ace GigE Vision camera in MATLAB and how to configure the camera's features.

To access a Basler ace GigE camera in MATLAB:

- 1. Connect your Basler ace GigE camera to a GigE network card recommended by Basler.
- 2. In MATLAB, click **APPS > Image Acquisition**.

The **Image Acquisition Tool** opens in a new window. In this example, a Basler ace acA2500-20gm camera is used. It is listed in the **Hardware Browser** pane.

 When you select the camera in the Hardware Browser pane, the available pixel formats will be listed in the Acquisition Parameters pane. Select the desired pixel format and click OK.

The camera has been opened and is ready for use now.

📣 Image Acquisition Tool	- 🗆 X
<u>File Tools D</u> esktop <u>W</u> indow <u>H</u> elp	
Hardware Browser 🕨 🖛 🛪 🗙	Preview - Select a device format.
Image Acquisition Toolbox acA2500-200m (21694230) (genti-1) Mono10 Mono8 (default)	Select a device format. Preview Stop Preview Stop Preview Stop Preview Stop Acquire Stop Acquisition Trigger Stop Acquisition Front Data
Information ± z × × acA2500-20gm (21694230) (gentl-1) Device: acA2500-20gm (21694230) Adaptor: gentl Device: acA2500-20gm (21694230) Device: acA2500-20gm (21694230) Adaptor: gentl Device: 1	Acquisition Parameters ± 2 × × Select an available format: Mono§ (default) Mono10 OK

4. In the **Acquisition Parameters > Device Properties** tab you can access all currently supported camera features:

L'activité bionètés . C . V	Preniew - acA2500-20gm (21694220) (gentl-1): Mono8 (default) + C			
Image Acquisition Toolbox acA2500-20gm (21694230) (genti-1) Mono10				
S Montol (Architel) *	Click Start Preview or Start Acquisition to begin.			
	Ready to start preview or acquisitions Preview Acquire			
	Start Preview Stop Preview Start Acquisition Trigger Stop Acquisition Export Data.			
Information ± 🗆 + 🛪	Acquisition Parameters			
	General Device Properties Logging Triggering Region of Interest			
Monoŝ	Selected source Intromit +	1		
Mono8 Device: scA2500-20g	Selected source Stream - Properties	ľ		
Mono8 Device scA2500-20g Resolution 2592x2048	Selected source Stream - Properties Reset to defaults			
Mono8 Device: scA2502-20g Resolution: 2592:2048 Selected source: 3treem0	Selected source Streams - Properties Reset to defaults			
Mono8 scA2500-20g Resolution 29922048 Selected source: Streem0 Number of frames to acquire: 1	Selected source Streams - Properties Reset to defaults Acquisition Frame Count: 1 255			
Mono8 Device: acA2500-20g Resolution: 2595x204B Selected source: 3brem0 Number of fremes to acquire: 1 Logging mode: memory	Selected source Stream C + Properties Recet to defaults Acquisition Frame Count: 1 255 Acquisition Frame Rate Abs: 50			
Mono8 Device: scA2500-200 Resolution: 2502x204 Selected source: 350wm70 Number of firmers to acquire: 1 Logging mode: memory Trigger type: immediate	Selected source Stream C - Properties Reset to defaults Acquisition Frame Count: 1 Acquisition Frame Rate Abs: 50 Acquisition Frame Rate Count: False ×			

5. In the **Preview** pane, click **Start Preview** in order to get a live image from the camera:



6 Troubleshooting

6.1 Troubleshooting USB 3.0 Cameras

6.1.1 Troubleshooting Using MATLAB

If the images you acquire are corrupt, try decreasing the bandwidth currently used by the camera in the MATLAB Image Acquisition Tool. You can do this in two different ways:

Image Acquisition Tool:

In the Image Acquisition Tool, go to **Acquisition Parameters** > **Device Properties** and make the following changes:

