DIPLOMA THESIS

ScenARy: Scene Awareness in HMD-assisted Multi-Machine Scenarios

by

Andreas Steiner, BSc

[redacted]

carried out for the purpose of obtaining the degree Master of Science
under the supervision of

Univ.-Prof. Dr.-Ing. Sebastian Schlund
Dipl.-Ing. Kostolani David, BSc

E 330 Institute of Management Science
Human-Machine-Interaction

submitted at

Technische Universität Wien
Getreidemarkt 9
1060 Wien, Österreich

Wien, 17th of January, 2023

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.

"Our intelligence is what makes us human, and AI is an extension of that quality."

– Yann LeCun

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.

Ich habe zur Kenntnis genommen, dass ich zur Drucklegung meiner Arbeit unter der Bezeichnung

ScenARy: Scene Awareness in HMD-assisted Multi-Machine Scenarios

nur mit Bewilligung der Prüfungskommission berechtigt bin.

Ich erkläre weiters Eides statt, dass ich meine Diplomarbeit nach den anerkannten Grundsätzen für wissenschaftliche Abhandlungen selbstständig ausgeführt habe und alle verwendeten Hilfsmittel, insbesondere die zugrunde gelegte Literatur, genannt habe. Weiters erkläre ich, dass ich dieses Diplomarbeitsthema bisher weder im In- noch Ausland (einer Beurteilerin/einem Beurteiler zur Begutachtung) in irgendeiner Form als Prüfungsarbeit vorgelegt habe und dass diese Arbeit mit der vom Begutachter beurteilten Arbeit übereinstimmt.

Wien, 17th of January, 2023

Vorname Nachname

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.
TU Bibliothek Your knowledge hub WIEN

Abstract

Global competition within the manufacturing sector nowadays is forcing more and more companies to implement modern, innovative technologies into their factories. Technologies such as Augmentet Reality (AR), are becoming more and more present since they can support human workers in their daily tasks. As an employee's workflow has changed in recent years towards shorter work cycles and more flexible job roles, their performance requirements have also increased significantly. Employees are now responsible for a larger work area and forced to work at several workstations, sometimes simultaneously. Due to the changing work environment, this so-called multi-machine operation cannot yet be accomplished in an ideal way with the help of AR, as the current technologies are limited to a single workstation. Better support would be ensured if it was possible to automatically recognize which machine is in usage. Used in multi-machine operations, in this case AR could provide workstation-specific instructions. Furthermore, Deep Learning (DL) enables context sensitivity - automated recognition - for AR-equipped devices. An assistance system based on AR and DL would therefore be able to support human decision-making in everyday work. This thesis is a proof of concept on the way to a fully developed assistance system to ease the workload of employees in multi-machine operations by giving the system the ability of context-awareness. The main contributions of this thesis are twofold - the creation of a new dataset called "*Assembly Factory Workplaces*", consisting of 4695 images, 6 workplace categories and 4 special categories for transitions – and the development of a new neural network, based on a modified *ResNet-50* architecture, which was first pre-trained on the *Places365* dataset and finally tuned with the self-developed dataset. This neural network shows an overall validation accuracy of 77% on the *Assembly Factory Workplaces* dataset. The performed use case, conducted at the TU Wien pilot factory, demonstrates that it is possible to install a neural network on a head mounted display (HMD) device (*i.e., Microsoft HoloLens 1*) and enable context awareness for AR devices. Furthermore, the observations within the test run have shown that the classification of main and special categories works both accurately and sufficiently fast under test conditions. This thesis provides a proof of concept that can be extended in the future by increasing the *Assembly Factory Workplaces* dataset to be more diverse, deploying the system on other augmented reality headsets or smartglasses, or try other neural network architectures.

Keywords— Industry 4.0, Assembly Factories, Assistance Systems, Multi-Machine Operations, Cognitive Stress, Context Awareness, Artificial Intelligence, Deep Learning, Residual Networks, Head-Mounted Displays, Microsoft Hololens

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.

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.

DE-Version

Der weltweite Wettbewerb in der Fertigungsindustrie zwingt heutzutage immer mehr Unternehmen dazu, moderne, innovative Technologien in ihre Fabriken zu implementieren. Technologien wie Augmentet Reality (AR) beispielsweise werden immer präsenter, weil sie die menschliche Arbeitskraft bei ihren täglichen Aufgaben unterstützen können. Da sich der Arbeitsablauf eines Mitarbeiters in den letzten Jahren in Richtung kürzerer Arbeitszyklen und flexibleren Tätigkeitsbereichen verändert hat, sind auch deren Leistungsanforderungen deutlich gestiegen. Mitarbeiter*innen sind nun für einen größeren Arbeitsbereich verantwortlich und gezwungen, zum Teil gleichzeitig, an mehreren Arbeitsplätzen tätig zu sein. Dieser sogenannte Mehrmaschinenbetrieb kann aufgrund der wechselnden Arbeitsumgebung mithilfe von AR noch nicht in idealer Weise unterstützt werden, da die derzeitigen Technologien noch nicht vollständig ausgereift sind. Eine bessere Unterstützung wäre dann gewährleistet, wenn automatisch erkannt würde, um welche Maschine es sich handelt. Eingesetzt in Mehrmaschinenbetrieben könnte AR in diesem Fall arbeitsplatzspezifische Anweisungen bereitstellen. Deep Learning (DL) ermöglicht dabei eine Kontextsensitivität – also ein automatisches Erkennen – für mit AR ausgestattete Geräte. Ein auf AR und DL basierendes Assistenzssystem wäre daher in der Lage, die menschliche Entscheidungsfindung und den Arbeitsalltag zu erleichtern. Diese Arbeit bietet einen ersten Entwurf am Weg zu einem vollständig entwickelten Assistenzsystem zur Entlastung von Mitarbeiter*innen in Mehrmaschinenbetrieben durch die Fähigkeit des Systems zur Kontextwahrnehmung. Die beiden Hauptbestandteile dieser Arbeit sind zum Einen die Erstellung eines neuen Datensatz mit der Bezeichnung „*Assembly Factory Workplaces* “ – bestehend aus 4695 Bilder, 6 Arbeitsplatzkategorien und 4 speziellen Kategorien für Übergänge – und zum Anderen die Entwicklung eines neuen neuronalen Netzwerkes, basierend auf einer modifizierten *ResNet-50* Architektur, welches zunächst auf den *Places365* Datensatz vortrainiert und schließlich mit dem selbstentwickelten Datensatz abgestimmt wurde. Dieses final trainierte neuronale Netzwerk weist eine Validität von 77% in Bezug auf den selbst erstellten Datensatz auf. Der im Zuge dieser Arbeit in der TU-Wien Pilot-Fabrik durchgeführte Testlauf konnte zeigen, dass es möglich ist, ein neuronales Netzwerk auf einem HMD-Gerät, beispielsweise der *Microsoft HoloLens 1*, zu installieren, um dadurch kontextsensible AR-Geräte zu schaffen. Des Weiteren konnte im Zuge des Testlaufes gezeigt werden, dass die Klassifizierung der Haupt- und Spezialkategorien unter Testbedingungen sowohl genau als auch ausreichend schnell funktioniert. Auf dem Konzept dieser Arbeit könnte in Zukunft aufgebaut werden, indem der erstellte Datensatz vergrößert, das System auf anderen Augmentet-Reality-Headsets oder Smartglasses eingesetzt wird oder neue neuronale Netzwerkstrukturen ausprobiert werden.

Acknowledgements

At this point, I would like to express my gratefulness to those who supported me during the past years both on and off my academic path and helped me overcome the many hurdles during my studies.

First of all, I am extremely grateful to my supervisor **Univ.-Prof. Dr.-Ing. Sebastian Schlund**, who gave me invaluable insight into the topic of assistance systems through his numerous lectures and made it possible for me to be part of such a great research group.

In line, I would also like to express my deepest gratitude to **Dipl.-Ing. David Kostolani, BSc**, for providing me with encouragement and patience throughout the duration of this thesis. He never wavered to help me out and by giving insightful and practical suggestions, the quality of the thesis improved a lot. Through his supervision and guidance, I have also learnt a considerable amount about the topic of deep learning. His passion and support made him an excellent instructor that I am glad to work with.

Apart from that, I would like to extend my sincere thanks to the innumerable members of the Institute of Management Science, who provided a friendly and supportive working environment. Everyone has welcomed me with open arms and I have never missed the desired support.

Finally, I would like to express my deepest gratitude to my family and girlfriend, whose help cannot be overestimated, which played a decisive role during my studies and who accompanied me in all aspects of my studies and life in general.

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.

Contents

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.

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.

List of Symbols & Abbreviations

$L(\boldsymbol{\theta})$	Loss function
α	Learning rate
β	Momentum
$\boldsymbol{W}$	Weights
$\boldsymbol{\theta}$	Parameter
$\boldsymbol{h}$	Hidden Representation
$\boldsymbol{x}$	Network/Neurone Input
n	Number of error parameters
v	Velocity
y	Network/Neurone Output
AE	Auto Encoders
AI	Artificial Intelligence
AN	Artificial Neuron
ANN	Artificial Neural Networks
AR	Augmented Reality
C	Number of channels
CNN	Convolutional Neural Networks
CPS	Cyber-Physical Systems
CV	Computer Vision
DL	Deap Learning
DNN	Deep Neural Network
E	Experience
FN	False Negative
FP	False Positive
FPS	Frames Per Second
GAP	Global Average Pooling
Grad-CAM	Gradient-weighted Class Activation Mapping
H	Image Height
HAR	Human Acticity Recognition

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.
TU Bibliothek Your knowledge hub WIEN

HCI	Human-Computer Interaction
HMD	Head-Mounted Display
HOI	Human-Object Interaction
IoT	Internet of Things
LSTM	Long Short-Term Memory
ML	Machine Learning
MLP	Multilayer Perceptrons
MR	Mixed Reality
MSE	Mean Square Error
NN	Neural Networks
ONNX	Open Neural Network Exchange
P	Performance measure
R-CNN	Region Based Convolutional Neural Network
ReLU	Rectified Linear Unit
RL	Reinforcement Learning
RNN	Recurrent Neural Network
RTA	Real Time Analysis
SELU	Scaled Exponential Linear Unit
SGD	Stochastic Gradient Descent
T	Tasks
TN	True Negative
TP	True Positive
URL	Uniform Resource Locator
UWP	Universal Windows Platform
W	Image Width
WinML	Windows Machine Learning
XAI	Explainable Artificial Intelligence
YOLO	You Only Look Once

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.
TU Bibliothek Your knowledge hub WIEN

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.

Chapter 1

Introduction

1.1 Motivation and Problem Statement

The rise of new manufacturing and information technologies has enabled new operations in the industrial sector. These technologies (*e.g.,* Cyber-Physical Systems, Internet of Things or Smart Data) can increase efficiency, lead to more flexibility and cost reductions, and provide autonomous decisions. Due to these significant changes in the industrial sector, the collective term "Industry 4.0" has been established. Driven by the beneficial reasons of these new technologies, companies implemented new concepts in their factories [1]. Despite the positive aspects mentioned above, these concepts also lead to greater diversification of manufactured products. As a result, the workflow changed to shorter cycles, flexible work tasks, and increased task monitoring, leading to higher performance requirements and more challenging jobs for employees. Consequently, it is desired to support human workers by forming more manageable working environments. Therefore, the performance imbalance between human workers can be overcome by developing ergonomic and human-centred workplaces and assistance systems. These tools or approaches are required to transform complex work tasks into feasible ones [2–4].

In this way, the implementation and introduction of assistance systems target the provision of support to workers through different levels of abstraction (*e.g.,* understanding, perception, reasoning, etc.). Despite the continuous progress of automation in the industry, assistance systems still focus on the human worker. A typical example of cognitive assistance is the use of augmented reality, *e.g.,* with head mounted displays (HMDs). However, implementing assistance systems that interactively support the human worker remains a challenge [5]. These assistive technologies often provide additional information (*e.g.,* in the form of visualisation of data) to the worker. Therefore, the attention of the human worker is required in various forms, but the cognitive resources of humans are limited. Negative output performance can occur by contesting the upper or lower limits of these cognitive resources [6]. In this case, a

mutual understanding of the goals between the human worker and the machine can benefit already complicated workflows. For a seamless integration of advanced manufacturing into smart factories, the digital infrastructure must capture, generate, or spread intelligence, improving the context-awareness of these assistance systems. Therefore, context-aware systems (*i.e.,* smart augmented reality) aim to reduce the cognitive stress of the human worker [2–4].

The rise of advanced manufacturing (*i.e.,* smart factories) based on network technologies or manufacturing data allows for a scientific decision-making process. By data extraction, images of the current workplace can be extracted and fed into the decision-making process for classification [7]. This decision-making process can be built into a context-aware assistance system, consisting of hardware and software. It supports the execution of the tasks or can be adapted depending on the task's progress. Therefore, the physical environment of a human can be tracked and analysed, providing feedback to the overall process (*e.g.,* cameras can determine the working environment of human workers). However, a meaningful technique (*e.g.,* the Deep Learning approach) is required to extract the relevant information of this countless data. Deep learning (DL), a field of machine learning (ML), has received a lot of attention in commercial and scientific research. It is a group of algorithms with generally supervised learning that implement neural networks consisting of many layers that extract a hierarchy of features from images [8]. Their ability of representation learning (*i.e.,* automatically detects important features) is the key to identifying the abstract and complex image characteristics of the workplace using neural networks. This ability contradicts the classical ML approach, in which hand-crafted features are used. In this case, scene classification based on deep learning has a high precision of determination and can be the solution to increase the context-awareness of augmented reality as an assistance system [4, 5].

Multi-machine operation can be described as a manufacturing process where one human worker is operating on multiple machines. It requires higher qualifications and cognitive skills of human workers compared to a single workplace operation. This presents an ideal use case for augmented reality to assist workers and augment their cognitive capabilities. However, as of now, multi-machine operation is too complex for augmented reality to be used and assist workers. In such environments, parallel processes create conditions in which the human worker has to decide what should be done next. This parallel process thinking can be overwhelming and can cause increased pressure on the human worker. In this case, implementing a context-aware system helps the assistance systems decide where the human worker is located. Based on location detection, adaptations can be made that support the human worker. Therefore, instructions can be adapted based on the recognised scene, more effective shop floor coordination can be made, and human cognitive demands are relieved [9].

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.
TU Bibliothek Your knowledge hub WIEN

1.2 Research Question and Research Objective

The question remains on what specific effects and benefits this technology (*i.e.,* scene recognition in workplaces) and its applications have on assembly workers. To reach an acceptance of this application, essential steps towards evaluating efficiency have to be made. Therefore, the first research question is: *How can workplace recognition enhance context-awareness of assistance systems?*

In the second step, a state-of-the-art analysis is performed by evaluating suitable deep learning algorithms, publicly available datasets and applications for scene classification. Different algorithms will lead to different information outcomes. These possible outcomes are related to the first research question to identify the most promising algorithm. Therefore, the second research question states: *Is there any available dataset for workplace scene classification and what is the most promising scene classification approach?*

Lastly, on the basis of a gathered data set, the most promising deep learning algorithm should be implemented in the TU Wien pilot factory. Therefore, the developed neural network should be deployed onto a HMD device to enable context-aware AR devices within factories. Subsequently, the last research question states: *How can the developed deep learning algorithm be deployed on a HMD device and how does it behave in the TU Wien pilot factory?*

1.3 Methodology

The first research question is answered by a literature search in the field of Industry 4.0 and deep learning. Google Scholar as the search engine will be used, and relevant words from these topics are combined to narrow the search result. The entire search query can look as follows:

$$\text{[context-aware* OR adaptive] AND [workplace* OR assistance system*]} \rightarrow \text{results} \quad (1.1)$$

Furthermore, the value of workplace recognition for assistance systems is determined by evaluating its strengths and weaknesses [3].

Based on Research Question 1, Research Question 2 should be answered by a state-of-the-art literature review of the existing deep learning approaches in scene classification. Therefore, different deep learning techniques (*e.g.,* Convolutional Neural Network (CNN)) for scene recognition are assessed. The entire search query can look as follows:

$$\begin{aligned}&\text{[Deep Learning OR Machine Learning OR Convolutional Neural Network*]}\\ &\text{AND [Image Classification OR Scene recognition]} \rightarrow \text{results}\end{aligned} \quad (1.2)$$

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.
TU Bibliothek Your knowledge hub WIEN

During this literature search, use cases, applications fields, and new developments of scene classification are derived. Then each application should be evaluated and compared with special care in network architectures and hyperparameters to identify the most suitable approaches. Furthermore, the relevance of their techniques and their performance should be described.

Afterward, the most promising scene classification algorithm is developed and tested in a use case within the institutes pilot factory to answer the third research question. To extract all the important characteristics of the use case, a dataset should be generated in which the algorithm should be trained. The dataset samples should be collected by web scraping with the approach of searching for different keywords from the various workplaces. Moreover, to counteract the size irregularity of the dataset class, synthetic image generation should be utilised to compensate this. If not enough images can be received through web scraping, data augmentation or dropout (*i.e.,* regularisation techniques) might be an option to increase the model's generalisability. This collected data should be labelled into different categories and divided into training, validation, and test sets. The training and validation set is used to train the network on this specific application area by learning important features. Training the data means updating the weight parameters of the neural network to increase the accuracy of the algorithm. Since during training the algorithm remembers the images within the training set, its performance must be evaluated on unseen data (*i.e.,* validation set). However, validation by improving the algorithm performance on the dataset might not be enough. Therefore, the developed model should be tested within a use case (*i.e.,* Multi-Machine operations), where the model is deployed into a HMD device. The global workflow of the algorithm can be explained by the detection of the human environment by taking a shot through a camera. Furthermore, scene recognition classifies the workplace in which the human is located. Finally, the algorithm prediction should be validated in terms of explainability, reliability, and clarity. These are important aspects in guaranteeing the acceptance of the human worker [10].

1.4 Expected Outcome

As a result of the first research question, values in terms of weaknesses and strength are evaluated to identify how workplace recognition enhance context-awareness of assistance systems. These values can be summarised as important holistic factors that can benefit from the implementation of this kind of technology in smart factories. This outcome of the first research question acts as a decision support for the selection of a scene classification algorithm. Based on research in the literature on possible applications for scene classification, the findings (*i.e.,* deep learning approaches and publicly available datasets) are evaluated and compared with each other. However, the result of the second research question should

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.
TU Bibliothek Your knowledge hub WIEN

identify the most suitable deep learning algorithm for scene recognition of assembly factory workplaces. The third research question should be the development of a scene classification use case for the TU Wien pilot factory based on the information gained through the previous research questions. Within the case study, the HMD device should be evaluated regarding its classifcation speed and classification performance of the dataset classes. Depending on which workplace the user is located, different instructions can be given to accomplish the task. Furthermore, Explainable Artificial Intelligence (XAI) should be used to describe black-box operations transparently. Undoubtedly, the organisation and the individual will benefit from this cooperation, from information processing to improved communication and support in decision making [11, 12].

1.5 Scientific Relevance

The competitive nature of the industry forces more companies to implement high-tech methods. Through the implementation of Industry 4.0, the role of human workers will change, leading to increased complexity and skill requirements in industrial work environments. To meet the need for these requirements, assistance systems can provide a remedy. However, Lucke et al. [13] define a smart factory as "*a factory that context-aware assists people and machines in the execution of their tasks.*" Because of this definition, implementing assistance systems without context-awareness might not be sufficient to assist the human worker. In this case, the relevant literature on context-awareness is elaborated with the further treatment of industrial applications and assistance systems to show sympathy.

Subsequently, context-aware assistance can be implemented through a deep learning approach (*i.e.,* scene classification) by improving interactions between humans and their surroundings. This system should classify the working environment by scene recognition to detect wrong or faulty actions before any initial disasters appear, preventing severe damage [4]. Furthermore, a dataset is generated to represent the application area (*i.e.,* Multi Machine Operation) environment. This dataset can be important for future research on this topic and can be a source for other algorithms in this area. Additionally, a scene classification algorithm has been developed for this application area and tested for this specific use case. A template for the algorithm is generated to allow its reproducibility and the derivation of applications in the industry. In this context-aware guidance system, feedback can be given to the human worker immediately if any new work plan are recognised. Furthermore, it can help understand the complexity of human work tasks, can reduce the cognitive demant of operating on multiple workplaces and can serve as a support in case of uncertainties. Therefore, this implementation is highly interested in industry and economy by reducing the entry threshold for workplaces and enabling the ability to assign employments to low-qualified human workers.

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.
TU Bibliothek Your knowledge hub WIEN

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.

Chapter 2

Research Background and Theoretical Principles

2.1 Assistance Systems

A strong need exists to support employees who work in the future of manufacturing by making the complexity of new industrial environments more manageable. Assistance systems have the potential to address this need by supporting human work tasks or reducing human cognitive stress levels. An assistance system in production can be understood as a context-aware system which consists of hardware and software to support the human worker with the execution of a task and can adapt depending on the progress of the task. Since assistance systems can be implemented in many areas and different applications, there are various ways of adapting to contextual information (*e.g.,* to specific users, or to physical and emotional states, etc.). By increasing computational power and emerging information technologies, assistance systems become feasible in the context of workplace recognition [4].

2.1.1 Cyber-Physical Systems

Recent evolutions in the manufacturing industry have shown that the development of Cyber-Physical Systems (CPS) enables synchronisation on the physical floor of the factory, often referred to as Industry 4.0. CPS is a transformative technology to connect physical assets with the computational capabilities of systems. This technology paves the way for the management of Big Data, examines interoperability among different machines to fulfil the goal of adaptability or self-awareness, and promotes machines' efficiency, collaboration, or resilience. Through the growing use of IoT technologies, sensors, cameras, or network machines, continuous data is generated (*i.e.,* Big Data). To handle this huge amount of collected data, computational power or cloud computing is required to extract useful information. Furthermore, it is worth mentioning that continuous data acquisition, timely adaptation, and

control are required to support dynamic environments such as the manufacturing industry (*i.e.,* real-time analysis (RTA)). If the computational effort of the system is too high, the processing of the data might take too long and impede the system from generating any meaningful value. To solve this risk, one might need to take more care during the development of the system, in relation to its rapid operability and lean design [1, 14, 15].

CPS contain two main functional components. The first is advanced connectivity that ensures real-time data acquisition from the physical world and information feedback from cyberspace. The second component is intelligent data management, including the ability to build a new cyberspace. This combination of data acquisition and further intelligent data management allows the system to be aware of the input, to predict a possible solution, and react appropriately according to it. For the purpose of this thesis, this might be a camera's photo shot of the workplace. Furthermore, the image of the workplace can be analysed using deep learning to extract meaningful information and to adapt the system according to the retrieved information [1, 16].

With the rise of CPS, physical work tasks constantly shift to digital ones. This shift should not lead to an additional burden, but should try to seamlessly integrate physical systems with digital infrastructure. The human worker should be at the centre of the technology and should benefit from its implementation. Since the inner complexity of the working tasks is increasing, there has to be more support from the outside so that the human worker can withstand this. A solution might be to optimise human-computer interaction (HCI) to benefit the user in the application of new digital technologies [2].

2.1.2 Types of Assistance Systems

The recent announcements of different assistance systems result in a research landscape lacking a coherent overview. Therefore, there are different approaches [1, 4] that organise heterogeneous assistance systems to reveal further developments. It is desirable to obtain a general framework that allows these assistance systems to be uniquely shaped based on the specific characteristics of a holistic framework. Applying such a framework can increase the understanding of new assistance systems, since a common vocabulary is established to identify different characteristics of assistance systems. This attribute identification allows easier decision making about future work, can be beneficial in learning about potential applications of assistance systems and in finding a common ground for discussions [4].

Table 2.1 presents an adaptation of the framework of assistance systems established by Fellmann et al. [4]. This framework is structured into four main categories with subjected features. Each feature can be described by multiple attributes, allowing to identify the general characteristics of an assistance system. The evaluated framework acts as an overview of the essential characteristics of the assistance system and tries to reveal possible use cases.

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.
TU Bibliothek Your knowledge hub WIEN

It is worth mentioning that this framework is developed based on the current information gained from existing assistance systems and might need adaptions in case of future developments. However, this approach is probably not suitable for comparing the performance of similar assistance systems. It can be seen rather as a rough classification of the wide variety of assistance systems [4].

Table 2.1: Adaptation of the assistance system framework established by Fellmann et al. [4]. The category **information** focusses on data *generation* and *presentation* of the result evaluated for the user. In the category **intelligence**: *state detection* reflects the ability to collect data under different conditions, *context sensitivity* refers to surrounding influences, and *learning aptitude* allows future behaviour to be improved by learning from previous data. The category **interaction** specifies the interface between humans and the assistance systems, where the feature *control* shows the executor of the job, *user involvement* classifies the level of cognitive or visual distraction by the used assistance system, *extent of immersion* describes inclusive surroundings for the human senses. Moreover, *input* and *output* should explain how the assistance system is fed with data and what it produces. The last category **system characteristics** contains the *transportability* of the assistance systems (*i.e.,* installation in other workplaces), *robustness* in the forms of surviving unintentional events or the *readiness level* (*i.e.,* simplicity of the system).

Category	Features	Attributes
Information	Generation	Manual Partly Automated Automated
	Presentation	Basic Intermediate Complex
Intelligence	State Detection	No Tools Products User
	Context Sensitivity	No Task Environment User
	Learning Aptitude	No Yes

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.

Category	Features	Attributes
Interaction	Control	Human Cooperation Machine
	User Involvment	Low Medium High
	Input	Traditional Modern
	Output	Visual Haptic Acoustic
	Extent of Immersion	None AR VR
System Characteristics	Transportability	Stationary Restricted Unrestricted
	Robustness	Low Medium High
	Technology Readiness Level	Low Medium High

The classification of assistance systems can be done in various ways. Perales et al. [1] describe another one, with the important features: *virtualization* refers to the possibility of monitoring physical processes or physical environments with the assistance system, *interoperability* describes the connection of machines and systems, *autonomization* describes the strength of the algorithm to make decisions, and *real-time availability* expresses the necessity to collect and analyse data in real time. However, Perales et al. mention that feature extraction is not enough to classify the assistance systems. Moreover, they established a two-step approach, which additionally classifies the system into the most relevant technologies, Cyber-Physical Systems (CPS), Internet of Things(IoT), Smart Data and Smart Factory. Applying this framework to this thesis, the technology for image-based workplace recognition is CPS and the features describe the following characteristics: *virtualization* to recognise scenes, *interoperability* for communication between the human and the system (*i.e., Microsoft HoloLens 1*), *autonomization* making decisions through the algorithm (*i.e.,* deep learning), and *real-time availability* adapting to dynamically changing workplaces [1].

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.

2.1.3 Digital Guidance for Workers

The technological and computational evolutions in recent years have led to new developments in digital assistance systems for human workers within factories. Although the assembly tasks for industrial workplaces become more complex, factories will not be without humans. Therefore, it is necessary to support existing worker capabilities with assistive technologies to cope with increasing diversity in work tasks [17]. A possible digital assistance system can be a system that includes augmented reality (AR) with context awareness as technology. When working on demanding tasks, context-aware augmented reality can support the worker by different work instructions at various workplaces. AR can be developed into an assistance system through head-mounted displays (HMD) or AR glasses. These instructions can be obtained by the headset without the need of the hands of the industrial worker. For example, displaying information on near surrounding devices (*i.e.,* terminals) takes time and is more cognitively demanding to transfer the information to the location where the information is required. The use of AR devices reduces head movement compared to tasks with tablets and consequently decreases the error rate in manual tasks [18].

If the instructions are different between the workplaces, HMD without context awareness cannot adapt to changing environments. Therefore, context awareness can be incooperated by an AR device by capturing images and classifying them according to their inheriting characteristics. The benefit of context-sensitive assistance systems is that they directly display the desired information in the position where the action is required. However, the use of a HMD with AR as digital assistance is not possible without the approval of the human worker, since the human-computer interaction must be accepted [19].

2.1.4 Human-Computer Interaction

If computers are to become intelligent, they have to naturally interact with humans in real-world applications. Therefore, they must have the ability to recognise the user and his/her intent. When looking at assistance systems, they aim to support the human worker, but sometimes lack natural interaction. In general, these systems produce additional output, which can be challenging for the limited cognitive resources of humans. Hence, cognitive-based systems learn and build knowledge by understanding natural language. Therefore, cognitive science with three independent parts: cognition, affect, and conation can be necessary to respect operations with assistance systems. Cognition, the ability to think and recognise surroundings, allows the creation of mental representations (*i.e.,* abstraction of reality) for further decision making. Affect is about perceiving emotions and feelings and conation points out the motivation and temperament. If a computer does not have at least one of these three abilities, it will produce mindless and impersonal interactions. Cognitive-based systems are able to put the content into context, providing confidence-weighted responses

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.
TU Bibliothek Your knowledge hub WIEN

with supporting evidence. In addition, they extend the capabilities of humans by augmenting human decision making and helping to make sense of the growing amount of data. Therefore, it is essential to enable these abilities for assistance systems to give the human worker an emotional connection with the machine and increase his/her acceptance. The objective is not necessarily to translate human-like behaviour into computer systems, but instead to enhance the cognitive capabilities of computers for improved Human-Computer Interaction (HCI). In this case, the decision-making process can incorporate some level of feelings of humans touching on their needs. The result might be greater pleasure at work or decreased cognitive overload for the human worker. However, an interesting upcoming development for modern workplaces could be the development of a system that assists human-to-human interactions [11, 20].

Of course, such systems are known to be abused if people are analysed and evaluated without their consent or by the manipulation of the decision-making process that is not in their benefit. But many technologies can be applied to either benefit the organisation or the human worker. However, the intention of assistance systems, CPS, or human-centred systems, is to benefit both the human worker and the organisation. These benefits should be visible by disclosing the system's working tasks and the business case. Moreover, the human worker should understand that through such inventions, his or her pricacy is not affected. The approval of workers for such systems is essential. Otherwise, misgivings of the system will damage the future human-organisation relationship [11].

From a technical perspective, the increasing complexity of robotics and environments overwhelms the cognitive capacities of a human worker. Traditional explicit control modalities (*e.g.,* keyboards, displays, GUIs, etc.) are added for each system without considering human cognitive capacities. Therefore, devices are continuously built with perceptional capabilities, shifting their explicit HCIs to implicit HCIs (*e.g.,* speech, gesture, etc.). Implcitit HCIs do not require a control modality to interact with a system. Moreover, computer understanding is gained by taking the assumption that the computer has a certain awareness of human behaviour. This means that a system must understand the context to be aware of the human environment, which can be further utilised for an implicit HCI [21].

For example, the human workers might have to handle conditions where they are operating on multiple machines simultaneously (*i.e.,* Multi-Machine Operations). Some of them will lack these capabilities, which will impede the entry level for such jobs. An assistant system that can determine the different workspace environments can enable the user to display workplace-specific information on an interactive device (*e.g.,* smart glasses). Therefore, such systems can reduce the cognitive overload of the user by offering a sufficient level of intelligence. Such systems must perceive the user's surroundings and autonomously evaluate a decision without any additional user input. By implementing these systems, some traditional control

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.
TU Bibliothek Your knowledge hub WIEN

modalities will become redundant and can be avoided. Furthermore, this allows for a minimal communication requirement between a workstation and the human worker [22].

If humans communicate with other humans, they identify objects in the real world by expressions. Through the rise of ML, it is now possible to mimic this behaviour and automatically refer to objects by machines. This identification of objects can be performed by extracting meaningful features from the objects. It is even possible to use whole images with meaningful features as input for classification. For example, this enables the classification of specific images of the workplace without identifying each object. With this implementation, the challenge for the assistance system of knowing the location might be solved. Therefore, the system can identify the workspace in which the human worker is currently located and can adapt its assistance to the user according to it. This dynamic system paves the way for the implementation of HCI in multi-machine operations without increasing the cognitive demands of the human worker.

2.1.5 Context-Awareness

Due to automation, workers must spend less time at single machines or work stations. These time savings lead to a change in work tasks, resulting in single workers operating multiple machines (*i.e.,* multi-machine operation). However, the shift from manual work tasks to planning and controlling requires the human worker to adapt to changing behaviours. These adaptations usually do not occur without user input, as those devices are not aware of their surroundings. Therefore, decreasing the cognitive stress of the user through the support of context-aware guidance systems allows adapting to changes in the physical environment. The context is described as any information that characterises the situation of relevant entities (*e.g.,* persons or workspaces) for system applications. This gained information can be processed and feed back to the user in an adopted way to support him/her in accomplishing the task [23].

Primarily, this means that the main goal is to present information to the user, which supports him/her in the decision-making process by the capablilty of context-awareness. However, to achieve this, the system must operate without additional user input. When considering the use study, the system should automatically adapt according to the workplace where the user is located, which can only be done by a context-aware system. Such a system can be implemented by any portable device (*i.e.,* hand-held device, smart glasses, etc.) and will only be required if the environment or the user's tasks are significantly changing. Furthermore, this context-aware system requires hardware and software that provide perception and computational capabilities. It can gain its perceptional capabilities through various ways (*e.g.,* sensors, audio analysis, video analysis, etc.) and its computational capabilities through processors [4, 23].

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.
TU Bibliothek Your knowledge hub WIEN

A context-aware assistance system can only be successful if it follows certain design criteria. Therefore, the system has to be usable for all human sizes, and it has to be operable in various physical environments (*e.g.,* different light conditions). Furthermore, the social aspect should not be neglected by implementing the system in a way that does not hinder human communication. Another requirement of the system is transportability, since the environmental objects (*i.e.,* machines or workplaces) are not located next to each other. Finally, the passing time helps determine moving parts or static parts and helps to identify the conjunction between two objects. In general, for the classification of entities, a unique identifier must be used. These are mainly physical objects with characteristic properties (*e.g,* proportions, spacings), which allow their identification. Position information is essential for defining orientations and spatial relations. Otherwise, it becomes quite challenging if the object in an image is not fully displayed or is too small-scaled for its determination. When interacting with the context, it is especially useful to further group the same context information into states. In addition to a state of context recognition, there should be states that allow us to determine the enter and leave of the context, to define the systems boundary conditions. Either to the lower boundary (*i.e.,* too close to the target) or to the upper boundary (*i.e.,* too far from the target). In case a state is entered, the system should have the capability to determine how long the user is within these states. This means that the longer the user is within this context, the slower the trigger the state will be left, giving the system the robustness to withstand unintentional events. Knowing the prelimitations of contex-aware assistance systems, its values and weaknesses can be disclosed [21, 23].

2.1.6 Values and Weaknesses of Digital Assistance

The everlasting progress of industry will not stop, increasing the expectations on human workers. Therefore, the development of assistance systems might counteract this behaviour and support the human worker in performing given tasks. However, it is not yet clear to what extent assistance systems will not contribute positively but will unintentionally lead to negative appearances. Considering that there are many different assistance systems, it might be challenging to provide values and weaknesses that agree on all of these. Therefore, the values and weaknesses of digital assistance through HMD are discussed [6].

The function of the selected assistance system is the perception assistance by context recognition and the cognitive assistance that can further support the decision-making process due to an information output. This assistance system has a direct influence on the human worker and just indirectly influences the performance of the workflow. With the application of this assistance system, the qualification requirement of the working task (*e.g.,* multi-machine operation) could be reduced through providing additional work instructions and therefore lowers the qualification level of the employee. The positive effect might be that this job position is suitable for a wider range of people. However, a human worker might have problems

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.
TU Bibliothek Your knowledge hub WIEN

when pure physical work tasks are shifting to cyber-physical tasks. The amount of virtual environment will grow in their jobs, and their acceptance of technology might be low. Furthermore, social interactions with other humans might get disrupted. By wrongly designing the system, it will present digital information to the human worker in inappropriate situations or by reducing his field of view and distracting his attention. During human-to-human interaction, there should be the possibility of turning off these systems to increase natural communication.

Human nature is feeling unwell in conditions that are attached to uncertainty. For example, if a human worker performs a multi-machine operation and comes into a situation where it is unclear what the next step is or what the right decision is, the work pressure increases dramatically because a wrong decision is attached to a negative performance of his or hers workstation. A solution might be an assistance system that can support this decision and, therefore, decreases the pressure on the human by removing his responsibility in case of an error. In case of failure, guilt lies in the assistance system and not in the human worker, making his or her work tasks more comfortable. Moreover, early detection in the case of failure and correction in the case of wrong decisions can further relieve the human worker.

Cognitive assistance enables the human worker to reduce his effort in handling the task and increases his flexibility. A human worker may be unfamiliar with the system at the beginning of the job and may have a longer adjustment period for this system. However, the information displayed, provided by the decision-making process of the assistance system, can act as a source of learning and continuously train the human worker to perform a task. Continuous learning with this system could make it redundant one day, and the human worker can perform the task without active assistance [3].

2.2 Deep Learning

As discussed in the previous chapters, relief of the human worker's strain by assistance systems can bring benefits in cognitive assistance of the human worker tasks. Although, physical devices (*i.e.,* cameras) are capable of capturing the surroundings of the human worker, a system which can process this information is required. Therefore, the images recorded by the cameras must be evaluated by an algorithm that can capture context. However, history has shown that abstract and formal tasks that are difficult to describe by human beings can be easily automated. These tasks (*e.g.,* identification of objects) that humans efficiently perform are subjective and intuitive, making them difficult to solve for any algorithm. Moreover, developing an algorithm that addresses this challenge is an enormous investment, since many conditions and boundaries must be determined. Doing this without any learning ability of the algorithm requires huge expenses, and such systems are mostly error prone and rigid. Nevertheless, we know that tasks such as image classification require tremendous flexibility

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.
TU Bibliothek Your knowledge hub WIEN

since each image will look different (*i.e.,* different light conditions or a variation in the exposure position), even the existing object in the image. Therefore, the solution to this issue is to allow the computer to learn from experience and accomplish the desired task based on experience under controlled performance measurements [24].

Before elaborating on technical aspects, definitions of the terms artifical intelligence (AI), machine learning (ML) and deep learning (DL) are necessary. Many times these terms are used interchangeably, but their principles differ. Artificial intelligence is referred to as a generic term (*i.e.,* without any explicit technology) that gives computers the ability to mimic human behaviour and solve problems. However, the other two terms describe a more technical meaning of how the computer is able to learn human behaviour. Therefore, machine learning , a branch of AI, is a technology that uses statistical techniques to learn from previous information and experiences. By learning from past information, the machine can make future decisions. Lastly, deep learning, a branch of ML, is a technology that was inspired by the human brain network by using artificial neural networks (ANN) to solve problems. It builds automatically representations of patterns from the data, without manual feature engineering [25].

Besides the three mentioned terms, Computer Vision (CV) focusses on helping to give computer visual awareness of the observed world. This is different from ML which gives machines the ability to learn. In contrast, CV is focused on giving machines the ability to see. It breaks down and tries to interpret visual information instead of processing statistics or simulated data.

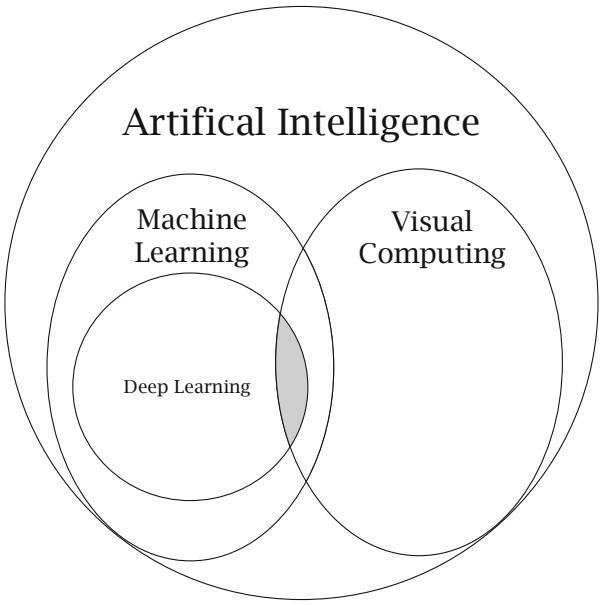


Figure 2.1: Illustration of the relationship between ML, CV and DL under the branch of AI. Adapted from [25].

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.

Visual information from images remained opaque for a long time, and to get the most out of the image data, computers must understand the content within the image. In general, CV is challenging due to the inherent complexity of the visual world and a true vision system must extract characteristics from each scene of an infinite number. Knowing the intersection of CV with ML and DL, we can illustrate the topic we focus on during this thesis (see Fig. 2.1) [26].

2.2.1 How can an Algorithm learn?

By knowing the ability of the algorithm to make decisions on its own, one can ask the question why it is called learning? Therefore, Mitchell [27] provides the following definition: "A computer programme is said to learn from experience E with respect to some class of tasks T and the performance measure P if its performance in T, measured by P, improves with experience E."

The task T can be understood as how the system should process an example (*e.g.,* collection of characteristics) that has been measured from an object. In the case of computer vision, the characteristics are usually defined arrangements of pixels in the image. Depending on the selected task (*i.e.,* analysis method), different results can be expected. For example, regression is a task in which the input is converted to one output of continuum values. On the other hand, the classification task involves input (*e.g.,* pixel values of an image) that is specified in one of k categories. Therefore, the learning algorithm has to produce a function f (see Equation 2.1) that assigns a probability distribution over all classes [24].

$$f : \mathbb{R}^n \rightarrow \{1, \ldots, k\} \tag{2.1}$$

To evaluate the abilities and limits of the machine learning algorithm, a quantitative measurement of its performance is required. Depending on the task used, different performance measurements should be performed. Looking again at the image classification task, the accuracy (*i.e.,* P) is identified by the proportion of correct outputs that the algorithm can produce for a single caterogry. In contrast, the error rate (*i.e.,* P) is the proportion in which incorrect outputs are identified. Both accuracy and error rate can be described on a scale of 0 to 1, where for a binary problem the accuracy of 0 indicates a wrong assignment and 1 a correct assignment [24].

The experience of the ML algorithm can be defined by a dataset containing various samples. Depending on the class of learning problem, the samples of the dataset can be labelled or not. Regardless of the above, the classification performance of the algorithm is evaluated on the dataset. Although the size and quality of the dataset reflects the expierence, incorrect or insufficient samples decrease the performance of the classification task [24].

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.
TU Bibliothek Your knowledge hub WIEN

2.2.2 Machine Learning Basics

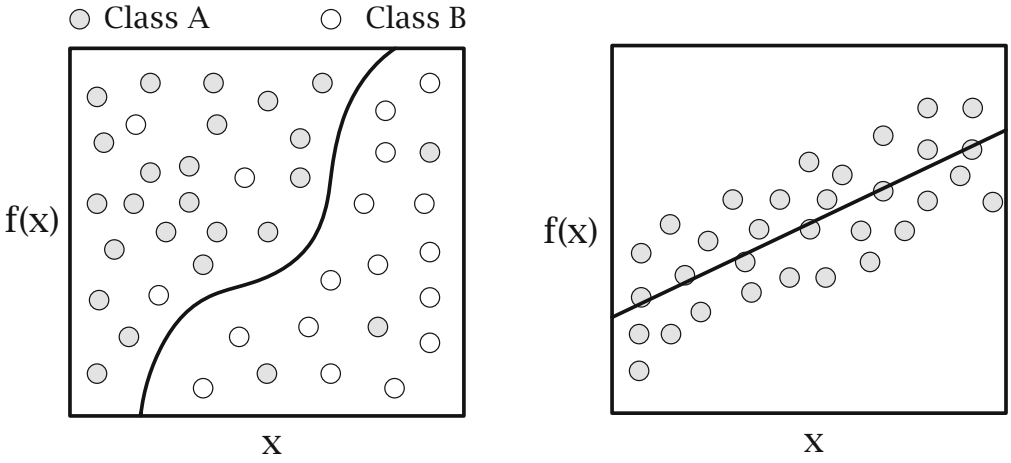
The hierarchical structure of AI, ML and DL concludes that to master deep learning, prior knowledge of ML is required to understand its connections. Machine learning is based on applied statistics and uses computers to approximate complicated functions. The complicated, mostly nonlinear function can be made up of simpler (*i.e.,* linear) functions by joining them together. Therefore, the algorithm is capable of solving complex applications. Furthermore, the performance of the output of the function is enhanced by iterations through an optimisation procedure (*e.g.,* gradient descent, etc.). During each iteration, the algorithm learns important information for future decision-making. In following sections are the learning techniques and ML fundamentals discussed [24].

Classes of Learning Problems

ML is a broad term, and the learning procedure can be selected depending on the problem statement. In general, learning problems can be divided into the main classes, supervised, unsupervised, and reinforcement learning. These classes show different approaches to information handling and algorithm training.

Supervised Learning Algorithm The goal of supervised learning is to uncover relationships between data characteristics that have already been measured. This gives the learning algorithm the ability to learn the characteristics of each data point, which is labelled into a category. Since the algorithm already knows the output during learning, it can assign the features of each data point with supervision to the different classes. Each category represents a class with characteristics that are used to train the algorithm on these data. Finally, the learnt features are predicted on unseen data to classify unlabelled images. More mathematically, these supervised learning algorithms estimate the probability distribution $p(y|\boldsymbol{x})$ using the maximum likelihood estimation to find the best parameter $\boldsymbol{\theta}$ vector. The probabilities of assigning a $\boldsymbol{x}$ to one y must be evaluated for all classes. If there are multiple classes, the clear assignment of $\boldsymbol{x}$ to a class may not be that simple. Therefore, training through multiple epochs is required to optimise $\boldsymbol{\theta}$, allowing us to assign $\boldsymbol{x}$ to one y more clearly. The main categories of supervised learning are classification, which predicts a class label, and regression, which helps to find correlations and addresses a numerical label. Well-known examples are the MNIST handwritten digit algorithm for classification [26] and the Boston house price algorithm [28] for regression. In general, classification is a special case of regression in which variables are categorical (*i.e.,* one of a fixed number of possible classes) [24, 28, 29].

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.



(a) Classification (b) Regression

Figure 2.2: Schematic representation of (a) classification by two different classes and (b) regression by linear fit. Adapted[1]

Typically, for supervised machine learning, large-labelled datasets are of great importance for high-quality training. Without a good dataset, the developed ML algorithm will have poor performance. Therefore, collecting data, classifying them and tagging them one by one is essential for the desired application and further research in various domains. Furthermore, each available open-source dataset drives the development of deep learning technologies. However, the dataset can be split into different parts (*i.e.,* training, validation, and testing) to perform different tasks. The training set is used to optimise the dynamic parameters (*i.e.,* weights and bias) of the algorithm, the validation set is used to find its static parameters (*e.g.,* learning rate, batch size, etc.), and the test set evaluates its performance. Unlikely splitting the dataset into different chunks makes the test set susceptible to uncertainty around the average test error, if the sizes of the different dataset parts are small [30].

Unsupervised Learning Algorithm In contrast to supervised learning, the input of the dataset does not include an annotation for unsupervised learning. For unsupervised learning, beneficial properties outside the structure of the dataset have to be extracted without any instructor or assistance. The main categories of unsupervised learning are clustering (*i.e.* finding groups in the dataset) and density estimation (*i.e.,* summarising the distribution of the data). During clustering, the algorithm has to classify the meaningful features into the same categories. Therefore, a relationship between the characteristics of the data that have not been designated has to be determined. For example, in image clustering, the learning algorithm has to be trained with a dataset that does not contain labels. The network has the general goal of finding similarities inside the input data to make appropriate clustering,

[1]https://www.javatpoint.com/regression-vs-classification-in-machine-learning, last accessed on 20.11.2022

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.
TU Bibliothek Your knowledge hub WIEN

keeping the boundaries of clusters as simple as possible. Another topic, Auto Encoders (AE), addresses unsupervised learning by reconstructing the training data and minimising the reconstruction error without any data labelling. It consists of an encoder function that tries to convert the input data into a different representation and a decoder that converts this representation back to its original state. From a mathematical perspective, the encoder transforms an input $\boldsymbol{x}$ into a hidden representation $\boldsymbol{h}$. Within the state $\boldsymbol{h}$ usually a lower dimensionality is occurring and information can be retrieved. By trying to convert $\boldsymbol{h}$ back to its original state, the difference between the original input and decoded representation (*i.e.,* ideallly close to the origin input) can be measured. However, it is still controversial whether AE are unsupervised learning approaches or not. But considering the definition that unsupervised learning approaches learn from input without any labelled output, AE can be interpreted as unsupervised learning [15, 24, 28].

Reinforcement Learning In reinforcement learning (RL) the algorithm is not based on a fixed dataset, but rather interacts with its environment to learn from previous actions. This interaction is only possible if the learning algorithm can receive information from the environment to determine its current state. The derivation of states from the environment is especially useful if new states arise. Then, depending on the information received from the environment, an agent (*i.e.,* a system that learns and makes decisions) can decide what actions should be taken. These actions are evaluated by a reward function and are rewarded or penalised based on whether the agent chooses a good or bad action. However, these actions are categorised into epochs in which the decision-making strategy is optimised by maximising the cumulative reward. In this way, the reinforcement model saves well-working actions and can improve over time. The agent can either explore new actions (*i.e.,* random actions) or exploit the historical decisions predicted by the model during the training stage (*i.e.,* model actions). Generally, with increasing time, the model is programmed to explore less and exploit the previously learnt actions to reinforce the learning strategy. However, the reinforcement learning approach has a long convergence time and underlines the drawbacks of high computational requirements compared to other learning approaches. Due to these drawbacks, RL is best suited for applications with various unknown states [14, 24, 25].

Generalisation, Underfitting and Overfitting

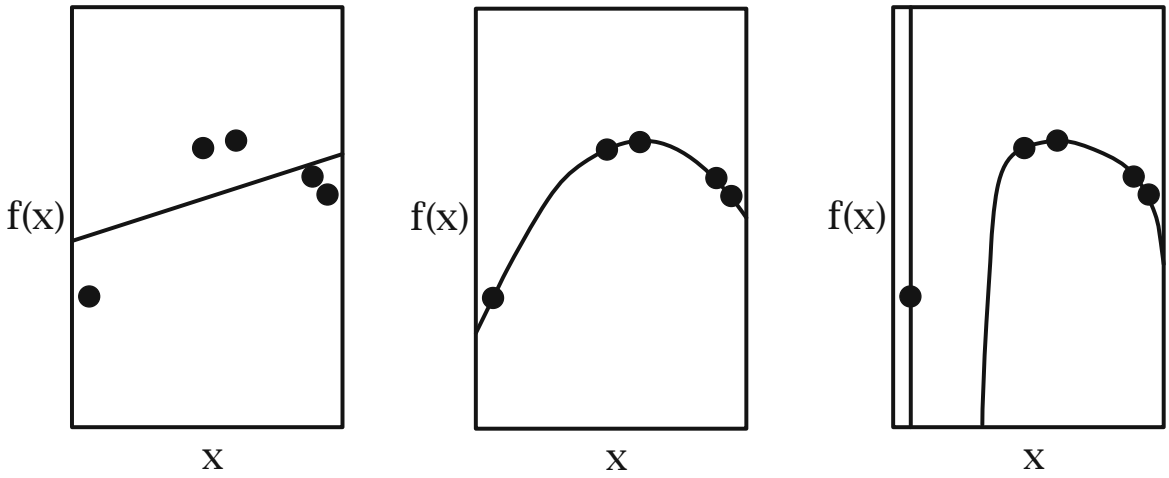
The challenge of the algorithm in machine learning is that it has to perform well on unseen data, which is called generalisation. This characteristic of ML is not restricted to any learning strategy, but in the following will be discussed for supervised learning. To give the algorithm the ability to perform well on unknown data (*i.e.,* test set), it had to previously learn from known data (*i.e.,* training set). In general, learning from large training datasets is beneficial for generalisation, since the algorithm remembers important characteristics more

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.
TU Bibliothek Your knowledge hub WIEN

precisely. Due to the labelled data in the training set, performance can be evaluated and optimised, reducing its training error. However, in addition to optimising the training error, the error value on a new sample (*i.e.,* generalisation error or test error) must be considered. Since the data on which the network runs on during deployment is not labelled, evaluating the generalisation error is difficult. To handle the uncertainty of the generalisation error, similarities between the test and the training set are used. Therefore, the effort lies in creating identical distributed training and test sets that are independent of each other to impede fake performance. By impeding the arbitrariness of the training and test sets, the generalisation error can be approximated by the training error, since both datasets underline the same sampling process. In this case, the two primary goals of generalisation are to reduce the training error and to increase the similarity of the training and test error without violating the independence of each other.

There are two crucial terms (*i.e.,* overfitting and underfitting) when discussing a poor generalisation, which occurs due to different causes. While underfitting represents a situation where the error rate of the training set cannot be sufficiently lowered, overfitting occurs when the gap between the training and test error is too large. Both phenomena can be described by the capacity factor. The capacity can be changed by changing the model's ability to generalise, for example, by changing its complexity or its learning behaviour (see Chapter 2.2.3). An appropriate capacity is chosen if the algorithm is suitable for various functions. However, a low capacity refers to underfitting and a high capacity refers to overfitting. In the case of an example of fitting a polynomial function (see Fig. 2.3), the capacity can be controlled by choosing the function boundaries (*i.e.,* order of the function). The order of the polynomial function must be selected before training, and its right choice is important for the working algorithm. Such a static variable is called a hyperparameter and will be discussed in Chapter 2.2.2. In general, the best capacity is achieved when the ML algorithm is adjusted to the true complexity of the task. However, in case of overfitting, the learning algorithm finds a suitable function that reduces the training error, but does not fit the structure of the recognition task. If no test set is available to validate the test error, these overfitting principles are often referred to as Occam's razor, which states that if similar solutions of different learning strategies are available, the simplest should be selected. Furthermore, overfitting occurs when individual characteristics including noise are learnt by the algorithm, which negatively impacts generalisation ability. Therefore, strategies are used that tend to reduce this behaviour. This includes data augmentation, early stopping, dropout, small convolutional kernel size, cropping, and warping [24, 31].

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.



(a) Underfitting (b) Appropriate capacity (c) Overfitting

Figure 2.3: Schematic representation of fitting of the same dataset by three different polynomial functions. (a) A linear function was chosen for fitting the data points, which cannot capture the curvature resulting in underfitting. (b) A quadratic function seems to fit the datapoints perfect, resulting in the optimal capacity. (c) The selection of a polynomial function of a higher order can pass through all datapoints perfectly but does not reflect the true complexity of the structure. This lead to erroneous results on unseen data point [24].