- Set Device Link Throughput Limit Mode to On.
- Device Link Throughput Limit: Change the value (in Bytes/s) until MATLAB stops acquiring corrupt images, e.g., 300 000 000 Bytes (i.e., 300 Mbytes).

uisition Parameters				± 81
neral Device Properties Logging Trig	gering Region of Interest			
DeviceLinkSelector				
Device Link Selector:	0 丈			
Device Link Current Throughput:	359,999,647 🜲			
Device Link Speed:	500,000,000 🜩			
Device Link Throughput Limit:	30000000	524200	410420400	
Device Link Throughput Limit Mode:	On 🗸	524200	415450400	
Device Manufacturer Info:	none			
Device Model Name:	acA4024-29uc			
Device SFNC Version Major:	2 🌲			
Device SFNC Version Minor:	1 🜩			
Device SFNC Version Sub Minor:	0 🔹			
Device Scan Type:	Areascan 🖂			
Device Serial Number:	22223214			
DeviceTemperatureSelector				
Device Temperature Selector:	Coreboard $ \smallsetminus $			
Device Temperature:	62			
Device User ID:				
Device Vendor Name:	Basler			
Device Version:	107405-01			
Digital Shifts	0			

Command Window:

```
In MATLAB, click HOME > Command Window and enter the following:
vid= videoinput('gentl', 1, 'BayerRG8')
src= getselectedsource(vid);
src.DeviceLinkThroughputLimitMode = 'On'
src.DeviceLinkThroughputLimit = 300000000
```

Basler Application Note

```
Sponsored Third Party Support License -- for use only to support products interfaced to
   MathWorks software under terms specified in your company's restricted use license agreement.
>> vid= videoinput('gentl', 1,'BayerRG8')
Summary of Video Input Object Using 'acA4024-29uc (22223214)'.
  Acquisition Source(s): Stream0 is available.
  Acquisition Parameters: 'Stream0' is the current selected source.
                           10 frames per trigger using the selected source.
                           'BayerRG8' video data to be logged upon START.
                           Grabbing first of every 1 frame(s).
                           Log data to 'memory' on trigger.
     Trigger Parameters: 1 'immediate' trigger(s) on START.
                  Status: Waiting for START.
                          0 frames acquired since starting.
                          0 frames available for GETDATA.
>> src= getselectedsource(vid);
>> src.DeviceLinkThroughputLimitMode = 'On'
```

If decreasing the value of the **Device Link Throughput Limit** parameter doesn't improve the image acquisition issue, you may need further debugging with the Basler pylon Viewer, which is part of the Basler pylon Camera Software Suite.

For further MATLAB troubleshooting information, you have the following options:

- Image Acquisition Toolbox User's Guide:
 - In MATLAB, click APPS > Image Acquisition > Help > Image Acquisition Toolbox User's Guide.



2. Scroll down and select Troubleshooting.

Basler Application Note

🕻 Help = 📫 🎃 🛫 🖓 🚺 Getting Started with th	re Image Acquisition Tool 🛛 🗮	+	- 0 #01	×
Documentation		Search Help		Q.
CONTENTS Close Close Coursectation Home Image Acquisition Toothox Gatting Started with Image Acquisition Toothox Image Data Acquisition	Hardware Configura Tool Menus [®] section	tion menus. For more informal of the Help on the Hardware E	tion about these commanda, see the "Image Acquisi prowser in the Desktop Help pane in the tool. Was this topic helpful? Yes	tion No
Acquisition Using the Image Acquisition Tool Image Acquisition Toolbox Image Data Acquisition Getting Started with the Image Acquisition Tool	Image Acquisition Toolox Documentation Examples Troubishooting Functions and Other Reference Reference PDF Documentation	Other Documentation MATLAS Imagis Processing Toerica Computer Vision System Toerica Data Acquisition Toelbox Simuliak Documentation Hensis	Support MATLAB Answers Installation Help Bug Reports Product Requirements Software Downloads	*