Hyperparameters

Developing an algorithm ML includes the choice of many different parameters. These parameters can be distinguished into trainable parameters (*i.e.,* parameters inside the model; learnt by the solver), nontrainable parameters (*i.e.,* inside the model; not learnt by the solver) and hyperparameters (*i.e.,* outside the model; cannot be learnt by the solver directly). The difference between nontrainable parameters and hyperparameters is that while nontrainable parameter are within the model (*e.g.,* frozen weights of an layer, etc.), hyperparameters define the model's structure from the outside. Therefore, hyperparameters are determined independently and are usually not learnt by the algorithm itself. Instead, those parameters are set statically and allow one to change the behaviour of the learning algorithm. Examples of hyperparameters are the number of layers created, the size of the layers, or even the choice of the activation function. Looking at the different results in Figure 2.3, the variation in order of the functions is striking. This degree can be seen as a hyperparameter that controls the capacity of the learning algorithm [32].

Generally, no example out of the test set should be used to evaluate the hyperparameters, to reach impartiality. Therefore, the training set is divided into two different sets. One set is used to determine the weights (*i.e.,* training set). The other subset is referred to as the validation set, which allows the hyperparameters to be continuously updated according to the optimisation of the generalisation error. Usually, a fifth of the training set is used for valida-

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.
TU Bibliothek Your knowledge hub WIEN

tion and the rest is used to optimise the training set. After optimising the hyperparameters, the overall performance of the network can be improved [24, 32].

2.2.3 Artificial Neurone and Neural Networks

The idea of NN has been developed and inspired by investigations on how humans are capable of capturing context. Therefore, the neural system of the human brain with the ability to learn and memorise was mimicked by artificial neural networks (ANNs), consisting of interconnected artificial neurones (ANs). These artificial neurones are combined within multiple layers and operate in parallel, illustrating the nerve activities of the biological neurones. The general goal is to minimise the network output error by optimising its parameters with the best possible selection [15, 33].

The artificial neurones are inspired by biological neurones within the human brain. Each artificial neurone (see Fig. 2.4) receives multiple inputs (*i.e.,* $x_0, x_1, ...x_n$) which are processed by multiplication with certain weights (*i.e.,*$w_0, w_1, ...w_n$). The weights are necessary adjustment values that try to optimise the output. For the task of image classification, the inputs are represented by the pixel values of an image, whereas the weights express a factor to change the colour channels or the pixel intensity of the input. In general, multiple artificial neurones are controlled by weights and a bias (*i.e.,* to adjust any offset) and can be combined into a neural network, which is a parametric model. This model is defined as $f(x;\boldsymbol{\theta}) = \boldsymbol{W} \cdot x + b$ with $\boldsymbol{\theta} = (\boldsymbol{W}, b)$, while $\boldsymbol{\theta}$ describes the parameters, consisting of a vector of weights $\boldsymbol{W}$ and a bias b. A special case of this model is a linear classifier with one artificial neurone.

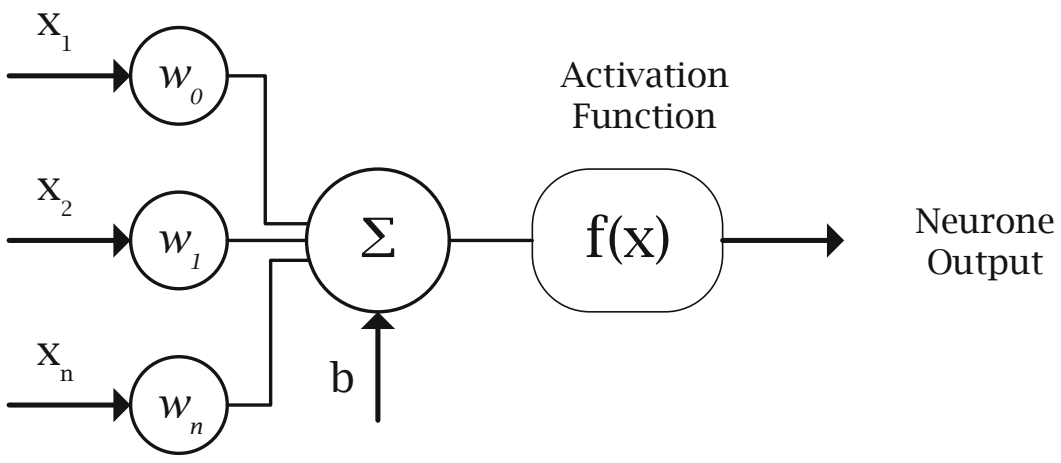


Figure 2.4: Schematic representation of the structure of an artificial neurone, including weights and a bias as changable parameters and an activation function as nonlinearity. The illustration is adapted from [33].

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.

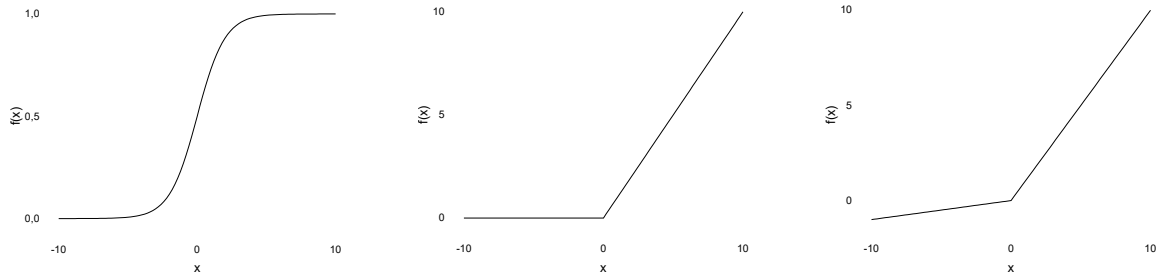
Furthermore, the relevance of the individual neurone is implemented by an activation function. It detects the importance of certain neurones and can act as a switch by turning off individual neurones. Without an activation function as nonlinearity, nonlinear problems cannot be solved by this parametric model, since a linear model in combination with other linear models still represents a linear model [33, 34].

Activation Function - Nonlinearity

Describing context within a scene is a complex patterns that cannot be described by linear correlations. Therefore, using activation functions (*i.e.,* nonlinearities) within NN enables the detection of complex shapes inside a structure. The activation function decides whether a neurone is activated and transforms a neurone signal based on a nonlinear monotonic function. If the activation function returns its input without any transformation (*i.e.,* identity function), every layer of NN will only be a linear transformer and the network will be a linear regression model. In Chapter 2.2.2, it has been shown that the linear regression model is sufficient for linear problems but not for complex higher-order problems [24, 26].

Overall, there exist many different approaches for activation functions. However, the most popular activation functions are sigmoid (see Fig. 2.5a), rectified linear activation function ReLU (see Fig. 2.5b) and leaky ReLU (see Fig. 2.5c). Among all activation functions, the sigmoid function is biologically inspired by mimicking neuronal activation in the brain. Although it has a regularising effect (see Chapter 2.2.3), forcing the features into a range between 0 and 1, it has the downside of relatively flat gradients during backpropagation. Parameters that receive small gradients work only for shallow networks. For deep neural networks with sigmoid activation functions, repeated multiplication of small slopes causes them to be close to 0. This phenomenon is called the vanishing gradient problem and can be solved by using steeper activation functions. Therefore, the rectified linear activation function ReLU allows a much greater gradient than the sigmoid function to overcome this problem. However, ReLU has the downside of dying neurones, where a negative input will lead to the deactivation of neurones without any representation of information. This is caused by negative input values that eventuate in an output of 0. An neurone with 0 output does not contribute to the model and therefore the neurone or node is called "dead" (*i.e.,* neurone remains dead). Therefore, leaky ReLU tries to mitigate dying neurones (*i.e.,* small negative slope function for values less than 0) by improving the ability to send gradients backwards. The small negative gradient of leaky ReLU helps defuse the problem of vanishing gradients. Besides these three mentioned activation functions, any non-linear function can be used [28, 32].

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.



(a) $\sigma(x) = \frac{1}{1+e^{-x}}$ (b) $ReLU(x) = max(0, x)$ (c) $ReLU_{leaky}(x) = max(0.1 \cdot x, x)$

Figure 2.5: Schematic representation of (a) Sigmoid, (b) ReLU, and (c) leaky ReLU activation functions on an input scale of [-10,10]. Adapted[2].

Regularisation

It is essential to find the best representation of data, which can be supported by penalties or restrictions. Therefore, regularisation is a technique that restricts training behaviour and increases the performance of the model on unseen data. It can reduce generalisation errors, especially overfitting, at the expense of increasing training errors. Adding a penalty called regularizer allows optimising the learning behaviour by slowing the convergence of the model. Although regularisation is not restricted to NN, its techniques to achieve this are dependent on the problem statement. For this reason, regularisation techniques are specially discussed for NN applications [24, 26].

For NN, regularisation can be achieved by techniques such as data augmentation, weight decay, early stopping, learning rate schedules, or dropout. All of these techniques slow down training and therefore decrease overfitting behaviours. However, two relevant techniques, dropout and data augmentation, will be discussed in more detail. Compared to a baseline NN, the regularised model shows an improved performance output on unseen data [24, 26].

While in a traditional NN all neurones are related to neurones in the surrounding layers (see Fig. 2.6a), the NN with dropout (*i.e.,* hyperparameter) enables the model to randomly deactivate neuronal activities inside the network (see Fig. 2.6b). Since the deactivated neurones can be seen as dead for this training step, their information transfer has to be taken by the remaining neurones. Consequently, the ability to learn is more difficult, leading to regularisation of the model and avoidance of overfitting. Therefore, for every optimisation step in the neural network, a different neurone connection is generated, except for weight sharing between the networks. In dropout, a single neurone cannot rely on its closely located neurones because they might be deactivated. Consequently, updating the neuronal weights is no longer performed simultaneously, preventing all neurones from convergence to the same

[2] https://medium.com/@shrutijadon/survey-on-activation-functions-for-deep-learning-9689331ba092, last accessed on 20.11.2022

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.
TU Bibliothek Your knowledge hub WIEN

goal. Hence, these neurones learn to extract features that are based on the conjunction with various neurones, resulting in increased robustness of the overall algorithm [35].

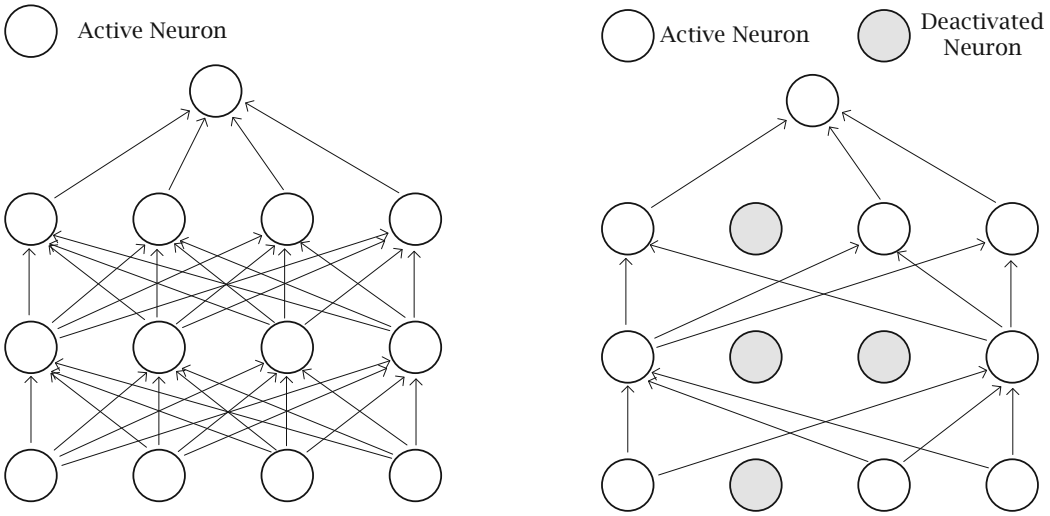


Figure 2.6: Schematic representation of (a) neural network with two hidden layers, and (b) neural network with two hidden layers and temporarily deactivated neurones. This illustration is adapted from [36].

Another regularisation technique, data augmentation, increases the model generalisation ability. Without data augmentation, the dataset might be too small. The consequence would be that the NN memorises the training set, leading to overfitting. Therefore, data augmentation increases the training set by artifically generating new samples by altering existing ones. In general, the data augmentation procedure is performed depending on the type of input data. For image classification, the images can be rotated by certain degrees, cropped to certain percentages, or zoomed in to increase the data variety. This technique adds noise to the training set and is mainly required if the richness of the existing dataset is not sufficient to train the algorithm appropriately. However, this technique must be used with caution. Creating artificial input for training can generate incorrect feature maps by omitting important characteristics. For instance, if an image of digit 6 is data augmented upside down, it suddenly transfers into a 9, which will trick the algorithm. In general, the use of data augmentation increases not only the amount of data it trains on, but also the robustness of NN [25, 37].

2.2.4 Neural Network Training and Evaluation

By knowing the fundamentals about ML and NNs, the question arises on how these networks are trained. Therefore, neural network training can be performed in various applications and

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.
TU Bibliothek Your knowledge hub WIEN

can show differences for each type of neural network. Despite the multiple applications, the fundamentals will remain similar. During this chapter, neural network training is discussed with a focus on image classification. Furthermore, performance metrics are mentioned, which help to evaluate the output of the model.

Feed Forward and Backpropagation

The multilayer configuration of NNs clarifies that training these networks is not that simple. Therefore, the training procedure consists of two phases (see Fig. 2.7), where the first phase (*i.e.,* feed forward) is the propagation of inputs through hidden layers (*i.e.,* the true value is unknown) to the final layer. During this phase, the inputs of each layer are multiplied by weights to generate an output, which is the input for the next layer. In the final layer, an input-to-output mapping is performed through a function that calculates the error (*i.e.,* loss) with respect to the target value. After evaluating the loss of the feed forward propagation, adaptations of the model parameters $\boldsymbol{\theta}$ (*i.e.,* weights and biases) are performed according to minimise it based on an algorithm. This procedure, called gradient descent, translates the error from the loss function through all layers to the first layer. During each layer, its parameters are adjusted to minimise those errors as much as possible. While the opimisation function is only used for the adjustment of the parameters using the gradient, backpropagation refers only to the method to calculate the derrivative of the loss function $L(\boldsymbol{\theta})$. In this case, backpropagation (*i.e.,* second phase) has to calculate all derivatives of the functions from the last layer to the previous. Therefore, any function that represents a layer must be real and differentiable. This training process is repeated for a specific number of iterations until the loss function is satisfied. A common way to save computational resources is to package these iterations into batches and update the parameters in each batch. In this case, one batch consists of multiple inputs that calculate the loss. Subsequently, these losses of all inputs in one batch are averaged and backpropagated. When multiple inputs are bundled in batches, volatility in the output can be flattened and backpropagation for each input can be avoided. If all the data is packed into one batch, the computational resources are likely not large enough to save the input into memory. On the other hand, if a batch consists of one input, the loss is calculated for each input, consequently lowering the training speed. In general, backpropagation is a procedure that calculates the loss for each weight by considering the chain rule backwards through the network [20, 24–26].

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.

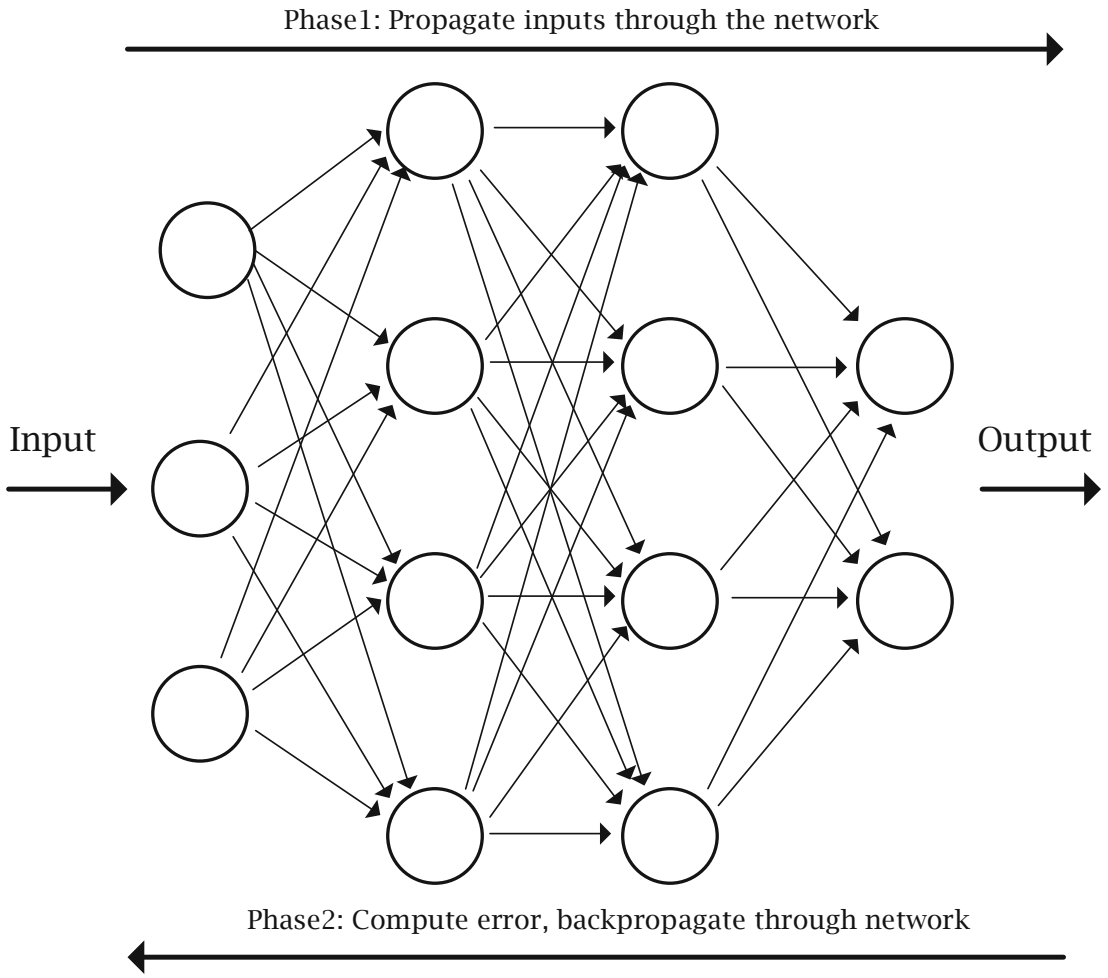


Figure 2.7: Schematic representation of feed forward propagation and backpropagation within a neural network [20].

Optimisation function and Momentum

The optimisation function is used to train the network by adapting the changeable parameters (*i.e.,* weights and bias) of NNs to minimise the loss function that evaluates its performance [32]. This procedure can be performed using different strategies. However, the most common is the gradient descent. Gradient descent is an iterative first-order algorithm to minimise the loss function $L(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ are called parameters. To calculate the gradients, a differential optimisation function with real parameters is required. These parameters are then updated in each iteration by $\boldsymbol{\theta}$=$\boldsymbol{\theta}$-$\alpha \cdot \nabla L(\boldsymbol{\theta})$, where α represents the learning rate (*i.e.,* hyperparameter). The gradient $\nabla L(\boldsymbol{\theta})$, is a vector that contains all partial derivatives of $L(\boldsymbol{\theta})$ and can be calculated by backpropagation. The learning rate is strongly necessary to update the parameters based on a certain step size, otherwise results might lead far astray. In general, gradient descent can follow various approaches. While the stochastic gradient descent approach (SGD) has one input sample, the batch gradient descent includes all samples.

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.
TU Bibliothek Your knowledge hub WIEN

By looking at these two extremas, another approach, mini-batch gradient descent, has been established. It takes a batch of samples and computes the gradient for the loss of the mini-batch. However, when using gradient descent, the gradient is usually an approximation and cannot be solved analytically. Therefore, the optimisation algorithm might not find a local minima of the function, but has the advantage of arriving at a very low value of the function that is useful enough. The risk remains, that the gradient descent might find poor local minima, which does not satisfy the overall goal of optimisation within NN. However, practise has shown that regardless of the initial condition, different local minima achieve equal results [24, 28].

By knowing that gradient descent can run into a local minimum that is not sufficiently low, it might happen that this minimum could not be passed through without any support, since α is too small. Therefore, there are strategies to avoid this. The so-called momentum helps the gradient descent to handle difficult loss functions better. Momentum is introduced to include inertia for gradient descent, by a velocity v. This velocity is continuously updated according to $v{=}\beta\cdot\ v{-}\alpha{\cdot}\nabla L(\boldsymbol{\theta})$, where $\beta{\in}\ [0,1)$ refers to the momentum (*i.e.,* hyperparameter). The parameters are then updated according to $\boldsymbol{\theta}{=}\boldsymbol{\theta}{+}v$. Without momentum, once the optimisation flow is interrupted (*e.g.,* by the example of SGD, each batch), the algorithm will change the direction according to the steepest gradient obtained. This results in oscillations along the optimisation path, which can be flattened by momentum. In general, the algorithm can involve past gradient evaluations in future behaviours [24, 28].

Loss function

Once we know how the general training flow works, more care can be taken into its individual components. While training is the state that adjusts the model weights, learning is the models ability to minimise the loss. Loss of a network determines the difference between the output of the model generated by the current parameters and the expected output. It tells the optimiser how well it is performing and acts as a penalty in the network for incorrect predictions. Furthermore, the objective of most DL approaches is based on minimisation of loss by gradient descent. In contrast, a high loss value indicates that the model does not fit well. However, the loss function defines the way the loss of the last layer of the network is calculated. Generally, the loss function depends on the problem we want to solve. For typical regression problems, the mean square error (MSE) is used (see Equation 2.5)[24].

$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n} \tag{2.2}$$

Where y_i is the true labelling, $\hat{y}_i$ refers to the predicted value, and n represents the number of error parameters. In MSE, $(y_i - \hat{y}_i)^2$ results in a preference for lower error values. The benefit of this loss function (*i.e.* convex shaping) is that larger error values result in a bigger

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.
TU Bibliothek Your knowledge hub WIEN

step to the next state if the predicted value is further away from the target. A convex loss function can prevent the optimiser from getting stuck in a high training error. However, this loss function is not suitable for classification, as it is hard to judge about any difference between the true and predicted labelling. Therefore, the cross-entropy loss function is used for classification tasks (see Equation 2.3)[24].

$$CrossEntropyLoss = \sum_{i=1}^{n} -(y_i \cdot log(\hat{y}_i) + (1 - y_i) \cdot log(1 - \hat{y}_i)) \tag{2.3}$$

This function measures the difference between the true labelling y_i and the predicted value $\hat{y}_i$ for all categories n. To get an idea of the function, we look at an example of predicting an input to one true class of n categories. For all categories that are not true, $y_i = 0$ results in a cross-entropy loss function of $-log(1 - \hat{y}_i))$. On the other hand, the category with the true labelling $y_i = 1$ has a loss function of $-log(\hat{y}_i)$. Therefore, it is immediately visible that a high true value and a low false value refer to a low loss value. Theoretically, the best loss value is 0, which is indicated by $\hat{y}_i = 1$ for the true class and $\hat{y}_i = 0$ for all other categories [28].

Performance Metrics

The trained model must be validated to show its effectiveness. Therefore, performance metrics are selected to check different functionalities of the algorithm. Loss is the most well-known performance metric that is minimised to adjust the training parameters. It is the distance between the target value and the values predicted by the model. In other words, loss is the value that represents the summation of errors in a model. Generally, the loss value is a subjective metric whose value depends on the loss function and the preprocessing of the data points. Hence, its distance between the true value to the prediction made by the model depends on the range of the data values.

For classification problems, accuracy is a more interpretable performance metric. It describes the number of errors made on the data, by what percentage of the test data are correctly classified. Both, accuracy and loss, are not equal, and their correlation allows room for interpretation. The best case is high accuracy with low loss, which refers to low errors in a few data. Although loss is the perfect choice for optimising training, accuracy better represents real-world applications, but with the drawback of losing information about distances. Generally, to interpret the results for multiclass classification in a more precise way, confusion matrices can be used (see Fig. 2.8). A confusion matrix shows the correlation of each class prediction and can be described for a two-class problem by four quadrants (*i.e.,* True Positive (TP), True Negative (TN), False Negative (FN) and False Positive (FP)). The descending diagonal represents the TP and TN values, which are desired to be high for a

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.
TU Bibliothek Your knowledge hub WIEN

good classification. Below the diagonal are FP and above the diagonal are TN. Although for some classification applications (*i.e.,* recognising cancer) their difference is crucial, for our application they will not be distinguished.[3]

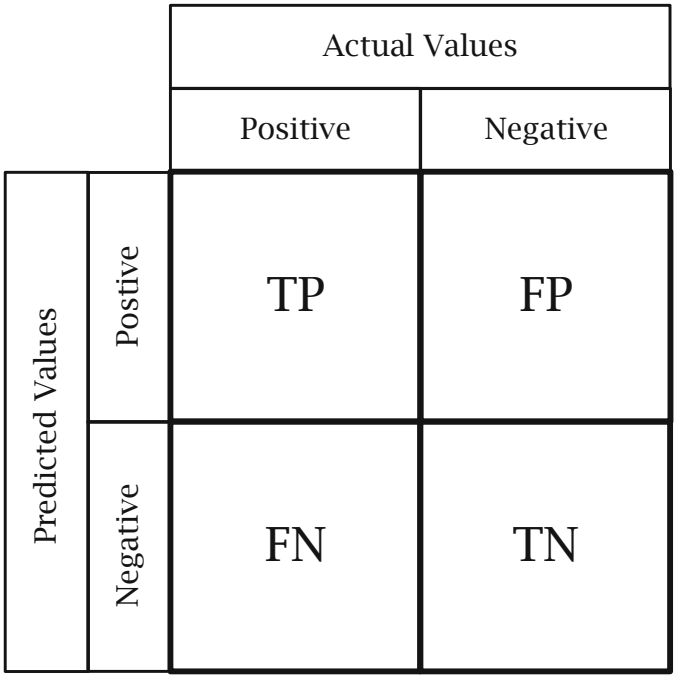


Figure 2.8: Schematic representation of a binary confusion matrix with the actual values on the horizontal axis and the predicted values on the vertical axis. Adapted[3].

Another performance metric, F-measure (F_1-score), shows how the model performs relative to all classes and is a more accurate performance metric for multiclass classification than accuracy $\frac{TP+TN}{Total}$.[4] F_1-score takes FP and FN into account, and its definition includes precision $\frac{TP}{TP+FP}$ and recall $\frac{TP}{TP+FN}$. The final F_1 score can be calculated for the total number of classes (see Equation 2.4).

$$F_1 = \sum_{i=1}^{n} 2 \cdot w_i \cdot \frac{precision_i \cdot recall_i}{precision_i + recall_i} \tag{2.4}$$

Where n is the number of samples and w_i the weighing factor of the imbalance within the classes [38].

[3]https://medium.com/analytics-vidhya/what-is-a-confusion-matrix-d1c0f8feda5, last accessed on 20.11.2022
[4]https://medium.com/@shrutisaxena0617/precision-vs-recall-386cf9f89488, last accessed 24.11.2022

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.

2.2.5 Types of Neural Networks

After knowing the basic terms about NN training and evaluation, different types of NN are discussed. Generally, neural networks can be classified into different categories (*e.g.,* structure, data flow, neurones, layer, or density). Therefore, this chapter focusses on fundamental networks with regard to their main characteristics. As mentioned in Chapter 2.2.3, ANN is a general neural network, which can cover several architectures (*e.g.,* CNN, RNN, etc.). However, due to the large number of different networks, the focus is on the most common [34].

Deep neural networks

DL involves neural networks with layers that extract a hierarchy of features from the raw input images. From an architectural perspective, these neural networks typically consist of different layers (*e.g.,* input, hidden, output) with neurones, activation functions, and weights. The depth of the system is determined by the number of layers between the input layer and the output layer, called the hidden layers. Each hidden layer is vector valued and receives input from the previous layer, multiplies it with weights, and computes its activation value. Historically, a deep neural network (DNN) was defined as a network with a depth of >2. This definition has changed with the rise of new models to hundreds or thousands of layers. Therefore, the larger the depth of DNN, the greater the representation capacity within the network. Although its great to increase the size of the neural network, its not always necessary and also comes with some downsides. These are mainly based on increased computational resources and the aspect that performance cannot be further increased. In general, all the network architectures mentioned in the following (*i.e.,* MLP, CNN or RNN) can be considered as DNN. If they are also called DNN depends on the depth of their networks. It is worth mentioning that a special layer within the neural network is the so-called fully connected or dense layer, where the neurones in a layer are connected to all neurones in the previous layer. The drawback of this special network is that the number of parameters increases rapidly, increasing computational costs [34].

Feed Forward Neural Network

Feedforward neural networks are one of the most basic networks, since the information flow is one-directional from input x through computations to output $f(x)$. This feedforward neural networks can be either single-layered or multi-layered. A special multilayer feedforward network NN is a multilayer perceptron (MLP). It consists of at least three layers (*i.e.,* input layer, hidden layer, and output layer) with an additional activation function. Generally, a feedforward NN is called MLP if it consists of fully connected layers or a specific activation function. Furthermore, the output of such NNs can be a function $f(x)$, which can be

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.

composed of a chain of functions (see Equation 2.5) presented by the depth of the model n (*i.e.,* number of layers in the network).

$$f(x) = f^{(n)}(f^{(n-1)}...(f^2(f^1))) \tag{2.5}$$

The general goal of the algorithm is that $f(x)$ approximates $f(x)$ *, which is the expected function. Therefore, during training, x must produce a value to minimise the deviation of $f(x)$ * from $f(x)$. For these networks, all weights must be selected initially and are static for the forward path. Feedforward neural networks can be best seen as function approximation machines to achieve generalisation. Compared to a linear model, the most significant difference of this neural network is that it is trained iteratively by an optimiser to the best approximation rather than solving the equation. This type of NN focusses on how the network is capable of learning instead of changing the architecture. Another type of NN that focusses on learning behaviour is recurrent neural networks. In general, feedforward networks can be used for simple classification tasks or face recognition. Their benefit lies in the fast and easy identification in one direction. The feed forward neural network is the simplest network in terms of how information is processed through a network [24, 34].

Convolutional Neural Networks

The convolutional neural network (CNN) is type of a NN with automatic feature extraction and outstanding performance in processing images, speech or audio. Today, most convolutional networks are trained in a purely supervised fashion, using feed forward and back-propagation through the entire network on each training iteration. Figure 2.9 shows the basic structure of a CNN, which contains one or more convolutional layers followed by pooling layers, where the output of these layers goes to a fully connected layer to classify the image. These convolution, pooling, and activation procedures can be repeated until multiple layers are represented within the network. The input of CNN is handled batch-wise and consists of an image, typically transferred to an array with rows, columns, and colour channels, where each entry represents one pixel. Afterward, the convolution process slices the input image into different feature maps and the pooling process downsamples the size of the convolution layer by using a pooling filter, thus increasing the invariance of the system. However, these two procedures of NN generally require fewer parameters than a fully connected layer. While the number of parameters describes the complexity of the network, the number of layers describes the depth of a NN. Each of the layers represents a different output of feature maps. Usually, edges and corners are detected in the first step, the complexity of the features will increase with higher depth and eventually transform into abstract formations. In the last step, a fully connected layer is fed, where each input represents an abstract feature of the initial image. The output of the fully connected layer should be a confidence score that

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.
TU Bibliothek Your knowledge hub WIEN

predicts every class label. Depending on the activation of the different features, a forecast can be given of what the input image tries to present. Moreover, the results may not lead to a unique identification and the boundaries might be blurred. Therefore, adequate algorithm training is required to improve its classification performance. In general, the significant benefit of CNN is the unnecessary extraction of manual characteristics, while its basic principle relies on weight sharing between layers. The key insight that justifies CNN for image classification is that we do not need complete information from all input data simultaneously, but rather have information only at the local level [12, 20].

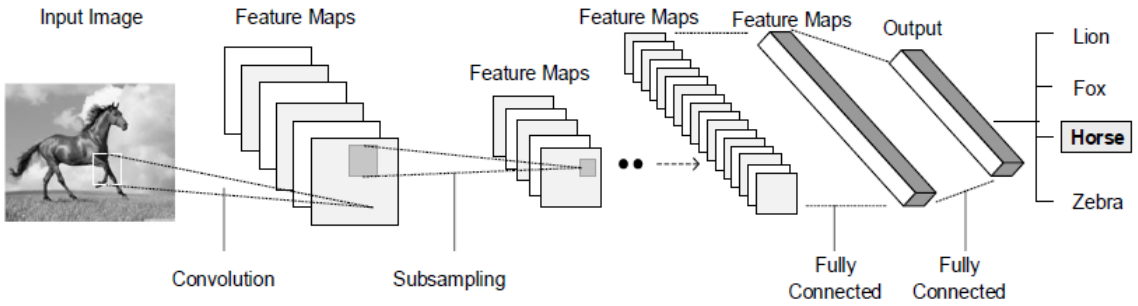


Figure 2.9: Schematic representation of a CNN for image classification. The input image is passed through convolutional and pooling layers into a fully connected layer for classification [12].

Filters and Features The convolution process concentrates on the proper extraction of image characteristics or patterns. Therefore, a processing technique (*i.e.,* filtering) is required that determines specific patterns within an input (*i.e.,* pixels of an image) and changes its values without changing the position. These patterns are striking characteristics of the image. Usually, every characteristic of the input image that supports the algorithm to detect the appropriate solutions is a feature. These features are detected by the weights of the filter, which alters the structure of the feature. Therefore, element-wise multiplication of an input array with an array of weights is referred to as a filter or kernel, which is the detector of features. For example, the detection of vertical lines (*i.e.,* features) in an input image can be accomplished by a 3x3 filter (*i.e.,* kernel size), where each filter entry illustrates one weight (see Fig. 2.10a). Another low-level feature extractor is the 3x3 filter for corner detection (see Fig. 2.10b) [26, 37].

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.

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

(a) (b)

Figure 2.10: (a) Schematic illustration of a 3x3 filter for vertical line detection within an input image [26]. (b) Schematic illustration of a 3x3 filter for edge detection within an input image [26].

These filters are usually a fraction of the size of the input image and can be drafted manually or automatically by the algorithm. In CNN, the filters are automatically chosen to find the most suitable features. Therefore, the filters are designed to systematically scan the input image for features according to a pre-selected scanning procedure. The repreated scanning on one input allows one to extract all important characteristics to create the feature map (*i.e.,* activation map). In summary, an input such as the pixels of an image is multiplied by a filter to create a feature map (see Fig. 2.11). When a stack of filters is applied to a layer, multiple feature maps can be extracted. However, the application of such a filter often comes with the so-called border effects and occurs depending on the filter scanning procedure. This effect reduces the size of the feature map compared to the input and can be a big deal for small images with large filters due to the fast loss of information. Suppose that every pixel of the input image once has the opportunity to be in the centre of the filter. Scanning of the input is no longer possible without moving beyond the boundaries of the input image, resulting in the border effect. Therefore, the extension of the area (*i.e.,* addition of pixels outside the image boundaries) in which the kernel works allows for a scanning strategy, which is called padding (*i.e.,* hyperparameter). Without padding, the corner pixels would be examined only once, leading to a reduced output size after scanning. A second way to manipulate the scanning procedure is to change the movement of the filter within the input image (*i.e.,* stride). This allows filter overlap when processing an input image, which consequently changes the output size. While padding is used to treat every pixel in the input equally, stride allows one to manipulate the movement of the filter and the size of the feature map. For example, a stride of 2 is used to half the input size [26, 37].

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.

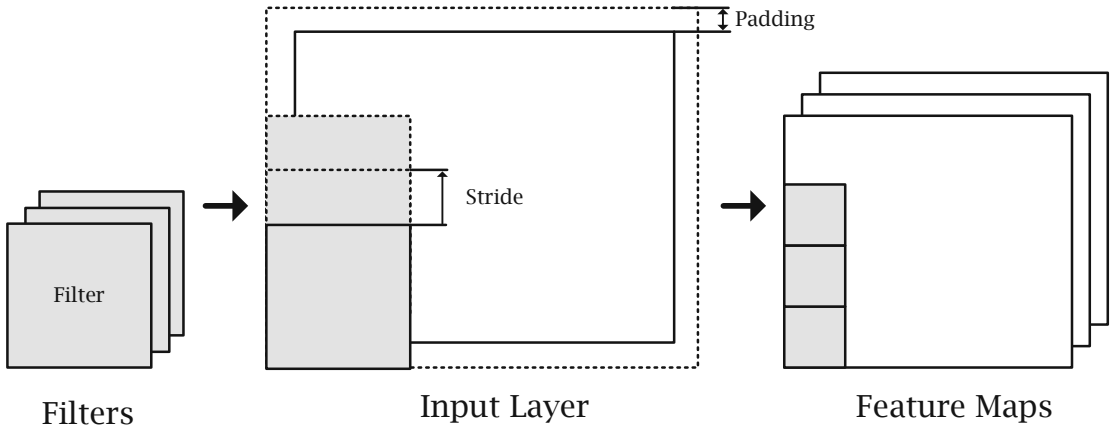


Figure 2.11: Schematic representation of filters and feature maps with stride and pooling. The illustration is adapted from [26].

Convolution Convolution originates from the idea that fully connected layers connect every value from one layer to the next, which might not be required to extract meaningful features. The term convolution describes the connection of two functions that overlap from different layers. Mathematically, convolution is a linear operation that multiplies some weights (*i.e.,* filter) with input, similar to the traditional NN. However, convolution was designed to elementally multiply a two-dimensional array of inputs (*e.g.,* image pixels) with a two-dimensional array of weights, often referred to as a filter or kernel. Therefore, convolution can be described as applying a filter to an input that results in the activation of pixels. The filter's size is usually smaller than the input, allowing it to extract details. Each filter is a channel of the convolutional layer and processes only one characteristic of the layer. Although the depper layers can see the entire image (*i.e.,* global features), the early layers learn more local features. The scanning of the entire input of the layer with different filters results in feature maps (*i.e.,* activation maps), of multiple features. In other words, instead of increasing the layer count to learn more features, the filters are replicated within a layer using a depth size D (*i.e.,* hyperparameter), which is known as a feature map. While traditional ML approaches rely on hand-crafted filters, convolution automatically detects features anywhere in the input image. This feature detection capability is based on stochastic gradient descent training, where the network is forced to extract features that minimise the loss for the specific task at most [24, 26].

Pooling As already mentioned, by implementing convolutional layers, the positions of features can be precisely detected within an input image. However, convolution performs poorly if the location of the feature is altered (*e.g.,* rotation of the image), leading to a different feature. Therefore, pooling addresses this problem by downsampling feature maps that still contain critical structural elements of the feature without fine details. This pooling layer

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.

is accomplished similarly to the convolutional layer and increases the system's robustness. Downsampling is achieved by summarising the feature maps into patches and applying a pooling layer. Alternatively, this can be accomplished by stride by changing the step size of the filter during convolution. However, the more robust way is to use a pooling layer after applying the nonlinearity function of the feature layer. This pooling layer can be generated by a small filter (*e.g.,* 2x2 pixels) with a specific stride (*e.g.,* 2 pixels), which consequently reduces the output of the pooling layer by a factor (*i.e.,* 2) compared to the feature map. The variation of the pooling filter and the stride allows generating different outcome sizes. The two main pooling procedures are average pooling and maximum pooling. While maximum pooling extracts the maximum pixel value for each filter, average pooling calculates the average value. Therefore, pooling combines all pixel values of one filter into one value. This new capability of pooling layers is referred to as model invariance to local translation, where a translation of the feature in the image will not change the pooled output. Generally, maximum pooling has overtaken average pooling, as it is more valuable to highlight the most present feature in the filter compared to the average value. In summary, the main difference in convolution compared to pooling is that each feature map is processed by a different filter and pooling applies the same procedure to every feature [24, 37].

Recurrent Neural Networks

If MLPs are extended to include feedback loops and have the ability to save state information $f(x)_{state}$, they are called recurrent neural networks (RNNs). These feedback loops can be generated by processing the previous output $f(x)_{state-t}$ after a specific time step t. Therefore, RNNs consists of a structure with multiple states and can share the parameter weights between those. Such networks usually save a part of the layer output in memory, which is further processed with the actual input from the next state. However, the complexity of network training increases with the captured time span and the dependencies of its states. A recurrent neural network is especially used for tasks that require the previous output to detect coherence. In such cases, the data is primarily dependent on historical information. Applications include text processing (*e.g.,* grammar checks) or text-to-speech processing. The challenge of RNNs is to train networks over a long period of time to capture the desired connections. This can be handled using the strategy of Long Short-Term Memory (LSTM). LSTM is a subtype of RNN, including a memory for state saving, and addresses the problems of RNN by limiting the information flow through time steps through gates. Typically, an input gate controls the input from the last layer and an output gate controls the output from the next layer. Its strength lies in efficiently processing time-series data that might have some time gaps (*e.g.,* losses within video frames) [15, 34].

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.
TU Bibliothek Your knowledge hub WIEN

Chapter 3

State of the Art

The daily tasks of human workers in factories are often supported by assistance systems. Since the supportive behaviour of these assistance systems might generate additional attraction for human workers, a system that can adapt and reduce the cognitive load on the human worker is desired. To enable this, the system can be context aware by capturing the environment of the human worker. One way of developing systems aware of the environment is to use image recognition. The captured images can be fed into an algorithm, which is a combination of data mining and image processing. While data mining attempts to answer the question of how image characteristics are extracted, image processing is focused on the preparation of images for the algorithm input. Image processing reshapes the resolution of the extraced images to fit the algorithm demand. Although improved image resolution might help increase the overall performance of the algorithm, it negatively affects latency due to the higher processing time. To effectively use data mining, the aspects of handling large amounts of image data and low latency for application execution must be considered. To fulfil the first aspect of handling large amounts of image data, the assistance system must be able to dynamically extract the context of the environment. To enable dynamic working environments, the overall latency of the algorithm has to be in the range of some milliseconds. This allows one to feed back the processed input to the user at an appropriate time. Image classification is an application which can fulfil the aspects of data mining. The term image classification can be understood as the classification of images into particular categories of information. However, the image classification algorithm also underlies some essential requirements. To enable low latency, training the image classifier using historical data is important. This historical data can be provided by images that illustrate the characteristics of the different workplace categories. Lastly, the classifier must fit the application case and perform the classification with a high degree of accuracy. Therefore, image classification can be used in the field of Image Mining within the Smart Factory and can provide useful information about the content of an image, especially in the human working environment [39].

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.
TU Bibliothek Your knowledge hub WIEN

Today, convolutional neural networks (CNN) are the state-of-the-art approach to solve problems in the field of images. A CNN has the ability to handle zoom or translation invariance within the image, due to its special layers (*i.e.,* convolution and pooling). Moreover, CNN can reduce the parameters required due to its special layers, compared to a fully connected network, which consequently benefits the fluent real-time application. Usually, during the image classification approach, the entire image is observed. Therefore, the most efficient characteristics can be extracted in each layer of the NN, starting from the corners and edges of the first layer and increasing to higher-order characteristics for the deeper layers. These identified patterns and structures of the image are used to derive semantic information. A special-case image classification approach is scene classification, where the region of interest is both foreground and background. It is a semantically coherent view of a real-world environment, including multiple objects. A scene content can exhibit different unique identifiers, which include objects inside the image, the locations and relations between objects, the surroundings of the objects (*i.e.,* background) and even events inside an object, which describe an activity inside a specific environment. The goal of scene classification is to classify an input (*i.e.,* images) based on its ambient content, objects, and layout into a countable number of classes. To understand what the most promising software for this use case looks like, a rough idea of how the latest deep learning technology can be applied is required [30, 40].

As discussed in Chapter 2.2.2, supervised learning algorithms need labelled input data, which allows to train a NN. Since the use case application should be made for the specific environment of an assembly factory, our network requires labelled data (*i.e.,* images) with this specific information. Therefore, state-of-the-art research is conducted to find the latest datasets that contain basic structures and objects. Most likely, these datasets will not contain all the context required to solve the desired classification. However, these dataset can be used as a good starting point for transfer learning to pretrain the model on low-level image features. By choosing a dataset with similar scenes to our use case, transfer learning those scenes allows extracting image characteristics. By learning image features through transfer learning of existing datasets, high-level features from the assembly factory environment will not be learnt well enough. Developing our own dataset, which represents these high-level features, is necessary to cover the full context of the use case. However, the appropriate choice of the dataset is not only important for our own application. Moreover, the choice of the model architecture is a crucial part for providing a solution that meets the requirements of the use case. It must be complex enough to extract details and simple enough to operate at low computational costs [40].

To find the most promising datasets and model frameworks, state-of-the-art research is carried out. Therefore, datasets are examined that include similar scenes for the working environment of our application. In addition, learning strategies, hyperparameters, and network architectures of other applications are examined to find features and gain insight for

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.
TU Bibliothek Your knowledge hub WIEN

the model construction of our own application. To fully understand the importance of these specific attributes, the use case and its operational goal will be described in the experimental part. In general, the state-of-the-art chapter covers dataset research that is used to extract low-level features, important aspects in generating a dataset for deep learning, and different model characteristics and architectures.

3.1 Datasets for Scene Recognition

Before fitting and optimising the network to its specific needs, the foundation must be built carefully. Without a suitable and high-quality dataset, training a neural network can become a serious challenge. Since the use case application should be made for an assembly factory, suitable images must be gathered that describe that context. Furthermore, the specific characteristics and boundaries of the human working environment must be defined to limit the operationability of the application to certain constraints. Therefore, a boundary of our use case is that all assembly images are located indoors, which limits the dataset to indoor images. By indoor scenes, we assume that the assembly is principally within a factory building where the full sky is not visible except for windows. In general, classification of indoor scenes is a challenging topic and can often not meet the accuracy of an outdoor scene classification task, due to many reasons. Human-made buildings and objects often show higher similarity than natural objects and require more effort from the algorithm. Furthermore, the focus on indoor scenes impedes classification due to poor and artificial lighting conditions. Additionally, since workplace predictions should be made dynamically, the dataset should contain any random scenes within the factory. Finally, these real-time predictions are based on data supplied in time sequences or videos, which capture scenes with temporal alterations [40].

3.1.1 Challenges in Datasets for Scene Recognition

Although the boundary conditions within the dataset can mitigate the challenge of scene classification, there are still other difficulties that exist. The two main ones in operating with scenes are annotation ambiguity and visual inconsistency, which are concluded from the nature of the scene context in the dataset. The first, visual inconsistency (see Fig. 3.1a) states that the same scene category shows significant differences in appearances, and the second, annotation ambiguity (see Fig. 3.1b) refers to different scene categories that have similar objects or appearances. Both challenges are owed by the human that generates the dataset. Generally, the annotator must rely on his intuition and knowledge to label training data with scenes consisting of varying intrinsic factors (*i.e.,* objects, background, or human activities). Therefore, scene categories are subjective and do not occupy a categorical distinction [40].

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.
TU Bibliothek Your knowledge hub WIEN

Visual inconsistency describes a high intraclass variance of a scene category which is based on low to no similarity of features between images. This significant difference in appearance within a scene category is caused by the subjective processes of labelling images into various categories. Furthermore, the annotation of images is impeded by different image conditions (e.g., shading, blur motion, poor resolution, filtering distortion, etc.) that increase the effects of intraclass variance. Therefore, large differences within the context of two scenes can occur, making it nearly impossible to visually connect both and classify them into the same category. On the other hand, annotation ambiguity goes along with a low inter-class variance, where the same visual occurrences in different categories result in class overlaps. If large amounts of specific features are represented in multiple classes, confusion or uncertainty for the classifier is the result. A clear prediction of a class might no longer be possible, which will cause random behaviour during classification. This is especially the case if a dataset consists of a large number of categories. A probable solution to this behaviour is to assign multiple ground-truth labels to one category. The ground-truth labeling can be done by applying an existing classification algorithm, trained on a working dataset, on the ambigous dataset. Based on this information, super classes can be built which combine similar classes to achieve higher classifier precision [31, 40].

(a)

(b)

Figure 3.1: Illustration of the two main challenges of scene datasets. (a) Visual inconsistency due to the same scene category (*i.e.,* mall) but with images of different content. (b) Annotation ambiguity due to different scene categories (*i.e.,* library, archive, and bockstore) but with similar content information within categories [31].

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.

3.1.2 Publicly Available Datasets

Most of the publicly available datasets focus on object categories, which provide labelled data in numerous ways (*i.e.,* bounding boxes, segmentation, etc.). Although these datasets were built for object recognition, transfer learning of these datasets can be a valuable choice to pretrain a network for scene recognition. Pretraining the model parameters on datasets is especially useful when the available dataset is small. Without any pretraining, the weights of the model are initialised randomly, making the training of a small dataset more challenging. Although the entire scene content is relevant for classification, training on an object-based dataset is feasible, since the actual use case scenes might have similar or the same objects within them. Therefore, the extracted features of the objects are partially inherent within the actual scenes, making it a good choice to get a first impression. Moreover, since the dataset for the use case was defined on the basis of characteristic objects within the working environment, learning these or similar objects can be beneficial. However, since multiple objects are possible to appear in one record of the human worker, the image might not be solely classified on one object, since the entire scene plays a significant role. In the following, relevant datasets that contain important objects or scenes for transfer learning are descirbed [31, 40].

ImageNet is a object-centric dataset which was built on the hierarchical structure provided by objects defined by *WordNet*. These objects are organised in different classes within a root-to-leaf structure with large subtrees (*e.g.,* vehicles) unfolding into synsets (*e.g.,* sailboat). The systematic way of organising data allows *ImageNet* to be more accurate and diverse on a larger scale than other datasets. *ImageNet* currently contains 14.197.122 images[1] with 21841 indexed synsets[2], where each synset consists of 500-1000 images [41].

The scene-centric dataset *MIT Indoor 67* (*i.e.,* MIT67), is focused on indoor scenes such as stores, buildings, or public spaces. It consists of 15.620 images[3] from 67 indoor scenes[4], where each image has a minimum resolution of 200 pixels [31] on the smallest axis. The dataset authors Quattoni and Torralba [42] state that most outdoor scenes can be categorised by global image properties, while for indoor scenes characterising objects might be more preferential. In addition, they state that local and global image information is important for the recognition of indoor scenes [31].

Another dataset is the *Scene UNderstanding 397* (*i.e., SUN397*) dataset, which comprises 397 categories [43] sampled, including 175 indoor scenes [43]. The dataset contains 108.754 images, where each scene has a minimum size of 200x200 pixels [43]. The *SUN* dataset categories were built similarly to *ImageNet* on the basis of the *WordNet* approach, which

[1] https://www.image-net.org/, last accessed on 20.08.2022
[2] Ibid.
[3] https://web.mit.edu/torralba/www/indoor.html, last accessed on 20.08.2022
[4] Ibid.

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.
TU Bibliothek Your knowledge hub WIEN

includes 70.000 words [43] to describe scenes. After preprocessing (*e.g.,* grouping synonyms or separating classes described by the same words, etc.), 397 categories [43] with unique identities that have a minimum of 100 images [43] were remaining. Therefore, the preprocessed *SUN* dataset is often referred to as *SUN397* dataset. This dataset is motivated by the authors, Xiao et al. [43], who proposed that the current scene dataset tasks fail to capture the richness and diversity of the scenes. They believe that recognising the types of scene within an image is more valuable than labelling an entire image with a scene. This means that one image is represented by a combination of multiple scenes, similar to real-world applications. Furthermore, they state that human performance is not necessarily the upper limit of classification, as computational methods could reach higher accuracy in some categories [43].

Zeng et al. [31] state that publicly available datasets, without semantic labelling, are not large and rich enough to cover the high diversity of environmental scenes. Additionally, the scene categories are often unbalanced, and the large variety of images in the classes results in a poor prediction of infrequent images within a class. Scene classification often lacks closed-world assumptions and is not robust enough to be used in different applications. Therefore, datasets are required that include segmental annotation information. The *LabelMe* dataset includes, within the training set, 1000 fully annotated images[5], and around 2000 partially annotated images[6] with 32.164 labelled objects[7]. Figure 3.2 shows an annotated image from the dataset *LabelMe*, where the lines indicate the segmented objects. However, for our proposed method within the use case, the scenes have a semantic multiplicity, where a scene belongs to multiple semantic classes. Therefore, multi-label images that include multiple classes are required, or only the most obvious class is assigned [31].

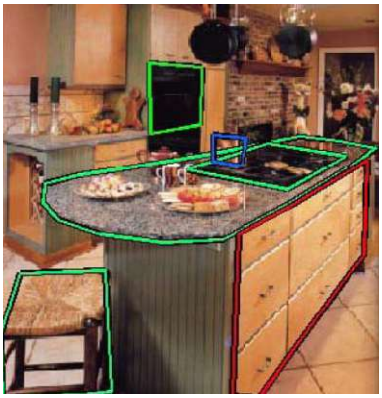


Figure 3.2: Visualization of the annotation of a kitchen image from the *LabelMe* dataset. The lines around the scene content represent segmented objects (*e.g.,* barstool, etc.) [44].

[5]http://labelme.csail.mit.edu/, last accessed on 10.09.2022
[6]Ibid.
[7]Ibid.

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.
TU Bibliothek Your knowledge hub WIEN

Zhou et al. [45] claim that, in general, scene recognition does not achieve the same performance as object detection, trained on *ImageNet*. Datasets, like *MIT Indoor67* and *SUN397* failed in terms of quantity to feed deep learning algorithms. Therefore, they introduced the *Places365* dataset with 10 million labelled images[8] of scenes. *Places365* is a scene-centric dataset, which was built to complement large object-centric datasets such as *ImageNet*. A dataset of this size ensures that CNN training becomes feasible and that the diversity of each class can be covered. It includes 434 scenes[9] that account for 98% [46] of scenes a human can encounter in the natural and human-made world. These scenes were selected with some minor changes from the *SUN397* database, which was based on *WordNet*. However, it is worth mentioning that each class of the dataset includes 15.000 to 30.000 training images [46], each image with a minimum size of 256x256 pixels [46]. To extract the richness of features from such a large dataset, the model architecture must be complex enough. If not, the large size of the dataset will not be beneficial and will not increase the robustness or performance of the system. Furthermore, Figure 3.3 is an illustration of possible categories show similar features to the environment [46].