3. Here, you will find troubleshooting tips related to the GenICam GenTL Hardware:

	- L		
oting in Im	nge Acquisition Toolbex 🛛 🗧 🔲 🖯		
	Search Help		
Close			
	GigE Vision Hardware Try these tips if you have problems using the toolbox with GigE Vision image acquisition devices. Information is provided for all platforms.		
0			
	GenICam GenIL Hardware Try these tips if you have problems using the toolbox with GenICam TM GenTL image acquisition devices.		
	Windows Video Hardware Try these tips if you have problems using the teolbox with image acquisition devices that provide Video for. Windows or DirectX [®] drivers		
	Linux Video Hardware. Try these tips if you have problems using the toolbox with Linux [®] video devices.		
	Linux DCAM IEEE 1394 Hardware Try these tips if you have problems using the toolbox with a supported Linux DCAM IEEE [®] 1394 hardware acquisition device.		
	Macintosh Video Hardware Try these tips if you have problems using the toolbox with a supported Macintosh video acquisition device.		
	Macintosh DCAM IEEE 1394 Hardware Try these tips if you have problems using the toolbox with a supported Macintesh DCAM IEEE 1394 hardware acquisition device.		
	Video Preview Window Troubleshooting Try these tips if you have problems using the Preview window.		
	Close		

Contact the MATLAB technical support team.

6.1.2 Troubleshooting Using the Basler pylon Camera Software Suite

The Basler pylon Camera Software Suite offers the following tools: pylon Viewer, pylon USB Bandwidth Manager and pylon USB Configurator.

If you need help using these tools, contact your local Basler Customer Service Team (www.baslerweb.com).

6.2 Troubleshooting GigE Vision Cameras

6.2.1 Troubleshooting Using MATLAB

If your GigE Vision camera is not shown in the MATLAB Image Acquisition Toolbox or the images you acquire are corrupt, you have to install the **Image Acquisition Toolbox Support Package for GigE Vision Hardware**. To do so, follow these steps:

- 1. On the HOME tab, click Add-Ons > Get Hardware Support Packages.
- 2. In the Add-On Explorer enter 'gige' in the search field:



 Select the Image Acquisition Toolbox Support Package for GigE Vision Hardware and click Install to install it.



4. When the installation process has completed, click Finish.

📣 Add-On Explorer	- D X
	Manage Add-One
🔺 🏽 R2017	×
Installation Com	te 2 Watnes
7/G= Con Example	BB Count order 🗐 Einderson 14 Jun 2017
1 5 1 0	e Manage
Overview	
Image Acquisition T- Vision compliant car	and video from GigE
This support packag	
	Finish
Comments and	

 In MATLAB, click HOME > Command Window enter the following to optimize the Packet Size and the Packet Delay parameters.

```
vid= videoinput('gige', 1, 'Mono8')
src= getselectedsource(vid);
src.PacketSize = 9014;
src.PacketDelay= 5000;
```

```
Command Window

Sponsored Third Party Support License -- for use only to support products interfaced to

MathWorks software under terms specified in your company's restricted use license agreement.

>> vid = videoinput('gige', 1, 'Mono8');

>> src = getselectedsource(vid);

>> src.PacketSize = 9014;

>> src.PacketDelay= 50000;

ft >>
```



When using jumbo frames for the **Packet Size** camera parameter, i.e., values above 1500 Byte, make sure that your network adapter supports that value. Otherwise, image acquisition may fail.

Increasing the **Packet Delay** camera parameter, may reduce the camera's resulting maximum acquisition frame rate.



For more information about troubleshooting GigE Vision cameras in MATLAB, refer to the following documents:

- MATLAB GigE Vision Quick Start Configuration Guide: <u>https://www.mathworks.com/matlabcentral/answers/uploaded_files/41167/GigEVisionQuickStart.pdf</u>
- MATLAB FAQ: <u>https://de.mathworks.com/matlabcentral/answers/91834-how-do-i-calculate-the-packet-delay-for-a-gige-vision-camera-to-prevent-dropped-frames</u>

6.2.2 Troubleshooting Using the Basler pylon Camera Software Suite

The Basler pylon Camera Software Suite offers the following tools: pylon Viewer, pylon GigE Bandwidth Manager and pylon IP Configurator.