(a) Archive (b) Assembly Line (c) Basement (d) Cublicle Office

(e) Dining Hall (f) Garage Indoor (g) Storage Room (h) Repair Shop

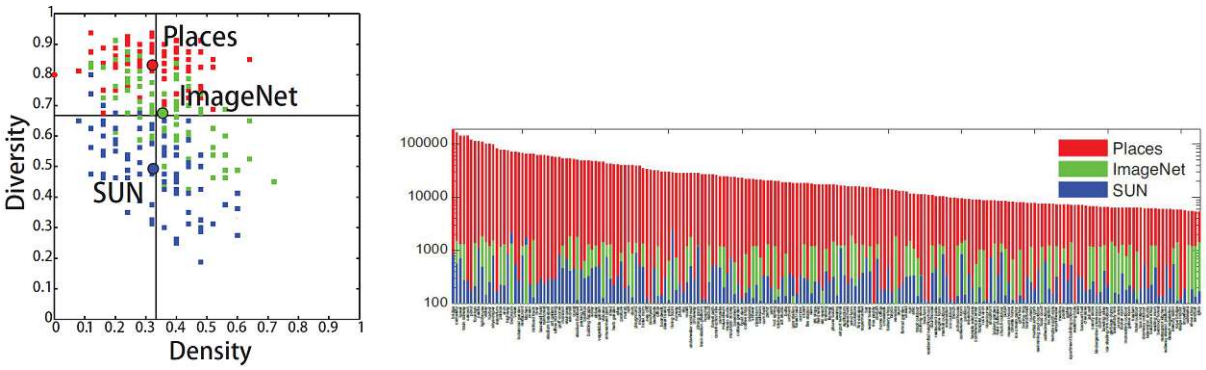
(i) Phys. Laboratory (j) Pantry (k) Office (l) Home Office

Figure 3.3: Illustration of relevant categories from the *Places365* dataset for pretraining the model.[10]

[8] http://places2.csail.mit.edu/, last accessed on 20.10.2022
[9] Ibid.
[10] http://places2.csail.mit.edu/explore.html, last accessed on 20.11.2022

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.
TU Bibliothek Your knowledge hub WIEN

Furthermore, the authors state that the Places365 dataset is more diverse and dense than any other scene-centric datasets. Diversity and density are important to validate the performance and robustness of the algorithm. If a dataset is not diverse and dense, its generalisation capability is low. While high density signifies that images generally have a high amount of similar neighbours, high diversity describes great richness within the scenes. Moreover, the authors show that there is a difference between the features learnt with an object-centric approach and a scene-centric approach CNN. Scene-centric CNNs have the capability to recognise more richness inherent in the scenes. During building their dataset they realised that datasets which include the same visual classes yet have different generalisation performance. However, the comparison between three mentioned datasets *Places*, *SUN* and *ImageNet* illustrates that the *Places* dataset is more diverse and dense than the other three datasets (see Fig. 3.4a). The *Places* dataset has the best density and diversity, and also has the highest number of scenes per category (see Fig. 3.4b) [45, 46].



(a) (b)

Figure 3.4: (a) Comparison of the density and diversity of the datasets *Places*, *SUN*, and *ImageNet* [45]. The *Places* dataset achieves the highest diversity at an equally good density. (b) Comparison of the number of images per category of each dataset. The *Places* dataset has the highest number of images [45].

Due to the limitation of available images for various application areas, researchers have also tested dynamic scene data by extracting images from videos. Furthermore, research has been conducted to directly apply dynamic scene data to an algorithm. The challenge in generating dynamic scene data lies in separating camera motion from inherent movement within the scene. Considering a camera motion as an addition to the movement of activities within the image, algorithms become uncertain about detecting meaningful features. However, describing an image is not necessarily done using deep learning techniques. Researchers have also tried a chaotic system framework for classification that does not follow any sequential order. In conclusion, dynamic scene data is an upcoming field of research with some minor drawbacks, which currently lacks qualitative data [40].

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.
TU Bibliothek Your knowledge hub WIEN

3.2 Model Frameworks for Scene Recognition

Since the choice of the right deep learning network is motivated by the complexity of the application, the architecture should be chosen to best suit the needs of the use case. In general, there are two different temporal differentiations for scene recognition, static scene recognition, and dynamic scene recognition. Static scene recognition focusses on classifying scenes based on spatial information within the scenes, without considering any temporal changes. It can be done by either a scene-centric or object-centric approach, where either the scene or the object is in the centre of recognition. On the other hand, dynamic scene recognition addresses spatial networks by additionally considering the temporal aspects of the scenes. These temporal changes are mainly based on analysing the change of scenes between different states. In this chapter, a basic introduction to static scene recognition is given, and further network architectures or characteristics of models are analysed [31, 40].

3.2.1 Static Scene Recognition

Static scene recognition is treated as a method of scene classification in which the input of the model exhibits scenes in different states. In a high-level approach, there are two main detection strategies for static scene recognition. Scene recognition based on object-centric or scene-centric approaches. For scene-centric approaches, the classifier dives into the surrounding area of the object and tries to classify the image based on the entire content. Therefore, further semantic labelling might be necessary to identify the connection between semantic parts (e.g., objects, textures, or background) within the scenes. In scene recognition, if the input images do not include semantic labelling, the model must learn contextual features from images with high-level convolutional layers. Therefore, CNN is expected to learn the deep features presented in the image. Although CNN has been trained on a dataset without any semantic labelling, the network still identifies semantic clues in the image by detecting objects alongside contextual clues. However, we want to detect the entire context of a scene without loss of information. A technique which allows detection of the entire image and not just the object is of interest. The concept of a scene-centric approach is to understand the content of the image holistically, by finding low-level features or segmented regions to conclude on the image context. In contrast, the object-centric approach is the procedure to classify a scene based on the response of an image to multiple object detectors (*i.e.,* object bank). Therefore, the objects are identified within the scene and the appropriate scene based on the inherent objects is determined. Undoubtedly, this approach is useful for describing multiple significant objects inside scenes and capturing high-level visual representations. For indoor scene classification, it might not be enough to extract just the objects, since poor performance can occur if the same objects are presented in different scene categories. Thus, the classification performance of scene-centric CNNs is expected to be better than that of

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.
TU Bibliothek Your knowledge hub WIEN

object-centric CNNs, since for indoor scenes features in different scales of the image can be extracted. In addition, they may extract more detailed information from a scene, such as fine local semantic regions, which are crucial to discriminate ambiguous scenes [31].

Due to the lack of sufficient dataset sizes in scene recognition, the latest research has focused on dynamic scene recognition using videos as input. To directly process the videos without any frame extraction, dynamic scene recognition captures, in addition to the extracted spatial features from static scenes, temporal information that connects these static scenes. The challenge of dynamic scene recognition lies in creating a model that is powerful enough to capture spatial and temporal information about the scene. RNN are the perfect choice to solve these tasks, especially LSTMs with the capability of short-term and long-term memory. Another applied architecture is T-ResNet which is based on a residual network. The T-ResNet model shows strong performance for objects with linear motion but has difficulties in mixed motion patterns (*i.e.,* intrinsic scene dynamics and camera motion). However, it is possible to use static scene recognition in a dynamic way by extracting image sequences from a video. For example, every second an image can be extracted from a video and classified into a category. This technique is less computationally intensive and can fit the need for an application area in which qualitative datasets are available [40, 47].

3.2.2 Network Architectures of Scene Recognition

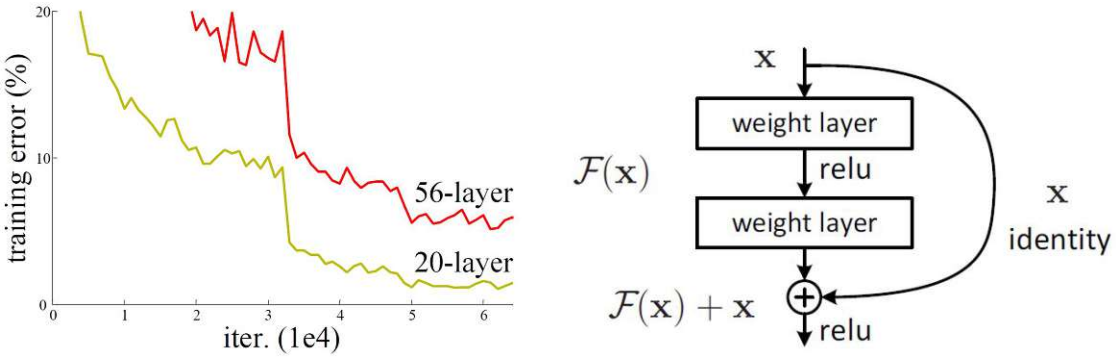
CNN Models for Scene Recognition

AlexNet is a neural network architecture, designed by Krizhevsky et al. [35], which shows that model depth is important for performance. However, this performance gain comes at the cost of more model parameters and longer processing time. Jmour et al. [48] have used a scene-centric strategy, where neural network training was accelerated by using pretrained model parameters to classify traffic sign images into 4 different categories. They have implemented this system using the well-known AlexNet architecture combined with some transfer learning techniques. Transfer learning is important, to initalize weights for training and consequently improve network performance. Without targeted weight initialisation, gradient descent might not reach a local optimal minima for these nonconvex functions. After pretraining the initial parameters of their model on a different dataset, all layers beside the last one were frozen. Freezing keeps the parameters within the frozen layer constant, without any update by further training. Finally, the parameters inside the last layer are trained based on a more specific dataset consisting of 360 traffic sign images [48]. For the traffic sign classification approach, the authors have remarkably achieved a 93% accuracy [48] for the test set. However, the well-known AlexNet architecture is not state-of-the-art anymore due to various reasons (*e.g.,* kernel size too large, low amount of layers, etc.), and new architectures have replaced its position [31].

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.
TU Bibliothek Your knowledge hub WIEN

Therefore, the authors Xiao et al. [49] stated that the original AlexNet model is outdated due to reasons such as large convolutional kernels or high stride in the first layer. During training, this large convolutional kernel and stride lead to a rapid decline of the input shape after the first layer, which corresponds to a loss of spatial information. However, the authors suggested decomposing the large convolutional kernel into a cascade structure with small convolutional kernels and a reduced stride. Since scene images have a rich foreground and background, a smaller convolutional kernel can recognise more fine-grained local features to distinguish between the targets within the scene. After changing the old-fashioned AlexNet model with some new design features, their model improvements on different datasets could be verified. Finally, they have tested their improved AlexNet model in 23 categories [49] from the *Places 2* dataset with 22650 total images [49] and achieved with 72% almost 10% [49] accuracy improvement compared to its unchanged version [49].

Deep Residual Network The choice of the right depth of a network is always an important decision. When choosing a network that is too shallow, the network might not be able to learn all the desired complex tasks. As learnt by the authors [31, 49], one can think that choosing a network that is deep enough could solve this problem. But in the early stages of neural networks, training a deeper architecture (*i.e.,* more layers) was not always beneficial. During training, such networks stopped improving with a higher training error than its shallower counterpart (*i.e.,* degradation problem) (see Fig. 3.5a). This degradation is not caused by overfitting, as the deeper model is stuck with a higher training error. Instead, it results from the initialisation of the network, the optimisation function, or the backpropagation algorithm (*i.e.,* vanishing or exploding gradients) [50].



(a) (b)

Figure 3.5: (a) Difference in training error between a 20-layer and 56-layer plain neural network over the number of iterations. The higher training error is notable for deeper NN [50]. (b) Schematic representation of a residual block of a residual neural network, where identity mapping ensures that the deeper layer does not perform worse [50].

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.

Generally, in NN the deeper nodes have a wide diversified connection to the nodes in the early layers. If these deeper nodes lose information during backpropagation (*i.e.,* the gradient approaches 0), the information cannot be passed to the earlier layers and the training improvement might stop. In contrast, if the gradients explode, training of a neural network becomes more challenging due to larger weight updates. However, as *CNN* depth increases, weight updates for the model make the training procedure unstable, increasing the risk that the gradient falls or explodes. This problem has been encouraged by the architectures of neural networks. For example, normalisation layers within the network keep the model weights low for computational purposes, but increase the chances of a vanishing gradient [31].

To address this issue, residual networks (*i.e.,* ResNet) can skip the hierarchical structure of CNNs by allowing the gradient information to be passed through layers without computation. Instead of hoping that each layer reaches a lower training error, as in traditional approaches, it is explicitly desired that each layer fits a residual mapping (see Fig. 3.5b). The residual mapping is formulated by $F(x)+x$, which can be carried out by feedforward neural networks with shortcut connections. These shortcut connections are responsible for skipping one or more layers during one training phase. Layers with one shortcut connection are specified as the residual block (*i.e.,* stack of layers). These residual blocks make it possible to learn identity functions. Therefore, the identity mappings replicate the input of the residual blocks to its output, which ensures that the higher layers do not perform worse than the lower layers. The ResNet architecture, consisting of multiple residual blocks, avoids signal loss by vanishing gradients with the help of skip connections, consequently increasing the accuracy of deep networks by optimising the training procedure [50].

The authors He et al. [50] tested this strategy by building a residual network on a plain network, which was inspired by the *VGG-19* architecture. Their residual network has a lower complexity than the *VGG-19* network, its parameter size is only 18% [50] of it. This plain network consists of 34 layers with 3x3 convolutional filters, which were upgraded with shortcut connections to its residual version. The input images of the network were a random region of 224x224 [50], which were cropped from the resized 256x480 images [50]. Furthermore, the network includes batch normalisation layers, SGD as the optimisation function, a batch size of 256 [50], an initial learning rate of 0.1 [50] that decays at 32.000 iterations [50] by 10. Moreover, the model is trained for 60.000 iterations [50], with a weight decay of 0.0001 [50] and a momentum of 0.9 [50].

Researchers A. Shah and K. Rana [51], focused on the classification of indoor scenes using a scene-centric deep learning technique based on residual neural networks. The classification should include different states (*i.e.,* empty, partially empty, disciplined, or indisciplined) of the class room scenes. Therefore, they developed their own dataset that contains these

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.
TU Bibliothek Your knowledge hub WIEN

state information. However, the lack of representatives within their dataset encouraged them to pretrain their model with the object-centric *ImageNet* dataset. After pretraining, they fixed the weights of some early layers to ensure that these early parameters do not change anymore, keeping the richness learnt for low-level features. As a model they have chosen a ResNet architecture (*i.e.,* 152 layers), where one residual unit consists of convolutional, batch normalisation, activation, and pool layers. The dataset that includes high-level feature information was images extracted from videos, resized to a shape of 224 x 224 pixels for the network input [51]. In the last layer, they used a softmax classifier to identify the label. As a result, their approach has achieved an accuracy of 83% [51] on a classroom scene test set [51].

Efficient Neural Networks Afif et al. [52] developed an application for five selected categories of common indoor scenery in the *MIT67* dataset. The goal of their work was to develop an indoor scene assistance system for robot service and blind people. Therefore, they have chosen a network built on the *EfficientNet* architecture that exhibits the desired inverted bottleneck structure. *EfficientNet* provides a new approach to network scaling, by uniformly balancing the depth, width, and resolution of the model. As evaluation method, they have chosen the cross-entropy loss function. They trained their network for 10.000 iterations [52] with a learning rate of 0.01 [52]. Furthermore, they selected a training and validation batch size of 100 [52]. Their system achieved around 95% accuracy [52] for a truncated *MIT67* dataset, consisting of 5 classes, and reached a processing speed of 12 ms per image, which corresponds to 83 FPS [52].

Advances Image Classification with CNNs

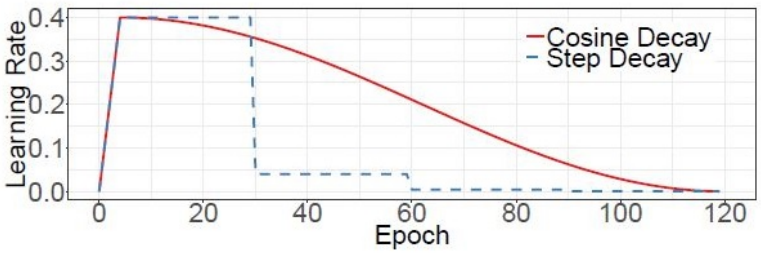
In recent years, new architectures (*i.e., VGG*, ResNet, *DenseNet*, etc.) have been proposed that steadily increased the accuracy of the model. However, these improvements in model accuracy are not the result of improved architecture alone. Small refinements in the data preprocessing strategy or training procedure, such as loss functions, learning rate schedulers, or optimisation methods, played a huge role in achieving this. Therefore, this short review lists the latest implementation details for improving the training procedure or model architecture, but barely changing the computational complexity [53].

Starting with the preprocessing strategy, a different procedure for the training and validation dataset is beneficial. Therefore, for the training dataset the following data augmentation strategies were used: randomly changing the hue, brightness, contrast, and saturation coefficients of an image between values of 0.6 and 1.4 [53]; cropping a rectangular region of the image and resizing it into a 224 by 224 square image [53]; adding noise as a destructive factor to the image; normalising the image by subtracting the mean and standard deviation of the entire dataset, for weight sharing between different datasets. On the other hand, the valida-

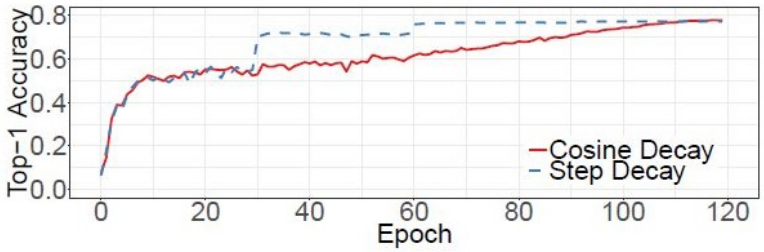
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

tion procedure does not require any special data augmentation strategy, just normalisation and image shaping to match the input size of the network. Another augmentation strategy, called mix-up, was mentioned, which overlaps two samples of different categories into a new sample [53].

For model training, weights have generally been initialised based on the Xavier algorithm and all bias parameters were set to 0. However, this weight initialisation is different depending on the type of layer (*e.g.,* batch normalisation, mean 0 and standard deviation 1). Furthermore, the authors state that training is more efficient using lower numerical precision and a larger batch size. Lowering the 32-bit floating point precision to 16-bit for the weights can increase the overall training speed by 2 to 3 times without disturbing the training process. A larger batch size (*i.e.,* 1024) reduces the noise in the gradient, allowing a higher learning rate. Generally, a larger batch size results in greater confidence in the loss evaluated. Regarding the learning rate, they mention that a too high learning rate results in numerical instability. Therefore, they recommend gradually increasing the learning rate from 0 to the desired learning rate. This strategy can be perfectly implemented by cosine learning rate decay, which further reduces the learning rate once the desired learning rate is continuously reached to zero for the total number of batches. Decreasing ensures that if a good local minima is reached, it will not be left anymore and will be further optimised. Compared to the well-known step decay, cosine decay is claimed to be more numerically stable (see Fig. 3.6) [53].



(a)



(b)

Figure 3.6: (a) Change in the learning rate over the number of epochs for cosine decay and step decay. (b) Comparison of cosine decay and step decay for Top-1 accuracy on ImageNet [53].

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.

Lastly, model tweaks are minor changes in the architecture without increasing computational complexity but with significant benefits in the model accuracy. The authors have shown some improvements for the original ResNet network (*i.e.,* 50 layers). Two tweaks appeared in the downsampling block of this architecture. When the kernel size and stride are incorrectly adjusted, information is lost during the convolution procedure. Therefore, the authors provided a solution by correctly adjusting these parameters. Another tweak to the original ResNet architecture is the 7x7 kernel within the first convolutional layer. It can be replaced by three 3x3 kernels, which are less computationally intensive. Through the improvements of these modifications, which were mainly caused by information loss, the ResNet architecture could obtain an 1% accuracy improvement, by the cost of 3% computational speed [53]. Finally, He et al. have evaluated all their improvements for the *ResNet50* architecture and achieved an improvement in the top-1 validation accuracy of 75.3% [53] to 79.29% [53] on *ImageNet*, compared to the unchanged model [53].

Selective Joint Fine-Tuning Ge and Yu [54] introduced a transfer learning method, called selective joint fine-tuning, to improve the performance of deep learning tasks with insufficient training data. Transfer learning applies the knowledge learnt in one domain to other related tasks. In general, after training the weights of a NN with the help of a source dataset, the weights are further optimised with the more specific target dataset. By applying a standard transfer learning approach on datasets with a large number of categories, this approach might lack the ability to connect meaningful features from both datasets. Therefore, the proposed method, selective joint fine-tuning, tries to simultaneously train a target learning task with a small dataset and a source learning task with large training data (see Fig. 3.7). Consequently, information is shared between those to datasets during training to increase the target training performance. This approach is based on the borrowing of samples from a large-scale labelled dataset for the source learning task. However, the source learning task does not use all of its existing training data, but only those with similar low-level characteristics compared to the target learning class. The core idea of their transfer learning approach is based on the use of a subset of similar images from the source dataset to train one target image. We can think of it as the fact that for each image in the target dataset, k nearest-neighbour training images from the source task are searched. During their selective joint fine-tuning approach, the learning rates started at 0.01 and were divided by 10 every 2400 to 5000 iterations. Most of the experiments were completed in 16.000 iterations [54].

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.

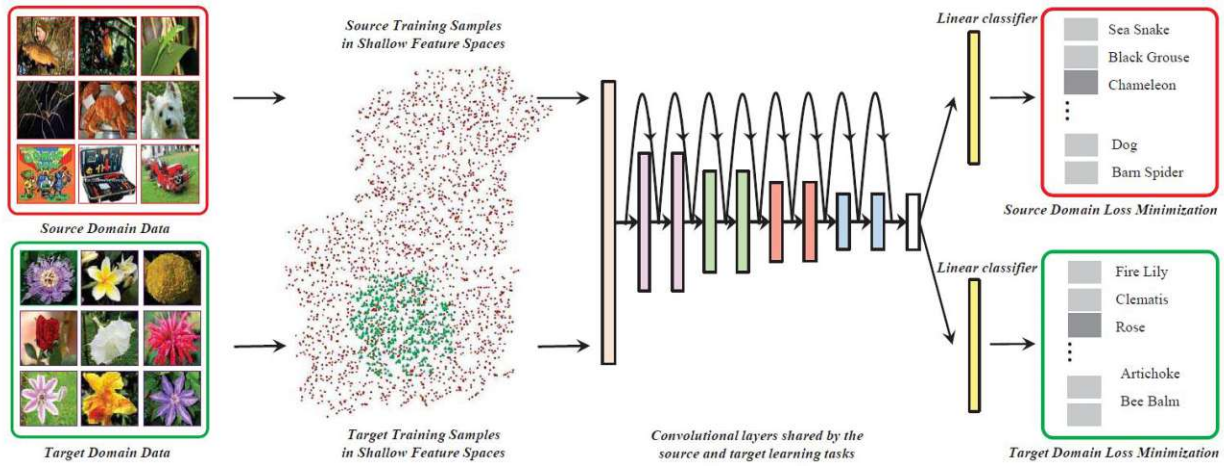


Figure 3.7: Schematic representation of selective joint fine-tuning. The convolutional layers of the NN are shared by a source and target dataset. Furthermore, the loss of the NN is simultaneously updated according to both learning tasks [54].

Hybrid Models for Scene Recognition

Multilayer feature-based method Recently, hybrid deep models have been proven to be effective methods for scene recognition. A deep hybrid model, the multilayer feature-based method, states that important slight clues (*i.e.,* information) are lost if only high-level features are extracted and fed into a classifier. Therefore, some methods use low-level features from the early layers along with high-semantic information of the features from the latest layers of hierarchical models. While the intermediate layers in CNNs can capture local features, top layers extract holistic features. This extension of extracting features from different layers allows for higher classification accuracy. However, extracting features from all layers to classify the image can result in overfitting. Therefore, usually only the features of some selected layers are extracted. For indoor indoor scene classification, intermediate and high layers (*i.e.,* parts and objects) are more important than low layers (*i.e.,* edges and textures), which capture only redundant and neglectable segments in multiple categories. Another multistage convolutional network, directed acyclic graph CNNs (DAG-CNNs), allows us to explore more complicated structural layouts. These models usually do not operate sequentially but are graph-related. DAG-CNN combine the local features of the lower layers and the holistic features of the upper layers, by connecting multiscale branches to the final classification layer [31, 55].

Semantic-Aware Scene Recognition Lopez-Cifuentes et al. [56] states that the number of parameters of conventional CNNs (*i.e., DenseNet-161*, *VGG-16*, etc.) has increased significantly compared to traditional AlexNet, although performance has only increased by a small amount (*i.e.,* 1% to 3% accuracy). Therefore, they provided a novel strategy to use object-level information within scenes to improve the training process without increasing the

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.
TU Bibliothek Your knowledge hub WIEN

number of parameters. Their approach relies on a semantic-driven attention mechanism to direct the learning process of common-scene objects. Therefore, they used a multi-modal model composed of a two-branched CNN, gathering context and image information, and an attention module. This attention module is inspired by saliency theories and only concentrates on task-relevant image content. In general, the model is a combination of a semantic segmentation branch and a feature-based branch to interpret semantic regions and exploit spatial relations. As training dataset for training *Places-365* was chosen for which the RGB images were resized to a dimension of 224 x 224 to fit the input structure of the network. By knowing where the object of interest is located within the scene or the general type of scene, we expect that the classes we have can be narrowed down sufficiently. For example, if a sea can be segmented within a scene, possible classes can be narrowed down, since only a limited number of objects can interfere with this segmentation [57].

The available datasets are only image-based and not semantically labelled. They have used an automatic semantic segmentator. The limitations of this segmentation-based approach are based on poor or imprecise segmentation. This was especially the case for cluttered or empty scenes. Figure 3.8 illustrates this problem that they have faced in the cluttered office or in the empty backyard. Finally, their approach achieved a better result (*i.e.,* 57 % accuracy) on the Places 365 dataset than *ResNet-50* (*i.e.,* 55 % accuracy) or *AlexNet* (*i.e.,* 52 % accuracy) [56].

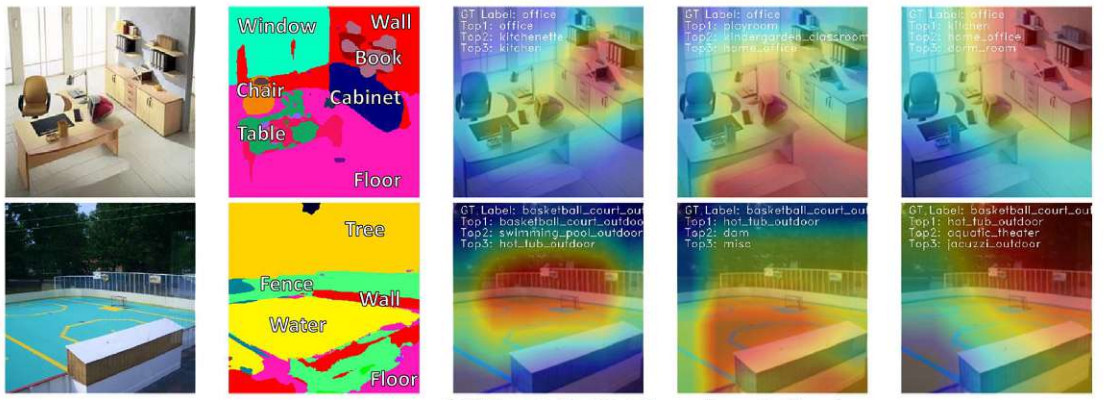


Figure 3.8: Illustration of class activation maps of two classes (*i.e.,* office and backyard) based on three different approaches. The two columns on the left show the RGB image and the semantic segmentation. The three right columns show the classification approach (*i.e.,* RGB branch, semantic branch and total network) [56].