If you need help using these tools, contact your local Basler Customer Service Team (www.baslerweb.com).

Revision History

Document Number	Date	Changes
AW00134301000	4 Feb 2015	Initial release version of this document.
AW00134302000	6 Nov 2017	Updated the document to reflect the use of MATLAB R2017a. Updated the document to reflect the use of pylon 5.0.11. Added Chapter 5 to reflect the operation of GigE Vision cameras with MATLAB. Added Chapter 6.2 to reflect the troubleshooting of GigE Vision cameras.
AW00134303000	6 Nov 2017	Corrected a typo.

TU Bibliothek Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar ^{WIEN} ^{vour knowledge hub} The approved original version of this thesis is available in print at TU Wien Bibliothek.

Anhang 5: Bedienungsanleitung

Schritte zum durchführen einer automatisierten Messung:

- 1. Im "Hardware Setup" Tab verbinden und testen der Hardware
- 2. Referenzieren des Linearschlittens
- 3. Im "Automated Measurement" Tab einstellen des Messgitters
- 4. Nach starten der automatisierten Messung dem Anweisungen des Programms folgen
- 5. Messdaten abspeichern (Messdaten werden automatisch im "Configure Evaluation" Tab geladen)
- 6. Einstellen der Auswertungsparameter
- 7. Das Zentrum des Strahls mit dem Guess Beamarea Button markieren
- 8. Starten der Auswertung
- 9. Sollte die Auswertung abbrechen ist meistens die Hintergrundkorrektur der Grund
- 10. Ist die Auswertung abgeschlossen werden die ausgewerteten Messdaten automatisch in den "Evaluation Result" Tab geladen

Schritte zum durchführen einer Livemessung:

- 1. Im "Hardware Setup" Tab verbinden und testen der Hardware
- 2. Referenzieren des Linearschlittens
- 3. Über die Kontrollfelder im "Hardware Setup" Tab Linearschlitten an gewünschte Position verfahren
- 4. Im "Live Measurement" Tab ein Probebild des Strahlprofils machen (Trigger Image Button)
- 5. Das Zentrum des Strahls mit dem Guess Beamarea Button markieren
- 6. Einstellen der Auswertungsparameter
- 7. Hintergrund messen mit dem Measure Background Button (dafür muss der Laserstrahl abgeblockt werden)
- 8. Starten der Livemessung über den Toggle Button Start/Stop Live Measurement