Combination of CNN and HOI One of the hallmark tasks in human-object interaction (HOI) is visual context analysis. The capablilty to capture HOI is important when different interactions between humans and objects should be extracted. The way the human and the objects interact can be analysed using different approaches. Moutik et al. [58] proposed a

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.

two-stage approach with an object detector and scene classification to recognise the interaction between a person and a book. Figure 3.9 illustrates this connection of two branches to identify HOI.

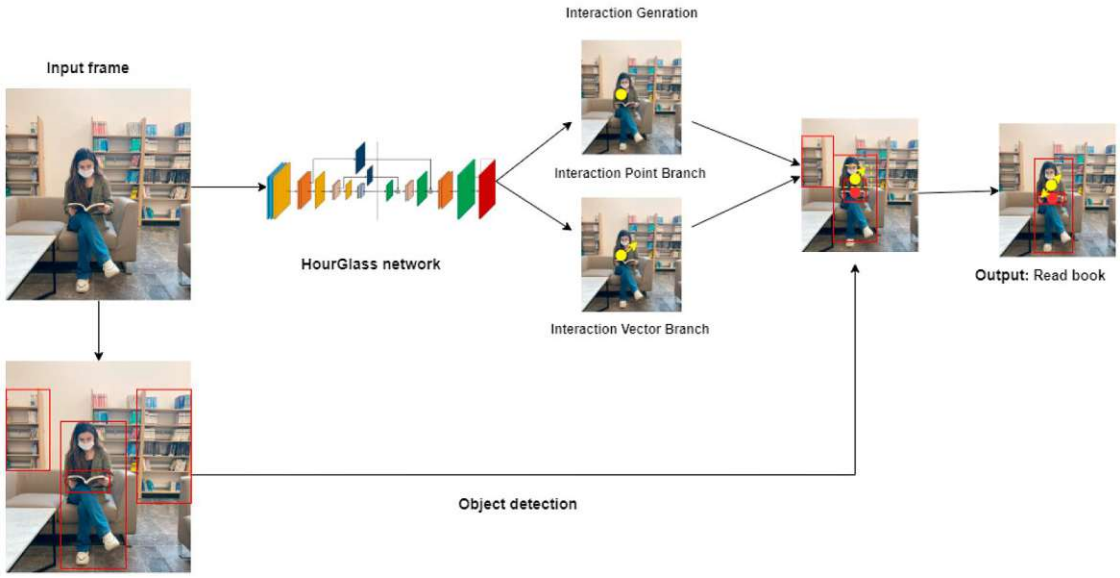


Figure 3.9: Schematic illustration of a CNN workflow for HOI detection, which is composed of feature extraction, interaction generation and interaction grouping. Within the interaction generation, interaction points and interaction vectors are produced. In the last step, interaction grouping, these are combined to give a final prediction [58].

Both branches are learning collaboratively, allowing to recognise the activity efficiently. This is based on knowledge distillation, which aims to transfer knowledge from one model (*i.e.,* teacher) to another model (*i.e.,* student). The basic idea behind this principle is that the teacher model transfers knowledge to the student model to obtain a competitive performance. As teacher model, they have used scene recognition to identify different semantic regions and as object detector, they have used a feature pyramid network (FPN). When both are combined together, the location of the desired object can be identified more easily. The interaction between a human and an object is identified by setting interaction points based on the human and object centre points. Finally, the object learns to calculate an interactive vector that illustrates the type of interaction. If the objects are getting closer, the length of the vector becomes smaller, indicating an interaction [58].

Combination of CNN and LSTM Padoy [8] and Yengara et al. [59] developed a phase recognition algorithm that was applied in the medical sector, especially in the operating room during surgeries. Due to the human-machine interaction within the operating room, context-aware systems are of great importance to strengthen this interaction. Therefore, their work focus on real-time algorithms for automatically recognising surgical phases, allowing better management within the operating room. This system is aware of the surgical context, human

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.

interactions, and movement, including activities (*i.e.,* surgical workflow) that take place inside the operating room. This context-aware system allows to automatically notify the user regarding the progress of surgeries. Their approach relies purely on captured laparoscopic videos, which should give an approximate remaining time for the current task. In case of any inconsistency in the workflow between humans and machines, this system can make recommendations to alter the workflow. However, since its objective is to recognise the current phase of surgery, temporal information is required to identify progress [8, 59].

Their developed phase recognition algorithm consists of a combination of CNNs with a recurrent neural network (RNN) (*i.e.,* long-term memory network (LSTM) (see Fig. 3.10). While CNNs extracts spatial features from different images, LSTM allows sharing of temporal knowledge between these. In particular, LSTM is important to draw conclusions about the correlation between different states to identify the current phase. The proposed network contains seven convolutional layers [59] that are connected to LSTM followed by a fully connected layer to predict classes. End-to-end training of a two-step model (*i.e.,* CNN-LSTM) requires more effort than for a single-stage model, since the loss has to be backpropagated through both the LSTM and CNN. Backpropagation through an LSTM network is not identical to that of a CNN and requires a specific backpropagation through time (*i.e.,* BPTT) algorithm, which works by unrolling all input timesteps [8, 59].

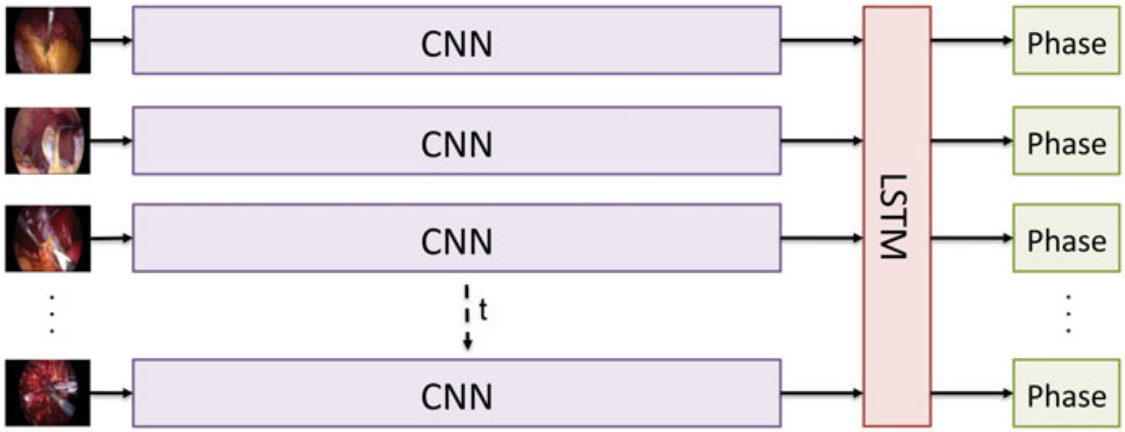


Figure 3.10: Schematic illustration of the CNN-LSTM architecture for surgical phase recognition. Multiple states are fed into CNNs that are connected to a LSTM model [8].

They conducted training on the *Cholec120* dataset, consisting of 120 lacroscopic videos [8] performed by 33 surgeons [8]. For training, the authors have set the following hyperparameters. The batch size for the CNN model was 50 [59] and for the LSTM 500x12 (*i.e.,* which corresponds to 500 time steps and 12 forward passes) [59]. As an optimiser, they have chosen SGD or ADAM. The training was carried out between 8.000 and 50.000 iterations [59]. The learning rate was set to 10^{-4} [59] if the model was pretrained, or 10^{-3} [59] if the model was not pretrained. Furthermore, the learning rate was scheduled for every 1/3 [59] of the maximum iterations, with a decay factor of 0.1 [59]. Generally, they observed that

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.
TU Bibliothek Your knowledge hub WIEN

CNN-LSTM training converges to poor local optima unless CNN is trained independently from the LSTM network. This application is a perfect example for dynamic scene recognition in which activities within the surgery room can be analysed [59].

Another work includes LSTM-CNN for Human Acticity Recognition (HAR) by analysing the acceleration of different body sensors. They proposed different model architectures for F_1 performance comparsion against the UCI-HAR[11] dataset, which is a dataset that includes 30 different human activities. Their initial attempts were based on a CNN architecture, which they optimised by including a global average pooling (GAP) layer instead of a fully connected layer and a final batch normalisation layer to stabilise the output, for which they reached a F_1 of 93.35%. Since the recording of the activities of the body sensors is a temporal sequence, temporal information can be analysed. Therefore, they improved the CNN architecture by a LSTM layer, which final model achieved a F_1 of 95.78%. However, they observed an increase in computation time per epoch from 1202 to 9416 ms, respectively [38].

Other works have shown the difference between static and dynamic networks for scene classes. Therefore, Feichtenhofer et al. [60] proporsed a temporal ResNet architecture, the T-ResNet architecture, to improve scene classification performance. Furthermore, a dynamic scene dataset *YUP++* was used, which was divided into static cameras and moving camera classes and contains classes such as *Falling Trees*, *Waving Flags* or *Fireworks*. Their proposed T-ResNet architecture achieved a classification increase of 6% [60] compared to the base ResNet architecture for static cameras. However, for moving camera videos, the T-ResNet architecture outperforms the ResNet architecture by 8% [60]. Remarkable is that although ResNet architecture is only analysing static images, the average classification performance of moving camera images drops by 13% [60] compared to static camera images. Another interesting approach was developed by Gu et al. [61], who tried to extract moving targets from satellite video scenes. Therefore, they utilised the T-ResNet architecture which achieved around 9% accuracy improvement compared to a common two-stream CNN. Lastly, another CNN-LSTM approach with spatial-temporal attention mechanism was used by Zheng et al. [62]. Their goal was to classify underground mine videos into possible categories. They have achieved an increase in accuracy of 2% with CNN-LSTM compared to the common CNN architecture [62].

Object-based Models for Scene Recognition

While a scene-centric classification approach focusses on the geometric properties of the environment, object recognition focusses on detection within images. Most of the methods for classifying scenes (e.g. kitchen, bathroom, etc.) rely on local and global properties of images, using feature-based methods. However, observations have been made that some indoor scenes

[11] https://archive.ics.uci.edu/ml/index.php, last accessed on 25.11.2022

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.
TU Bibliothek Your knowledge hub WIEN

are better described by the inherent objects. Object detection can be done either by a single-stage method or by a multi-stage method. A single-stage method, You Only Look Once (YOLO), is a widely used object detection algorithm used for real-time applications. Another object classifier, region-based convolutional networks (R-CNN) is a two-stage detector with a pipeline of object localisation and classification. In contrast to a single-stage approach, both the object localisation and classification tasks have to be optimised, making it less stable for training. Nowadays, the detection of controlled environments for object detection works pretty well. However, uncontrolled environments such as occlusions, arbitrary viewpoints, and cluttered environments remain a challenge. For example, it is easy for the object detector to detect a table, but if it is cluttered with many objects, detection becomes more difficult. Therefore, the extraction of global features from images with chaotic background results in a degraded model performance, compared to images of objects with a plain background. This suggests that a chaotic background introduces noise into the features. Generally, for each image where objects are not detectable due to the different conditions (*i.e.,* lightning, spatial relations, etc.), image classification can be the method of choice. [31, 44, 63, 64].

Compared to object recognition, scene recognition not only identifies the target of objects, but also spatial correlations between objects. Therefore, the advantage of image classification is that they yield more enriched spatial information than object detection because of the advanced recognition pattern. However, a promising approach is the combination of object-centric and scene-centric approaches. Harzallah et al. [65] show that the combination of object localisation and image classification can increase scene identification, as intraclass variations can be better handled. They believe that image classification and object detection use different information for processing, depending on the scale of the object. Therefore, better results are expected when both are combined. This assumption can be verified by the observation that processing an image by either a classifier or a detector leads to different results due to different extracted features. While image classification tries to predict the class of an image, object detection tries to identify the localisation of one or more objects inside an image. If the object is small and appears in a non-standard context, the object detector may manage to find it, while the classifier cannot detect it anymore. However, if the object occupies a large part of the image and cannot be fully displayed (*i.e.,* detector struggles to identify truncated objects), image classification can be the matter of choice [31, 65].

The performance improvement, by the combination of an image classification method and an object detector method can illustrated by the example of two images of a car dataset. The first image (see Fig. 3.11a) includes cars inside a street. The striking feature is the small size of the car, which appears in a non-standard environment. The detector might find the object and identify the class, while the classifier will struggle. If an image (see Fig. 3.11b) includes mainly truncated cars, they may be hard to identify by the object detector, but the classifier might have enough information to decide on the presence of the object.

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.
TU Bibliothek Your knowledge hub WIEN

This increased classification accuracy is based on the different contents observed by both approaches within an image [65].

(a) (b)

Figure 3.11: Illustration of two different images of a car dataset that are beneficial to be identified either for (a) object localisation or (b) image classification [65].

3.3 Comparison of the Algorithms

Although the authors [44, 65] have shown that for scene classes it would be beneficial to use object detectors, a single-stage object detection approach might not detect the richness of the inherent features within the scenes. This is especially the case for cluttered indoor scenes or annotation ambiguity. To avoid the problem of annotation ambiguity, where the same objects are presented within different scenes, object detection should not be used alone. Therefore, we are benching object-centric approaches and focus on a scene-centric approach, as the authors [65] state that scene-centric approaches can yield more enriched spatial information.

The choice of the right deep learning architecture is important for extracting the richness and diversity of the scenes. If the resolution of the input scene is not large enough, the full context of the scene cannot be captured. For high-resolution images, the network depth has to be deep enough to increase the receptive field and to detect finer features. The AlexNet network, established by Jmour et al. [48], may not be deep enough to perfectly capture the context of scenes, without losing spatial information from images. This is caused by large convolutional kernels and high stride. Therefore, the applied filters are too inaccurate and fine details, such as lines, cannot be captured from indoor scenes. Although Xia et al. [38] show model improvements, AlexNet may not be the best application for our use case, but can be chosen as a baseline model.

Deep residual networks, introduced by K. He et al. [50], enable to increase the depth of the neural network without decreasing the generalisation ability of the network. They show ex-

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.
TU Bibliothek Your knowledge hub WIEN

cellent performance metrics for well-known datasets such as *Places365*, which contain scenes similar to our application environment. Usually, since such networks include the safety aspect of identity mapping, the network parameters can be increased without loss of performance and the danger of vanishing gradients. Furthermore, the high number of layers enables the detection of the richness of large scene datasets such as *Places365*. For example, Shah et al. [51] provided an approach to use a pretrained 152-layer ResNet architecture to perform a scene classification task within class rooms. Furthermore, improving this architecture with minor tricks, mentioned by T. He et al. [53], can further enhance the ResNet classification performance without increasing the complexity of the mentioned architecture. To further improve RTA, a ResNet architecture with an optimal trade-off between classification performance and classification speed might be interesting.

Selective joint fine-tuning is an effective way to transfer learn a small dataset, by simultaneously training the target dataset with the k-nearest neighbours of the source dataset. However, this is not beneficial for all learning tasks. If the indoor scene classification approach is narrowed down to a small number of categories, transfer learning probably will not struggle finding similar images and can be done in its original form without any advanced approach. Considering that the application domain will increase for future developments, selective joint fine-tuning can become the matter of choice [54].

Lopez-Cifuentes et al. [56] established another technique, semantic-aware scene recognition, to cover the diversity of indoor scenes. In general, semantically aware scene recognition can help identify objects or regions of interest within scenes more easily. Although semantic-based approaches show good clasification performance, problems for empty or cluttered scenes are known for this technique. This might be challenging for our application, since it is based on indoor scenes within a factory, and cluttered workspaces or large empty-floor scenes are usually the case. Therefore, semantic classification might be interesting for extracting more fine details if the application is expanded to differentiate between workplaces of one class but will not be further pursued for the current application.

Moutik et al. [58] proposed a hybrid model system that is capable of identifying the interaction between the human worker and the object. This system is based on the centre points of the human and object and an interaction vector, which alteration unveils temporal information of static scene images. Another hybrid approach, including object detectors and scene classification to capture different perspectives of the scenes, was mentioned by Harzallah et al. [65]. Both of these mentioned hybrid deep models have the capability to extract more precise and detailed information about humans and the workplace. However, for smart factory applications with AR, where only the hands of the human worker are visible, detecting interactions between the human and the environment is hardly possible. Furthermore, as we have already mentioned within the challenges of image classification, workplace

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.

recognition should have the ability to operate in real time and work fluently. Therefore, although multistage detectors generally achieve better detection performance than one-stage detectors, one-stage detectors are mostly more time efficient and have greater applicability to RTA. In general, multistage detectors should be neglected, unless their increased feature extraction provides some beneficial output [64, 66].

Another hybrid-based model looks promising for further improving classification performance. This hybrid-based model mentioned by Zeng et al. [31] tries to identify different scalar ranges of the model and weights them for the final classification. As we are processing scenes, there might not only be high-level features important but also some low-level characteristics that might be beneficial for classification. However, this approach was not chosen, since the assembly factory scenes show similar low-level characteristics where the performance increase through this hybrid-based model is not guaranteed.

Looking at the CNN-LSTM approach, carried out by the authors [8, 59], to identify the activity progress of states, the LSTM extension is responsible for processing temporal knowledge. Furthermore, the authors [60, 61] have shown that a *CNN-LSTM* architecture could achieve remarkable improvements in classification accuracy (*i.e.,* up to double digit numbers) compared to standard CNN architectures. However, through this SOTA analysis, two main drawbacks of the CNN-LSTM architecture have been identified. On the one hand, as stated by Xie et al. [55], the LSTM extension increases the computation time by a multiple. This is not desired if RTA on a HMD device must be guaranteed. The other aspect is that meaningful temporal information must be present to significantly increase the classification performance of a CNN-LSTM approach. Therefore, studies [60, 61] with a significant temporal change within their video classes could achieve remarkable results with the CNN-LSTM approach. However, if the temporal information is not significant, as in works [38, 62], the classification performance cannot be dramatically increased using the CNN-LSTM approach. Therefore, the trade-off between increased classification accuracy and calculation speed is no longer worth it. Lastly, another important information is that despite static or dynamic scene recognition, the classification performance by datasets with moving cameras will definitely decrease significantly [60]. This is an important aspect to consider when deploying the network on a mobile HMD device.

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.
TU Bibliothek Your knowledge hub WIEN

Chapter 4

Proposed Method

In this chapter, a module for scene-awareness in HMD-assisted work scenarios is introduced, and the proposed method to accomplish it is described in detail. Therefore, the general challenges and requirements of the planned module are mentioned, and basic functionalities of assistance systems that are treated by the context-aware middleware are explained. In the implementation design chapter, the necessary parts that have been accomplished to generate the context-aware middleware are included. Therefore, the software is described in detail regarding its dataset, its model architecture, its training procedure, and its evaluation results. Finally, in the experimental chapter, the built image-based workplace recognition middleware is tested by a use case within the institute pilot factory. In this use case, the algorithm is validated for its interference speed and usability for the human worker. Furthermore, the functionality of the implemented software is evaluated with respect to the classification accuracy of use case workplaces. In summary, important application criteria of this use case are derived for the future development of context-aware digital assistants for workers.

4.1 Challenges and Requirements

The possibility of a system becoming context-aware is based on some essential criteria. This system has to capture the environment of the human worker in the form of images, classify these images into one possible workplace, and should feed back the results to the human worker in an appropriate time. Moreover, it has to be pre-trained by a dataset, which includes images from the different workplaces. Lastly, to establish such an assistance system, a considerable amount of software as a middleware has to be developed. In the following are the most important challenges mentioned which need to be considered during the development of such a module.

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.
TU Bibliothek Your knowledge hub WIEN

Dataset Quality: The quality of the dataset is an important characteristic for the overall performance of the prediction. Since the algorithm is based on neural networks that include changable parameters, training the algorithm on a specific dataset (*i.e.,* that includes images from workplaces) is necessary. Without this qualitative dataset the model predictions become poor and the system cannot achieve the desired support for the human worker. Because there is no scene dataset available for the manufacturing industry, a generation of the dataset that contains the richness and diversity of the workplace is desired. Challenges for the development of a dataset for scene classifcation can be data availability or similarity between classes.

Real Time Analysis: RTA is characterised by system execution in real time. A characteristic factor, the latency, is the time between the triggered event and the required response of the workplace classification. This gap, where no information is recognized, should be as small as possible and can be reduced by optimising the processing steps of the algorithm. For this reason, the processing steps include the collection and transfer of images to the location of the analysis. Furthermore, for one captured image, one forward path of the neural network for one captured image must be calculated. Once the system knows the category of the workplace, the output is transferred and displayed to the human worker in an interpretable form. Overall, the necessary reaction to workplace detection can be given within a latency step and fully depends on the processing speed of the used device. In general, RTA is important for workplace-image classification to immediately support the human worker. Otherwise, usability of the system will be low [39].

Black Box Challenge: The rise of algorithms with decision-making capabilities leads to mistrust if the human worker cannot comprehend the output without any explanation. Therefore, the system should always try to find a way to explain its decision to increase its trustworthiness. Besides the desired instructions for the human worker, the algorithm must explain its decision based on a processing result (*e.g.,* accuracy of the prediction, by detected features). Otherwise, the negative effect of no explanation can lead to system abuse and can harm human working conditions [37].

Effectiveness of Context-Awareness: The development of the classification system alone is not enough to get detailed feedback on its functionalities. Moreover, the system should be tested with an experiment or an use case to show its strengths and weaknesses. Since the proposed technology is only part of an assistance system, which is responsible for capturing the environment of the human worker through context awareness, its full usability for the human worker cannot be tested yet. Therefore, the main focus should be to provide feedback on classification performance within the application area, the dataset quality by wrong classification, and processing time.

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.
TU Bibliothek Your knowledge hub WIEN

4.2 Functionality Classification

The desired functionality of the assistance system and the module can be classified within Fellman framework (see Table 2.1). To get an idea to what extent the middleware should be part of an assistance system, Table 4.1 demonstrates some functionalities. These functionalities are important characteristics for identifying an assistance system. However, only those that have been fully completed by the developed middleware are discussed.

State detection & Context sensitivity: State detection is the ability to gather information about the current state, especially about the working environment. Therefore, to ensure that the working environment is captured, HMD devices worn by human workers can be beneficial. This leads to the detection of the current human workplace without capturing the human worker. Finally, this system is context-sensitive by the capability distinguish between different workplaces.

Table 4.1: Implementation of the assistance system framework established by Fellmann et al. [4] for the developed middleware of image-based workplace recongition.

Category	Features	Attributes
Information	Generation	-
	Presentation	-
Intelligence	State detection	Automatically
	Context Sensitivity	Environment
	Learning Aptitude	-
Interaction	Control	Cooperation
	User Involvment	Low
	Input	Modern
	Output	-
	Extent of Immersion	-
System characteristics	Transportability	-
	Robustness	-
	Technology Readiness Level	-

Input: The inputs for the middleware are images that are detected by a camera system (*i.e.,* HMD devices). Furthermore, the extracted images are fed into the algorithm (*i.e.,* neural network) for classification.

User involvment & Control: This middleware should be able to detect the workplace where the human worker operates without user involvement. The easiest form of location detection are cameras (*e.g.,* HMD devices) that human workers wear to capture their visual

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.

location. This allows one to identify the user working environment without directly monitoring the human worker. Since direct user involvement is not required, the control is performed indirectly by user movement and can be well established in the working tasks. Therefore, the execution of the job might be supported by reducing cognitive stress without additional strain on the human worker.

4.3 Implementation design

The implementation design chapter shows the steps taken to create the middleware that allows image-based workplace recognition. Initially, starting with a general picture of the workplace application area and the final specification of the dataset classes. Furthermore, the construction steps of the dataset are explained. Finally, the core components of the model architecture, model training, and network results are mentioned.

4.3.1 Dataset

By specifying the assembly factory as the application, it is clear that various objects can exist within each workplace. In general, these objects are not uniquely assigned to one workplace and can be present in multiple classes. For example, a robotic arm can be used as a processing machine if used for welding, but also as a conveyor belt if it moves products from one place to another [66]. Since any workplace scene can be seen as a formation consisting of multiple objects, it is not possible to classify workplaces solely as objects without considering the environment. Hence, to give prospects for the environmental context, the algorithm must have knowledge of all scenes. Therefore, the algorithm must be trained on a dataset that contains characteristics of all workplaces.

Publicly available scene datasets (*i.e., SUN397*, *Places365* or *MIT67 Indoor classes*) can be a perfect choice for training, since they contain classes similar to industrial workplaces. However, not all classes of publicly available datasets are similar to the workplaces of our application. When taking all classes of these public datasets for pretraining, the parameters of the network might overfit. Hence, the generation of truncated datasets with the most relevant classes, by evaluating each class regarding its suitability as workplace, can create a remedy. Unlikely, these truncated datasets do not represent the full complexity of the actual application. Therefore, we mainly use the similarity of these datasets to pre-train our model using transfer learning. As revealed through SOTA research, the *Places365* dataset has achieved the best results in terms of richness and diversity compared to other datasets, which is why it is used for pre-training our model. Since classes of these datasets still might have inter-class variations to the actual application classes, they do not fully represent the features of the application objects and content. For this reason, the main objective was to

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.
TU Bibliothek Your knowledge hub WIEN

develop a dataset that best fits the conditions of the assembly factory application domain. Furthermore, our own developed assembly factory dataset should finally meet the features of the workplaces on which the model should be fine-tuned. Lastly, the dataset can also include images from the actual workplace location. Although the generalisation capability will not increase through this act, the model will show improved classification accuracy for the specific environment. However, just training on this specific application is not beneficial, since the dataset will not have the robustness to perform consistently for small workplace changes.

(a) *Assembly Line* (b) *Machine Tool* (c) *Table*

(d) *Shelf* (e) *Robot* (f) *Terminal*

(g) *Empty* (h) *Too Close* (i) *Combined Workplaces* (j) *Neutral*

Figure 4.1: Illustration of the relevant categories of the *Assembly Factory Workplaces* dataset. While categories (a-f) represent a possible workplace class, categories (g-j) are special classes that set application boundaries and improve transitions between workplaces.

For our self-developed assembly factory dataset, six different main scene classes (*i.e., Assembly Line*, *Machine Tool*, *Robot*, *Shelf*, *Table* and *Terminal*) were chosen, each illustrating one possible workplace. Furthermore, to support transitions between workplaces and restrict

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.
TU Bibliothek Your knowledge hub WIEN

the boundaries of the application, four more classes (*i.e., Empty*, *Too Close*, *Combined Workplaces* and *Neutral*) that should help project the application area more clearly were added. Figure 4.1 shows sample scenes, resized to 256 x 256 pixels, for each class in the dataset.

Since the four special classes should represent the application area more accurately and improve the relationships between workplaces (see Figure 4.2), each is described in more detail to show its intention:

For example, when the user with AR glasses is too far from one specific workplace, it can happen that multiple workplaces are present in one scene. Therefore, for scenes that include multiple workplaces, which are difficult to uniquely classify into one category, the category *Combined Workplaces* was generated. The indentification of that class should notify the user to adapt his position more precisely to show the occupation along a workplace.

Another case might occur when the user with AR is located in an area of the assembly factory, where no workplace is represented. This can happen when the user is too far from any workplace and the objects are too small, so that no features can be detected. This class, *Empty*, should ensure that no workplace is identified instead of deciding in which workplace the user is presented.