Literatur

- [1] Joachim Grehn, Gerd Harbeck, Peter Wessels. *PSSC Physik.* 2. Aufl. Friedr. Vieweg + Sohn . Braunschweig, 1974.
- [2] Jürgen Eichler, Hans Joachim Eichler. Laser, Bauformen, Strahlführung, Anwendungen. 3. Aufl. Springer-Verlag, 1998.
- [3] Chunlei Guo, Subhash Chandra Singh. Handbook of Laser Technology and Applications.2. Aufl. CRC Press, 2021.
- [4] Peter Adams. "Die Strahlqualität von Lasern". In: Laser Technik Journal 1.2 (Okt. 2004), S. 63–66.
- [5] Norman Hodgson, Horst Weber. Laser Resonators and Beam Propagation. 2. Aufl. Springer-Verlag, 2005.
- [6] Claude Rullière. Femtosecond Laser Pulses. 2. Aufl. Springer Science+Business Media, Inc, 2003.
- [7] Pulse-Compression. 2022. URL: http://www.swampoptics.com/pulse-compression.html. zugegriffen am: 23.02.2022.
- [8] Laser und Laseranlagen Pr
 üfverfahren f
 ür Laserstrahlabmessungen, Divergenzwinkel und Beugungsmaßzahlen – Teil 1: Stigmatische und einfach astigmatische Strahlen (ISO 11146-1:2005); Deutsche Fassung EN ISO 11146-1:2005. Norm. Apr. 2005.
- [9] Laser und Laseranlagen Pr
 üfverfahren f
 ür Laserstrahlabmessungen, Divergenzwinkel und Beugungsmaßzahlen - Teil 3: Intrinsische und geometrische Laserstrahlklassifizierung und propagation sowie Details der Pr
 üfverfahren. Norm. Feb. 2004.
- [10] Datenblatt Thorlabs PS908. 2021. URL: https://www.thorlabs.com/newgrouppage9.cfm? objectgroup_id=142. zugegriffen am: 20.10.2021.
- [11] Unbeschichtetes rechtwinkliges Prisma Thorlabs PS908. 2021. URL: https://www.thorlabs.com/images/GuideImages/3765_PS908_SG.jpg. zugegriffen am: 12.10.2021.
- [12] Übertragungskurve von unbeschichteten BK7. 2021. URL: https://www.thorlabs.com/ images/TabImages/Uncoated_N-BK7_Transmission_780.gif.zugegriffen am: 12.10.2021.
- [13] Datenblatt ND-Filter (Thorlabs NE40B-B, NE50B-B). 2021. URL: https://www.thorlabs. com/newgrouppage9.cfm?objectgroup_id=6273. zugegriffen am: 12.10.2021.
- [14] Bild ND-Filter (Thorlabs NE40B-B, NE50B-B). 2021. URL: https://www.thorlabs.com/ images/large/24280-lrg.jpg. zugegriffen am: 12.10.2021.
- [15] Übertragungskurven für OD 2.0 OD 6.0. 2021. URL: https://www.thorlabs.com/images/ TabImages/B_ND_Filters_Transmission_26_G1-350.gif. zugegriffen am: 12.10.2021.
- [16] Datenblatt Schrittmotor Sanyo Denki 103H5205-5240. 2021. URL: https://docs.rs-online. com/9f7b/0900766b813b30f6.pdf. zugegriffen am: 20.10.2021.
- Bild Schrittmotor Sanyo Denki 103H5205-5240. 2021. URL: https://docs.rs-online.com/ 9f7b/0900766b813b30f6.pdf. zugegriffen am: 20.10.2021.
- [18] Momentenverlauf Sanyo Denki 103H5205-5240. 2021. URL: https://docs.rs-online.com/ 9f7b/0900766b813b30f6.pdf. zugegriffen am: 20.10.2021.

Literatur

- [19] Arduino UNO Rev3. 2021. URL: https://store.arduino.cc/products/arduino-unorev3?selectedStore=eu. zugegriffen am: 20.10.2021.
- [20] Bild Arduino UNO Rev3. 2021. URL: https://cdn.shopify.com/s/files/1/0506/ 1689/3647/products/A000079_03.front_710x467.jpg?v=1615307638. zugegriffen am: 20.10.2021.
- [21] Arduino UNO Rev3. 2021. URL: https://store.arduino.cc/products/arduino-motor-shield-rev3?selectedStore=eu.zugegriffen am: 20.10.2021.
- [22] Bild Arduino UNO Rev3. 2021. URL: https://cdn.shopify.com/s/files/1/0506/ 1689/3647/products/A000079_03.front_710x467.jpg?v=1615307638. zugegriffen am: 20.10.2021.
- [23] Datenblatt Basler ace GigE acA1300-60gmNIR. 2021. URL: https://docs.baslerweb.com/ aca1300-60gmnir. zugegriffen am: 20.10.2021.
- [24] Bild Basler ace GigE acA1300-60gmNIR. 2021. URL: https://docs.baslerweb.com/ images/image-ace-gige-mono.jpg. zugegriffen am: 12.10.2021.