Likewise, if the user is too close to the desired object, only low-level features can be extracted. In general, the system may have difficulty extracting meaningful features to identify unique workplaces due to a similar close-up appearance. Therefore, the system should inform the user to relocate for new classification.

Lastly, neutral backgrounds are added as a category to classify any scenes that are not within the scope of the assembly factory, where no workplace information is available.

Overall, the behaviours of the class *Too Close* or *Combined Workplaces* are based on the principles of visual inconsistency and annotation ambiguity which were already well discussed in Chapter 3.1.1. Furthermore, the class *Empty* can be seen as a new class with its own detectable features. Generally, when establishing a new dataset for scene classification, it is wise to identify its generalisation capabilities (i.e., performance on unseen data) by testing it with a popular CNN architecture (i.e., AlexNet, GoogleLeNet, VGG16, etc.). By doing this, the well-known problems of dataset generalisation can be identified early on [31, 40].

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.

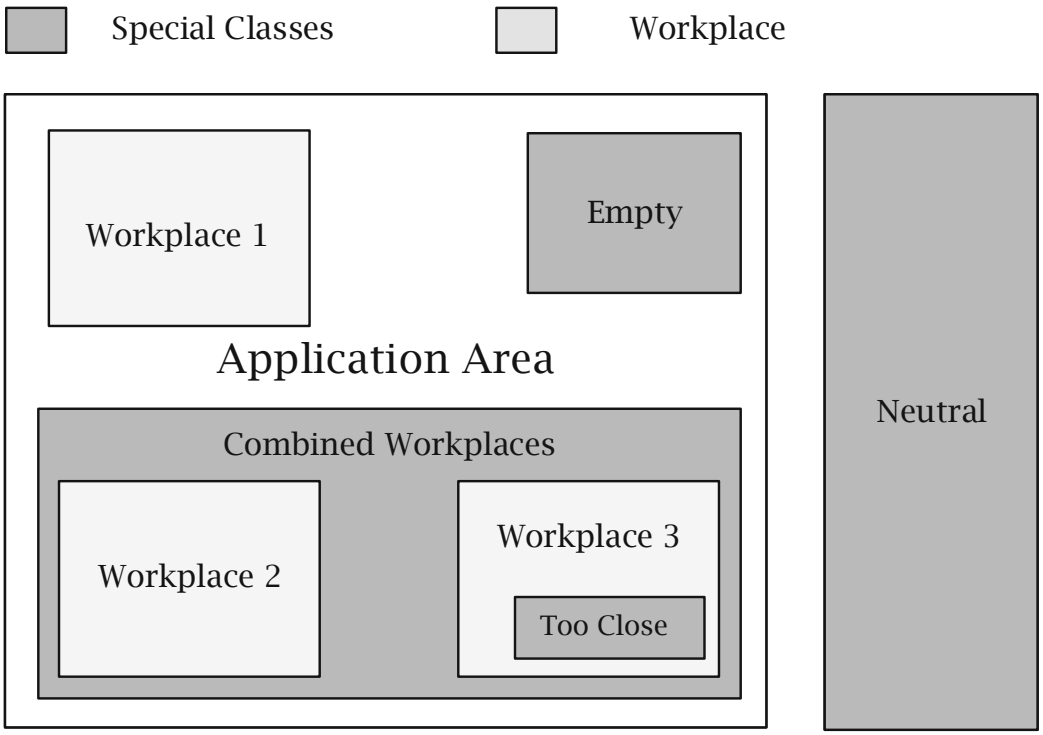


Figure 4.2: Schematic representation of the application area of the *Assembly Factory Workplaces* dataset, including the four special classes (*i.e., Empty*, *Combined Workplaces*, *Too Close* and *Neutral*).

Construction of the dataset

The assembly workplace dataset was inspired by the construction of the *Places365* dataset and was developed based on three fundamental steps [46]. Starting with querying and downloading of the images, labelling the images with ground truth categories, determining inconsistent images within a class by screening and finally the synthetic data generation to fill new or existing classes with artifical images.

The application classes are built on a set of keywords that are consulted to query multiple search engine requests (*i.e.,* on Google Images, Bing Images, and Flickr). However, one dataset class can represent multiple keywords to ensure that more images are found through various search engines. Furthermore, since most search engines have an inherent limitation of search results, filter categories (i.e., colour, type, size, date) are set to increase the diversity of the findings. When scraping on multiple search engines with multiple filters, it must be ensured that duplicate images are not presented. Therefore, during image scraping, a history is generated to ensure unique images by using the Uniform Resource Locator (URL). Since unique URLs are not entirely an indication of duplicates, it is a good choice to additionally apply Principal Component Analysis (PCA) to remove the same images from each category.

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.

This has been done by the software *Vispics*[1]. However, similar images might not be a large interference and some duplicate images will not cause a significant performance decrease (*i.e.,* unless diversity is large enough), since data augmentation procedures are applied anyway during training.

In the next step, the ground-truth label verification of the image was performed manually in a three-iteration process. In the first iteration, all images that did not fit any class were removed (around 60% of the total scraped images). In the second iteration, each image was analysed to identify whether the objects and features presented fit better into another category. During this iteration, images were rearranged between the different classes with the focus on generating classes with equal size. Furthermore, scraped images that do not fit any of the six workplace classes were classified into one of the two special classes (*i.e., Empty*, *Too Close*) or were deleted. In the third iteration, the premature dataset was trained on the model, and the resulted confusion matrix was examined to indicate intersections between two classes, which were further optimised.

Lastly, for categories with no samples or a low number of samples, synthetic image data was generated. Therefore, the special class *Too Close* was further filled with image fractions of the six workplaces by extracting 1/25 close-ups of their total image size. The other class *Combined Workplaces* was generated based on the combination of p random images from the six workplace classes. The number of random images is chosen with the rationale that some workplace classes already include a portion of another class. This cannot be avoided, since the transitions between workplaces are blurry. Therefore, p might be a parameter, easy to adjust, to enable transitions. Moreover, the *Neutral* class was created based on the generation of images with 256 x 256 random pixel values. Despite the special classes, synthetic images were also generated for the workplace classes. This procedure is only possible if the class includes unique objects within the scene. Therefore, the object is extracted from an image with an inappropriate background and superimposed on suitable backgrounds to generate new data. This procedure is especially useful if there is not enough data available for one category. For this dataset, the category *Robot* lacks representative data. Therefore, the *Robot* objects have been extracted from its background and augmented with different sizes on new backgrounds. The new backgrounds were based on scenes that illustrate common backgrounds for assembly factories (*i.e., Empty*).

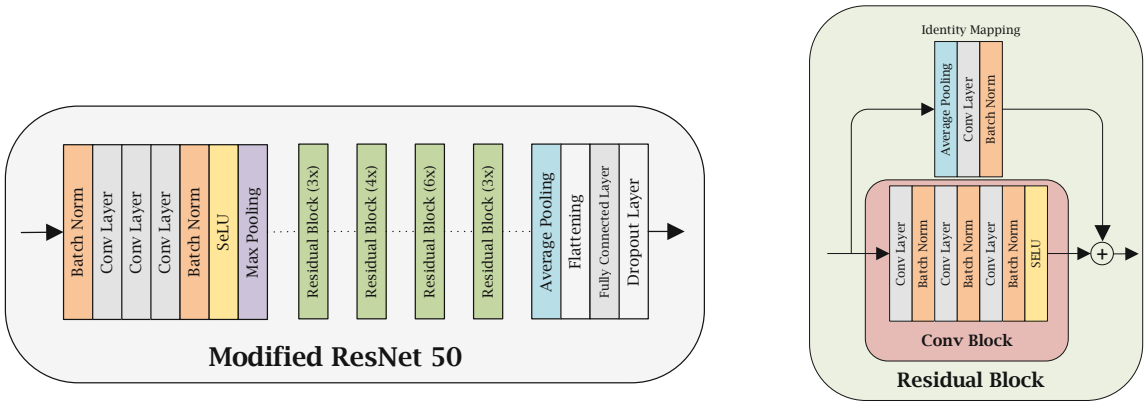
Finally, the finished *Assembly Factory Workplaces* dataset consists of 10 different categories with a total of 4695 images. What matters in multiclass classification is whether the classes are balanced, because usually models are biased towards larger classes (*i.e.,* trained more on this class due to more images). Generally, class imbalance was avoided by equally arranging the number of images for each class within the dataset.

[1]http://www.visipics.info/index.php?title=Main_Page, last accessed on 20.10.2022

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.
TU Bibliothek Your knowledge hub WIEN

4.3.2 Model Architecture

Once the dataset has been generated, the development of the neural network was carried out, which is the core of the image-based workplace middleware. Initially, the deep learning approach (*i.e.,* supervised) is selected based on the available data that need to be processed. However, based on the state-of-the-art chapter, research of previous applications has been carried out to identify and compare the most suitable architectures. As model, a modified ResNet architecture was chosen, which basic principle of residual networks being introduced by K. He et al. [50]. Based on the author Lopez-Cifuentes et al. [56], residual networks show excellent performance on the *Places365* dataset, which shows similarity to our dataset. The developed ResNet architecture consists of 4 different residual blocks and a total of 122 layers (see Fig. 4.3a). Each residual block consists of a convolutional block and an identity function to skip connections (see Fig. 4.3b). The convolutional block consists of convolutional and batch normalisation layers, stringed together. Scaled Exponential Linear Unit (SELU) activation functions are used as the last layer within each convolutional block of the network. Furthermore, the proposed network was modified on the basis of the tricks (B,C,D) proposed by T. He et al.[53], *i.e.,* adaptions changes of stride and kernel size for some layers to extract information without loss. As first layer, a batch normalisation layer was chosen to alter the images not normalised during preprocessing. Furthermore, the ReLU activation layers were replaced by the SeLU activation layers, as they have shown greater performance in test runs. Lastly, depending on the type of dataset (*i.e.,* training or validation), dropout was applied as a regularisation technique.



(a) (b)

Figure 4.3: (a) Schematic illustration of the architecture of the modified residual neural network. (b) Schematic illustration of a convolutional block and a residual block from the network.

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.

4.3.3 Preprocessing of the Data

The purpose of preprocessing the input data is to support the algorithm in its performance. The input image is a tensor with a shape (C, H, W), where C is the number of channels and H and W are the height and width of the image. Furthermore, the bundling of images into batches aligns a batch dimension in front, resulting in a four-dimensional tensor as input for the network. Knowing the height and width of the image, the input shape of the network can be correctly adjusted. It is noteworthy that the input shape of the network should be consistent, at least for the dimensions C, H, and W. Since all images of the dataset are RGB, each image consists of 3 channels. However, the height and width of the images within the dataset are not consistent. Therefore, independent of training and validation procedures, all images are resized into a shape of (256, 256). Despite image resizing, image preprocessing is also important to increase the diversity of the available dataset by data augmentation. These data augmentation strategies act as regularisation and were used solely for the training cycle. During training, data augmentation is applied as a sequence of transformations to preprocess the image. Each transformation changes the image with a certain probability to generate new images for training. For example, in one data augmentation strategy, the *RandomRotation* function rotates the input image within a range of 0 to 10. Furthermore, since two different datasets are used for training, the images within both can be normalised to allow weight sharing across the training procedures. Therefore, the input is normalised with the mean value and standard deviation of all input images (see Equation 4.1).

$$\boldsymbol{x} = \frac{\boldsymbol{x} - \mu}{\sigma} \tag{4.1}$$

Where, $\boldsymbol{x}$ is the input of the network, μ is the mean of the dataset and σ is the standard deviation of the dataset. By normalising image pixel values relative to the mean and standard deviation of the dataset, consistent results can be achieved when applying the same model to similar dataset images. For scene datasets, calulating the mean and standard deviation is not easy, but as we know through gradient descent, input data can be handled in batches. To find a remedy, the average mean is calculated over all batches. However, the standard deviation is more challenging since the average σ between batches is not equal to calculate across the entire dataset. The strategies for calculating the mean and standard deviation of a dataset originate from the information of Nikita Kozodoi[2].

4.3.4 Network Training

During the training procedure, the model parameters (*i.e.,* weights and bias) are optimised based on the dataset images with the goal to minimise the error in each iteration. There-

[2]https://kozodoi.me/blog/, last accessed on 02.10.2022

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.
TU Bibliothek Your knowledge hub WIEN

fore, images are grouped in batches and randomly shuffled before feeding to the network to ensure variety during training. Additionally, since algorithm training is a time-consuming procedure, multiprocess workers (*i.e.,* 2 for the training and validation dataset) speed up the training process by parallelising it. For the training *Adam* was chosen as the optimiser and *Binary Cross-Entropy With Logits Loss* (*i.e.,* is a combination of binary cross-entropy loss and a sigmoid layer) was set as the loss function. In particular, this means that the optimiser and loss function are selected during compiling to decide the model architecture. In the fitting step, the weights of the compiled model are adapted in each iteration. This is done by using the backpropagation procedure in a specific number of iterations. Whereas the total number of iterations can be calculated by $epochs \cdot \frac{dataset\,size}{batch\,size}$. After each training iteration of the training set, the network predicts the validation set to test its performance on unseen data. However, hyperparameters are the most crucial part of training a deep learning algorithm, and their right settings are important for the generalisation of the model. The learning rate is the most well-known hyperparameter, which is updated in each iteration. Furthermore, a cosine learning rate scheduler is implemented to change the learning rate for each epoch based on a cosine function. Finally, to test whether the model has developed correctly without any logical errors, small input data is fed to it to explore whether it can overfit [24, 26].

Pretraining

In reality an entire convolutional network with random parameter initiation is only rarely trained from scratch. The reason for this is that the self-developed datasets are generally not large enough to generalise well. Therefore, it is common to pretrain a model on a well-working dataset and use these weights as initialisation for the desired dataset. As pretraining, a truncated *Places365* dataset (*i.e.,* consists of 10 classes with a total of $4 \cdot 10^4$ images for the training set and $1 \cdot 10^3$ images for the validation set) is chosen, from which input images are preprocessed by the mentioned data augmentation strategies. The pretraining is carried out for 54 epochs with an overall training time of 8 h, a training batch size of 64 and a validation batch size of 32. Lastly, the initial learning rate α was set to $1 \cdot 10^{-2}$ with a cosine learning rate scheduler, which reduces α to $1 \cdot 10^{-5}$ for the total epochs. Finally, the proability for dropout of the dropout layer was set to 0.25.

Fine Tunining

After pretraining the model with the *Places365* dataset, the learnt features are transferred as weights for training the self-developed dataset. Besides the hyperparameters already set by the pretraining, some of them have to be adapted. As suggested by Yengara et al. [59] the learning rate is reduced for the training of the self-developed dataset, to avoid overwriting the learnt behaviour during prertaining by incorrect weight updating. Therefore,

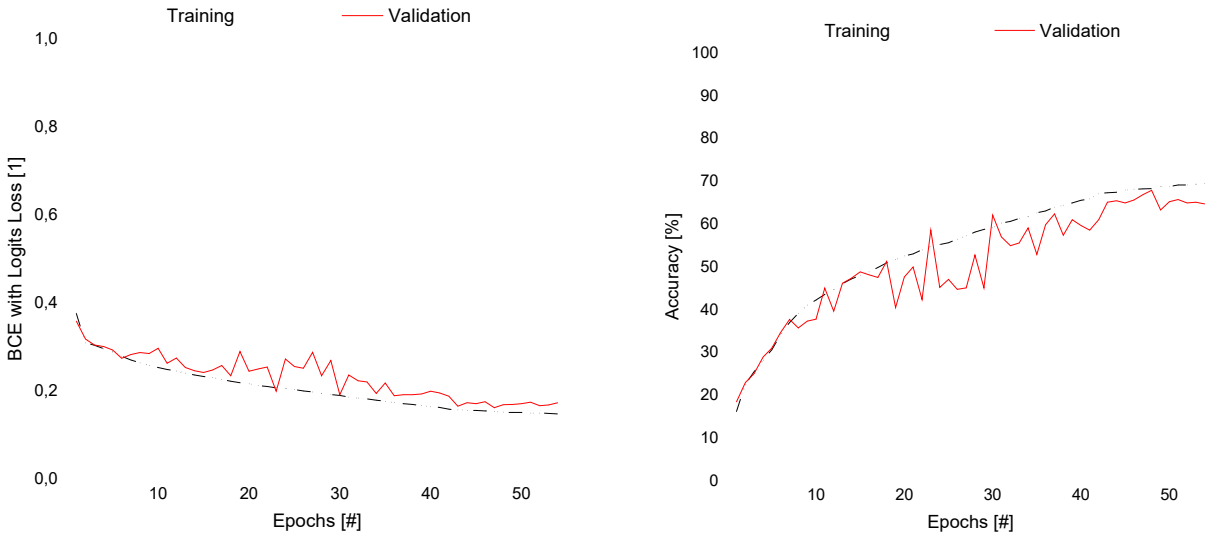
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

the initial learning rate is set nearly one order of magnitude (*i.e.,* to $2{\cdot}10^{-3}$) lower than during pretraining. Furthermore, the number of training epochs is set to 34 and the probability of dropout to 0.35. Lastly, all augmentation strategies stayed the same as for the pretraining, besides the model output size adjustment to the new number of dataset classes.

4.3.5 Network Results

Pretraining Results

Figure 4.4 shows the *Binary Cross-Entropy With Logits Loss* and accuracy over the number of training epochs (*i.e.,* 54) for network pretraining on the truncated *Places365* dataset. Both figures show the pretraining results for the training and the validation dataset. Within Figure 4.4a the loss of the training and the validation dataset declines over the number of epochs. For the total number of epochs, the training dataset reaches a loss of 0.15 and the validation dataset reaches a value of 0.17. Although regularisation strategies of data augmentation and a dropout layer with a propability of 0.25 are applied to the training dataset, the loss of the training dataset is smaller than of the validation dataset. However, training and validation loss can be adjusted in relation to each other based on the previously mentioned strategies.



(a) (b)

Figure 4.4: Results for network pretraining over the number of epochs.(a) *BCE with Logits Loss* for the training and validation dataset, (b) accuracy for the training and validation dataset.

Furthermore, Figure 4.4b shows the classification accuracy for the training dataset and the validation dataset. Both, training and validation accuracy ascend for each epoch and the training accuracy reaches higher values than the validation accuracy. Remarkable are the

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.
TU Bibliothek Your knowledge hub WIEN

fluctuations within the validation accuracy which can be an indication for a small validation batch size (*i.e.,* 32). The decisve value for the validation accuracy lies arround 65% which is satisfactory compared to similar works [56] for the entire *Places365* dataset.

Furthermore, Table 4.2 shows a more detailed representation of each class for the pretraining in the form of the confusion matrix. The entire table is normalised for each actual class prediction. Through this normalisation, the main diagonal values indicate the classification accuracy for each class. Moreover, the mean of the TN and TF values indicates the total overal accuracy. Generally, the confusion matrix reveals the learnt features for different classes. For pretraining, an equally distributed classification accuracy could be reached for each class, despite the *Repair Shop*. This class has a low classification accuracy of 0.34 and shows a strong misclassification with other classes (*e.g., Garage Indoor*). The reason might be that weak features are recongized, which interfere with other classes. Lastly, to interpret the entire result, the F_1 score was calculated for the pretrained model which lies around 0.66.

Table 4.2: Confusion matrix for pretrained neural network, where the vertical label indicates the predicted condition and the horizontal label shows the actual condition. The values are rounded to two decimal places and all values above 0.10 were highlighted.

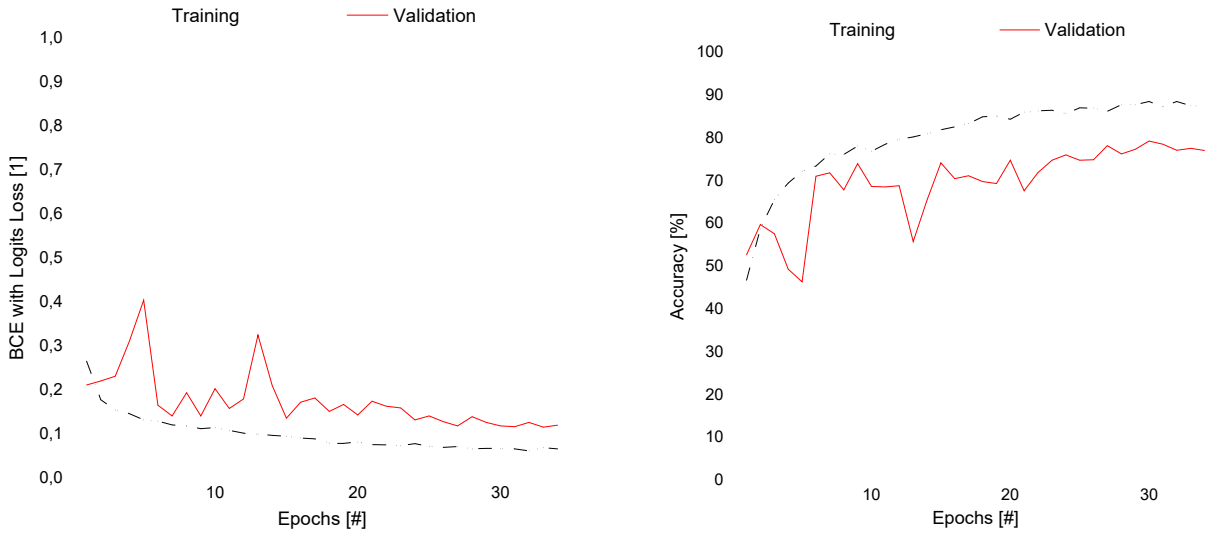
	Archive	*Assembly Line*	*Basement*	*Dining Hall*	*Garage Indoor*	*Office*	*Pantry*	*Physical Laboratory*	*Repair Shop*	*Storage Room*
Archive	**0.83**	0.04	0.00	0.00	0.00	0.00	0.00	0.02	0.03	0.08
Assembly Line	0.04	**0.61**	0.01	0.00	0.04	0.05	0.00	0.02	**0.20**	0.02
Basement	0.02	0.06	**0.61**	0.05	**0.12**	0.02	0.01	0.02	0.07	0.01
Dining Hall	0.02	0.04	0.00	**0.84**	0.00	0.05	0.00	0.01	0.03	0.01
Garage Indoor	0.00	0.01	**0.10**	0.00	**0.68**	0.01	0.03	0.00	0.05	0.07
Office	0.05	0.00	0.01	0.00	0.03	**0.75**	0.02	**0.10**	0.03	0.00
Pantry	0.01	0.00	0.01	0.02	0.01	0.02	**0.86**	0.00	0.04	0.03
Physical Laboratory	0.01	0.09	0.00	0.00	0.02	**0.12**	0.00	**0.64**	**0.11**	0.00
Repair Shop	0.04	0.04	0.07	0.02	**0.20**	0.07	0.03	**0.11**	**0.34**	0.08
Storage Room	**0.10**	0.01	0.03	0.00	0.03	0.04	0.06	**0.12**	**0.10**	**0.50**

Fine Tuning Results

The pretrained model is further fine-tuned with the *Assembly Factory Workplaces* dataset for 34 epochs. Figure 4.5 includes the training results for the *Binary Cross-Entropy With Logits Loss* and the accuracy over the number of epochs for training and validation. As the model does not have random initialisation of the weights, the loss values and corresponding the accuracy already start at a lower, or higher initial value. As seen in Figure 4.5a, the training dataset reaches a loss of 0.06 and the validation dataset a loss of 0.12 at the end

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.
TU Bibliothek Your knowledge hub WIEN

of the training. It should be mentioned that the model stops training automatically after 7 epochs, for which the validation loss cannot be further decreased. Figure 4.5b shows a strong discrepancy between validation and training accuracy, which indicates that the model overfits. This can be avoided by increasing the dataset size or implementing more regularisation techniques. Finally, the model reaches a classification accuracy of 0.77 with the corresponding F_1 score of 0.76.



(a) (b)

Figure 4.5: Results for the fine-tuned model on the *Assembly Factory Workplaces* over the number of epochs.(a) *BCE with Logits Loss* for the training and validation dataset, (b) accuracy for the training and validation dataset.

In the second step, the confusion matrix of the fine-tuned model was built (see Fig. 4.5). Remarkable is the high classification accuracy for the classes *Combined Workplaces* (*i.e., 0.98*) and *Neutral* (*i.e., 1.00*). Although it is intended that *Neutral* does not show any misclassification with any other class, it was expected that *Combined Workplaces* has a strong misclassification with other workplaces, since it should present the transistions between classes. The weakest classifcation accuracy of a workplace class was achieved by the *Table* class (*i.e., 0.40*), which shows a strong misclassification with the *Machine Tool* class (*i.e., 0.22*) and the *Terminal* class (*i.e., 0.18*). This implies that the detected features within the *Table* class might not be unique enough, since the tables are also represented within the *Terminal* and *Machine Tool* classes.

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.

Table 4.3: Confusion matrix of the neural network, fine-tuned on the self-developed dataset. The vertical label indicates the predicted condition, and the horizontal label shows the actual condition.

	Assembly Line	*Machine Tool*	*Combined Workplaces*	*Neutral*	*Robot*	*Shelf*	*Table*	*Terminal*	*Too Close*	*Empty*
Assembly Line	**0.77**	0.07	0.00	0.00	0.03	0.07	0.02	0.03	0.00	0.01
Machine Tool	**0.16**	**0.46**	0.02	0.00	0.05	0.03	0.09	**0.13**	0.06	0.00
Combined Workplaces	0.00	0.00	**0.98**	0.00	0.00	0.02	0.00	0.00	0.00	0.00
Neutral	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
Robot	0.04	0.08	0.00	0.00	**0.81**	0.02	0.01	0.00	0.00	0.03
Shelf	0.08	0.07	0.00	0.00	0.00	**0.78**	0.02	0.00	0.03	0.02
Table	0.05	**0.22**	0.05	0.00	0.02	0.00	**0.40**	**0.18**	0.09	0.00
Terminal	0.01	0.07	0.00	0.00	0.00	0.00	0.07	**0.82**	0.03	0.00
Too Close	0.09	0.04	0.07	0.00	0.01	**0.10**	**0.11**	0.00	**0.56**	0.01
Empty	0.07	0.00	0.00	0.00	0.07	**0.21**	0.00	0.00	0.00	**0.64**

4.3.6 Experiment

Since the developed NN has been tested so far only with the validation dataset, the use case was carried out to test the basic functionalities of the system within a real environment. Therefore, the trained model was applied within the TU Wien pilot factory to show the effectiveness of class recognition and the capability of RTA for the assistance system middleware. The pilot factory is a typical assembly factory environment with different workplaces, which are limited to indoor scenes. Furthermore, the objects and environment in the Pilot factory should not be seen as static. This means that the process should be functional, although minor changes are occurring in the scenes. These minor changes can be illustrated by appearing and disappearing small objects (*e.g.,* working tools, etc.) in the scene. The developed and trained model is deployed on a HMD device (*i.e., Microsoft HoloLens 1*) and is tested for classification accuracy. Since the HMD is worn by the human worker, the images, fed into the algorithm, are indirectly decided by the movement of the human worker. The visual field of the user is captured in random time steps. Therefore, basically any image within the pilot factory can be classified. In the following, the experimental chapter includes the description of the HMD, the deployment of the model on the device, and the results of the use case.

AR Device

In our use case, *Microsoft HoloLens 1* was chosen as the HMD which is classified as mixed reality (MR) device. The *Microsoft HoloLens 1* contains a visor with all the necessary sensors

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.
TU Bibliothek Your knowledge hub WIEN

and cameras to detect the receptive field of the user (see Fig. 4.6a). Additionally, this visor includes holographic lenses, which allow displaying holograms to the user. Furthermore, its cameras can take photos of up to 2MP[3] and the device can process this environment based on retrieved images from timeframes with its integrated processor. This processor (see Fig. 4.6b) has an Intel 32-bit architecture[3] with 64 GB Flash[3] and 2 GB RAM[3]. Lastly, the battery capacity of the device allows 2-3 hours[3] of active operation. The HoloLens has been in the long-therm service state since 2018[3] and will not receive further updates. The latest version includes the Windows SDK 1809[3], which provides libraries and tools to build Windows applications. Within this latest software, one of the most severe limitations is that applications are only supported by being deployed to HoloLens 1 via the Universal Windows Platform (UWP) [67].

(a) (b)

Figure 4.6: *Microsoft HoloLens 1* pictures of the (a) visor, integrated with its cameras and sensors, and (b) of its built-in processor.[3]

NN Deployement on the AR Device

The trivial approach to connect the trained network and the HoloLens 1 is to run the network on an external server and feed it with the retrieved images from the HMD. However, this approach has several downsides. Within assembly factories, the density of equipments with wireless connections is high. Although these connections can be guaranteed within a factory, a stable connection with low latency might not always be created. Furthermore, connecting the HoloLens 1 to a web-enabled network can make it more susceptible to data abuse. Therefore, a standalone application might be the way to go. In addition to the trivial approach, there exists the possibility of directly processing the data on the device [67]. Here is an approach described on how to deploy the software on the device (see Fig. 4.7).

[3]https://docs.microsoft.com/en-us/hololens/hololens1-hardware, last accessed 24.08.2022

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.
TU Bibliothek Your knowledge hub WIEN

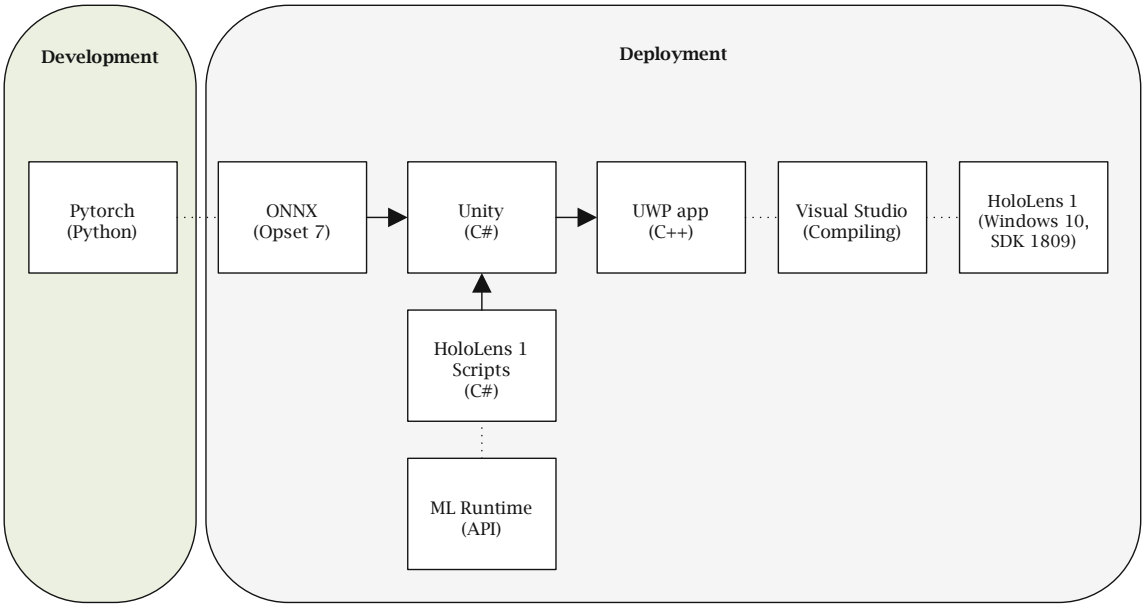


Figure 4.7: Schematic representation of the NN deployment on the *Microsoft HoloLens 1*.

Since the entire model was developed with the Python programming language by the *Pytorch* neural network library, it has to somehow be interchangable with the device that should perform the classification. Therefore, Open Neural Network Exchange (ONNX) is ideal for allowing NN models to be converted to a format that can be applied in different programming languages. Through this conversion, only the model with its parameters is exported. Training has to be done before, within the Python framework. In detail, the developed NN in *Pytorch* was exported to ONNX with opset version 7, to support the latest SDK 1809 for HoloLens 1. After conversion of the NN to an exchangable format, it has to be integrated within an UWP application, which is the only possibility to directly deploy the network on the device. The Universal Windows Platform (UWP) is a Windows runtime that allows to execute applications under Windows 10 and Windows 11. Within an UWP app, it is possible to integrate specific APIs for control of the HoloLens 1. To create the UWP application, the game development system *Unity*, programmed in C#, was used. *Unity* is an engine for 3D and 2D game development. It has support for MR, including the HoloLens. *Unity* applications consist of scenes that contain 3D elements and the game logic. These scenes are created with *Unity Editor*, which can be displayed to the user in the form of holograms. The game logic can be implemented with C# scripts, which allow camera frame extraction, feeding the model, and converting its output. Within these scripts, the most important library is *Windows Machine Learning* (WinML), which applications can run standalone on the HoloLens. WinML is a machine learning interference API, which originates from the *Microsoft AI* framework. It was developed on Windows 10 by Microsoft and is part of the standard Windows 10 SDK. Finally, after creating the UWP app in *Unity*, Visual Studio

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.

was required to build the app and release it to the Hololens. The entire UWP application is created and modified based on the work of Rene Schulte[4].

It is noteworthy that there are multiple approaches to deploy NNs to the HoloLens 1. Other approaches include *Barracuda* or *TensorFlow* [67]. For example, *Barracuda* is a NN interference library for the HoloLens 1 and supports GPU processing. However, the UWP approach is the only one for directly deploying the model on the device. Unfortunately, this library only supports CPU processing, which dramatically increases the processing time. However, the benefit compared to *Barracuda* is that it can run on all Windows 10 devices and *Barracuda* can only be used in the *Unity* environment.

Experimental Outcome

After deployment of the UWP app on the *Microsoft HoloLens 1*, its functionality was tested within the pilot factory. Special care was taken in the interpretability of the workplace class, recognition of the workplace classes, the special classes with its transitions, and finally the classification speed of one image cycle.

Workplace Classes Instructions for the user can be given with the HoloLens 1 capability to generate holograms and display them to the user. For this use-case, these holograms can be any digital object. In this case, the digital object is a text with the two most appropriate classes and their related classification accuracy. Additionally, text-to-speech software was used to give sound information (*i.e.,* class with the highest accuracy) to the user. By moving through the pilot factory, it is possible to classify scenes of the workplace classes when certain criteria are met. Fast movement with HoloLens 1 sometimes leads to misclassification due to disproportionately poor camera focussing. Furthermore, the correct position in the workplace is important for classification. It remains that misclassification can occur when looking at an identical workplace. This might result from an unintentional internal processing of the device and should be identified in future work.

Special Classes In general, four typical classification challenges occurred when observing the environment from the worker's perspective using HoloLens 1 within the experiment. First, for scenes where no workplace is recongised, but the user is within the application area, the special class *Empty* should be detected. This class usually represents large and empty factory areas. However, it was not always easy for the system to recognise, since this empty area consists mainly of low-level features that overlap with the *Too Close* class.

Next, the class *Too Close* worked well, as approaching too close to a workplace resulted in indistinguishable low-level features and the important high-level features to recognise the

[4]https://kodierer.blogspot.com/2018/06/content-for-unite-berlin-autotech.html, last accessed 24.10.2022

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.
TU Bibliothek Your knowledge hub WIEN

workplace were no longer represented. Therefore, once the user is located too close to a workplace, guidance information can be provided. Moreover, this class was also consistently recognised by looking at the floor.

Furthermore, since it is possible for the human worker to be in a position to look at two or more workplaces, adequate feedback should be given without misclassification. This special class *Combined Workplaces* was barely recognised. However, since this class was built by aligning three workplace class images into one image, misclassification could occur depending on the workplaces captured by the device.

Lastly, the class *Neutral* was generated to cover all images that do not exhibit the features of any of the other nine classes. This class was supposed to act as a safety mechanism to avoid incorrect classification of images outside the application area. However, since our application area consists of scenes and not only images with objects, the inheritent features were too diverse to avoid classification outside the application area and similar features to our classes always got detected outside the application area. Therefore, during the use case it was not possible to detect the *Neutral* class.

Intereference Speed In general, the retrieval time of images from AR glasses should be chosen wisely. The interference speed is evaluated based on how many images can be processed per second, called frames per second (FPS) [64, 66]. For this use case, the interference speed of a classification cycle is around 0.25 FPS. This loe interference speed is due to the network architecture with around 22 Mio. weights and most likely on the fact that the UWP application with the WinML library only supports CPU processing. Approaches like the *Barracuda* one are faster since they can use GPU resources. However, these approaches are limited to the Unity environment and cannot be established with UWP apps and therefore cannot be deployed to the *Microsoft Hololens 1*.

Explainable AI Lastly, to test an edge case and to interpret the way the model behaves, explainable AI (XAI) was used. The purpose of XAI is to make neural networks more transparent and explainable. For the XAI analysis the gradient-weighted class activation mapping (Grad-CAM) was applied. This is an approach in which any desired layer or block of a class can be selected. The technique produces from the gradient information of chosen layers or blocks a localisation map, which highlights the important regions for predictions of a class. For the developed network, the last convolution block was chosen to interpret the high-level features of the network predictions [68].

Figure 4.8 shows the edge case, where scenes are recorded from the TU Wien pilot factory and represent typical workplaces from the self-developed dataset. These images were initially fed into the neural network to calculate its prediction. The centre images highlight the highest prediction accuracies and their related classes. Right beside the images are the

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.
TU Bibliothek Your knowledge hub WIEN

corresponding results of XAI. The image on the left could be classified with an accuracy of 95% as *Machine Tool* and the XAI approach, on the left, highlights the area that is important for the prediction. Furthermore, from this workplace class (*i.e., Machine Tool*), three close-up areas are extracted. Therefore, the area directly on the machine (*i.e.,* top image) and the one on the floor (*i.e.,* bottom image) show low-level characteristics and are classified as *Too Close*. This illustrates that after the classification of the *Machine Tool* class, the work instructions can remain for this class, if the special class *Too Close* is recognised. However, if looking at the close-up appearance of the right centre image, the scene classification approach recognises the class *Terminal*. This means that workplaces can be recognised within workplaces, and once the *Terminal* class is recognised, work instructions can change.

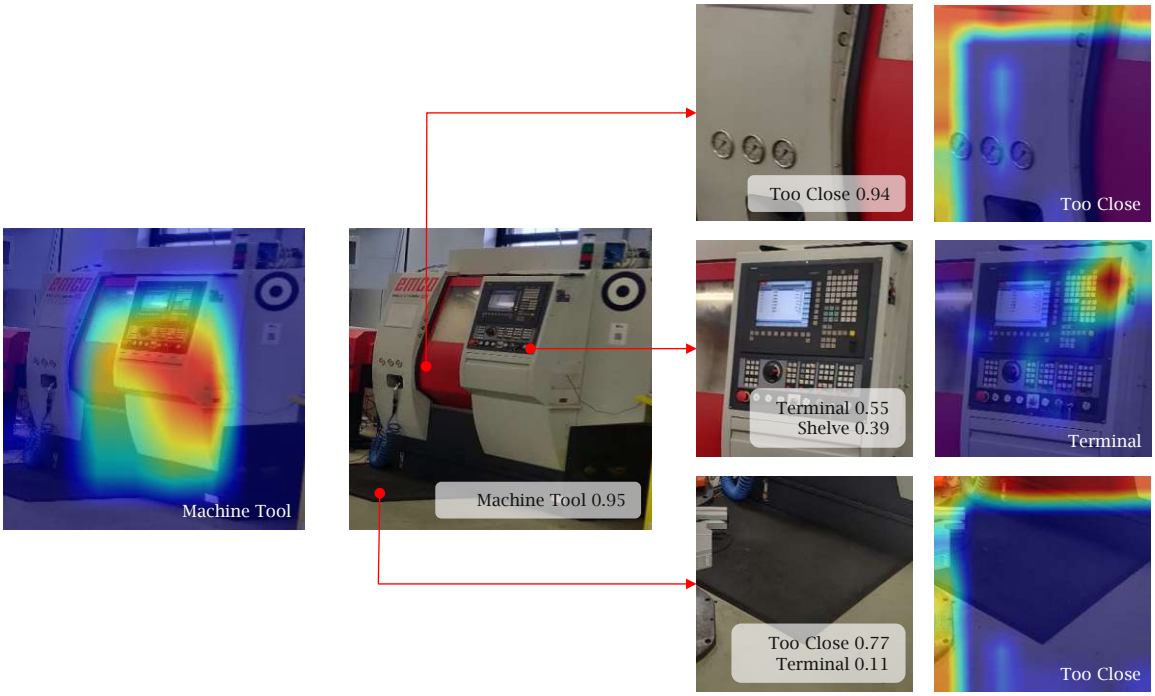


Figure 4.8: Illustration of an edge case with scenes from the TU Wien pilot factory. The three images next to the class *Machine Tool* (*i.e.,* at the right) show close-up appearences from this class. While the four images in the center show the class prediction by the network, the outer images represent class activation maps to interpret how the network predicts the class with the highest prediction accuracy.

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.

Chapter 5

Discussion

This chapter should include answers to the research questions that will be discussed in the following sections. Starting with "*How can workplace recognition enhance context-awareness of assistance systems?*". Then moving over to "*Is there any available dataset for workplace scene classification and what is the most promising scene classification approach?*", by dealing with the developed dataset and the developed neural network. Lastly, the question "*How can a deep learning algorithm be deployed on a HMD device and how does it behave within the TU Wien pilot factory?*" should be treated.

Context awareness in assistance systems to support industrial workers

Context awareness has the ability to recognise work environments within assembly factories based on significant characteristics. For a digital assistance system, it can be enabled by giving it the capability to retrieve the surroundings of the human worker in the form of images. Moreover, it must have the ability to interpret these images by their features and classify them into a possible workplace. Therefore, without monitoring the human worker, the system can retrieve the visual field of the industrial worker. Hence, context-based digital assistance systems might be the result of alleviating the cognitive demand of the human worker. To establish such a context-sensitive assistance system, HMD devices, such as *Microsoft HoloLens 1,* are the perfect choice.

Such a context-sensitive device can be especially useful for training new workers or for people with a low cognitive load limit. Through a context-sensitive device, the right instructions can be displayed in the right place without requiring any user action. Displaying instructions on HMD devices can improve task performance and reduce task load [18]. Therefore, the entrance barrier for certain jobs can be reduced. However, reducing the cognitive demand of the human worker with a new technology is a conflicting topic, as its novelty can already increase stress. Therefore, further investigations are required to validate this hypothesis.

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.
TU Bibliothek Your knowledge hub WIEN

The increasing shift from operational to more monitoring and controlling tasks can increase the demand for Multi-Machine Operations. There are already applications that use a mobile device (*e.g.,* a tablet or a smartphone) as an instruction system for the human worker. However, these systems may not guarantee the same efficiency as through HMD devices, as hands-free operations are not possible and head-movement might be reduced by HMD devices. Within this thesis, Multi-Machine Operations was addressed to differentiate between entire workplaces. Another possible extension might be the combination of the current workplace scene recognition with human activity recognition to gather more context of the application area. Such an application enables to differentiate betweem activities within an identified specific workplace [18].

Generally, work instructions should be adapted according to the cognitive needs of the human worker and should be kept as simple as possible. Therefore, continuously presenting task instructions to the human worker can have a negative impact on performance. Furthermore, HMDs are difficult to work with and by working with centrally placed instructions, the industrial worker may not be able to interact with the work objects due to holographic obstacles. When information is always displayed in the centre of the device, the picking time of a part and its placement at assembly positions might be decreased. Therefore, a context-aware HMD device can differentiate regions of the visual field and place instructions on regions of low interest is desired.

Importance of the diversity and richness of the *Assembly Factory Workplaces* datset and its limitations

The aim of this thesis, to gather context awareness by classifying workplace images, is only possible by using indoor scenes of assembly factories for training. SOTA research was conducted to find the most suitable datasets to best fit the requirements of the application. In this regard, it is important to know that workplaces do not exhibit only one specific object for their classification. Moreover, the entire scene is important, including the foreground objects and background details. For this reason, suitable classes of the public scene dataset *Places 365* were chosen to pretrain the model. However, the selected classes are only a fraction of the diversity and richness of possible workplaces within an assembly factory. A dataset that satisfies these criteria does not exist as of now. Therefore, within this thesis, a new dataset was introduced, *Assembly Factory Workplaces*, which was used to finally train the model for the desired application.

The dataset is limited to indoor scenes, because most assembly factories are within a building and not outside. Limiting training to a specific genre allows us to improve the classification performance by restricting the number of classes. For example, the *Places 365* dataset has

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.
TU Bibliothek Your knowledge hub WIEN

434 different classes[1], and the latest classification algorithms can only achieve accuracies of up to 60% [56] for the entire dataset. Therefore, the truncation of the dataset to include only 10 similar classes to our dataset helped to learn only relevant features to our dataset through pretraining. Furthermore, since images are captured randomly with the HMD device, every possible scene within the factory can be classified. Of course, unintentional scenes will occur and misclassification cannot be avoided, but special classes were introduced to absorb unintentional behaviour of the classifier. For example, if the floor or ceiling of the factory are specific classes within the dataset, predictions of those can be included. Assuming a situation, a human worker is located at a specific workstation and constantly engaged in one work task. The classification system will continuously recognise the desired class. However, if the human worker is briefly distracted by looking at the ceiling or floor, the system will classify these scenes within a workplace class if the entire dataset is unaware of special classes. Therefore, special classes *Neutral*, *Combined Workplaces*, *Empty*, and *Too Close* are introduced, whose classification can be treated differently from standard workplace scenes. For example, if a special class is detected, the previous workplace can still be in focus. Therefore, temporal change can be used more effectively, improving user experience.

Blurry boundaries between workplaces make it difficult to identify transitions or even unique workplaces. In general, workplaces within assembly factories are not strictly separated from each other. Moreover, workplace scene classes can be superimposed or even not visible at all within one scene. There is no clear structure of the workplaces within the assembly factory, so basically every position must be respected. Therefore, if using sequential information more properly (*e.g.,* through dynamic scene recognition, etc.) the classification of blurry boundaries might be increased.

The represented dataset contains the richness and diversity of all workplace classes. Therefore, the generated dataset should act as a strong baseline for assembly factory applications. Future adaptations are possible to define the workplaces more precisely and to extend the classes by allocation. In general, all these changes are possible since the developed learning approach does not have a limit on detectable classes. To expand applications across various factory domains, workplaces within other factories have to be identified and added to the model. It should be kept in mind that adding new classes increases the functionality in other domains but could potentially lower the total classification accuracy. Therefore, it is recommended to always validate the functionality of the system after such changes. So far, the developed model cannot differ between two workplaces of the same class. Therefore, the dataset facilitates just one specific workplace occurrence for a human worker. For example, the *Terminal* class can be recognised, but the differentiation between two different *Terminal* workplace classes is not implemented. This might be future work to enable the identification of different workplaces of the same class.

[1] http://places2.csail.mit.edu/, last accessed on 20.08.2022

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.
TU Bibliothek Your knowledge hub WIEN

Static scene recognition to unlock context awareness for digitial assistance systems

The supervised learning approach makes it possible to train the model in different images, improving classification performance in unseen workplaces. As mentioned through SOTA analysis, there exist different approaches to develop NN models for scene recognition. On the one hand, there are object-centric approaches, which try to identify unique objects inside an image for classification. As already stated, an object is not a unique identification of a workplace scene. Therefore, it is not recommended to use these approaches exclusively. A hybrid-based approach (*e.g.,* object localisation and classification [65] or a combination of CNN and HOI [58]) can extract more detailed features, but these approaches usually take longer for processing. Therefore, we initially started with a lean model design with the ulterior motive that processing such a model on a deployed device will take even longer. For this reason, it was decided decided to use a single-stage scene recognition approach, to meet the requirements of RTA. The adapted *ResNet50* architecture was chosen because the study in [56] has shown that it shows good performance on the *Places 365* dataset, which shows classes similar to our desired application. Further adaptations to the model were made to implement the latest SOTA strategies [53] for model training to increase overall classification performance.

Another interesting research area for future work involves models based on dynamic scene recognition. These models use not only the spatial information of the images but also the temporal information through changes between scenes. CNN-LSTM models have already shown excellent results compared to static scene recognition for human activity recognition or for natural scene recognition (*e.g.,* waterfalls, landslides, etc.). However, there is an uncertainty that arises when looking at dynamic scene recognition for workplace assembly scenes with HMD devices. Most workplace classes do not have inherent movement, such as waterfalls or landslides. Therefore, the benefit of dynamic scene recognition might be more stability for the classification of the special classes or transitions through temporal information. However, if no meaningful temporal information can be gathered, the trade-off between increased classification accuracy and classification speed might not be worth it. Therefore, temporal information can also be artificially generated by static scene recognition by averaging the last classification results.

Lastly, learning aptitude is still an open task. Currently, it is not supported that the model is self-learning through its application. Therefore, a validation procedure with the human worker can be implemented. During the usage of the application, if the human worker can provide feedback on the wrong classification of the device, further improvements can be implemented. This can be beneficial for specific real-world applications to increase the accuracy of classification.

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.
TU Bibliothek Your knowledge hub WIEN

Testing the Model Behaviour deployed on a HMD Device within an Assembly Pilot Factory

The use case focusses on the application within the TU Wien pilot factory, where the workplace classes are represented. For the use case, *Microsoft HoloLens 1* was chosen as HMD device. This use case validates the assistance system module in terms of usability. Therefore, the detection of different workplace classes, special classes, and intereference speed was observed.

This use case should be an attempt to test context sensitivity of a system developed on a HMD device within an assembly factory. Therefore, some critical appearances could be observed. On the one hand, the classification time of 4000 ms is certainly too long. However, the user has to wait too long for contextual feedback adaption. Through the observations of the use case, a classification time of around 1000 ms should be attempted to provide at least a continuous flow of recognition and bridge misclassification results. For future research this suggests to try an approach where a HMD device supports GPU processing (*e..g, Barracuda*), as this might be the most promising adaption. Another adaption might be to change the model architecture. As model architecture, a more lean CNN model can be tried to increase the classification speed at low costs of classification performance. For example, the number of residual blocks of the ResNet architecture can be reduced to increase the classification speed. Or, although *AlexNet* was not chosen for this application, it might be worthwhile to test the improved version of Xiao et al. [49], which has fewer parameters and can show a faster classification speed. Moreover, some adjustment parameters might be beneficial to individually control the number of FPS that are processed by the device or to regulate the sensitivity of the classification (*i.e.,* averaging strategy to include timeframes of the previous classification outputs, which will be considered for future decision-making). Without any user-related settings, the overall usability of the assistance system might drop dramatically.

Although the human worker can be supported by the *Microsoft HoloLens 1*, the device can restrict the visual field of the human worker. This limited total field of view can result in negative exposure for the industrial worker, since the accomplishment of the working tasks might be hindered. The instructions displayed on the AR device might occlude important areas. Optical see-through smart glasses, such as *Google Glasses*, typically have a smaller field of view for AR, do not impair as strongly and are more lightweight [18]. In general, the user should only be immersed in the virtual world as much as necessary. If no holograms are required and only the text displayed is sufficient, smart glasses (*e.g., Google Glasses*) might be beneficial. Therefore, the device should be chosen depending on the output type that needs to be displayed. Although the HoloLens is a supererogation for just displaying text, it keeps the information output flexible. Nevertheless, it is recommended to test other

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.
TU Bibliothek Your knowledge hub WIEN

AR devices, such as *Microsoft HoloLens 2* for faster processing or *Google Glasses* with less *VR* immersion.

In general, the system cannot decide to capture only workplace classes. Therefore, the entire application area has to be covered and should also include the classification of non-workplace classes to increase overall performance of the device. However, by limiting the application area more strictly, classifying non-relevant information can be avoided. In addition, the system can differentiate between different workplaces, but it cannot identify worktasks within workplaces. Thus, multiple task-specific instructions for one workplace cannot be differentiated based on context. Lastly, the system does not capture and save the images that have been processed. Therefore, it is unknown which images eventually got classified and whether HMD still shows some drop in classification performance compared to digital validation.

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.

Chapter 6

Conclusion

To sum up, this thesis is a proof-of-concept to showcase that context-sensitivity in terms of location or scene-awareness can be implemented in a HMD device to help human workers in assembly factories. Therefore, the benefits (*i.e.,* enabling automated AR-instructions within multi-machine operation, supporting the decision-making process, and removing the responsibility of the human worker) and drawbacks (*i.e.,* unfamiliarity with the system, reduced field of view, and split attention) of an guidance system with context awareness were mentioned. Through the latest research on supervised learning algorithms for indoor scenes, important key aspects of neural networks and datasets for training could be identified. Thus, it has been determined that the application within an assembly factory can be best represented by indoor scene classes. Therefore, a newly developed dataset, *Assembly Factory Workplaces*, with around 4695 images, 6 workplace classes and 4 special classes classes, was created. Furthermore, as a neural network, a modified *ResNet-50* architecture was introduced. To increase the robustness and performance of the model, pretraining was carried out on a truncated *Places365* dataset. To finally validate the test results, a use case was conducted within the TU Wien pilot factory. Therefore, the trained neural network was deployed on a HMD device (*i.e., Microsoft Hololens 1*). Since the proposed system is only a middleware to support context-awareness in multi-machine operation, the use case focused on identifying the classification speed (*i.e.,* around 4000 ms) and the identification performance of the classes, which was sufficient for most scene classes (*i.e., Terminal*, *Too Close*, *Shelf*, *Robot*, *Assembly Line*) within the assembly factory. Moreover, some classes (*i.e., Table*, *Combined Workplaces*, *Empty*, *Neutral*, *Machine Tool*) could not be classified properly which suggests further research. Overall, this thesis provides a proof of concept that shows that scene-awareness for assistance systems is achievable. However, there are still open topics that need to be further examined. On the one hand, the posistive effects of these AR-based instructions in multi-machine operations need to be proven by experimental studies with workers. Furthermore, other neural networks (*i.e.,* espically CNN-LSTM or Object Detection with CNN), a more diverse and rich datasets and other HMD devices or smart glasses, might be worthwile, to improve the functionality of context-awareness.

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.
TU Bibliothek Your knowledge hub WIEN

List of Figures

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.
TU Bibliothek Your knowledge hub WIEN

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.
TU Bibliothek Your knowledge hub WIEN

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.

List of Tables

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.

Bibliography

[1] D. P. Perales, F. A. Valero, and A. B. García, "Industry 4.0: A Classification Scheme," in *Closing the Gap Between Practice and Research in Industrial Engineering*, ser. Lecture Notes in Management and Industrial Engineering, E. Viles, M. Ormazábal, and A. Lleó, Eds. Cham: Springer International Publishing, 2018, pp. 343–350.

[2] F. Longo, L. Nicoletti, and A. Padovano, "Smart operators in industry 4.0: A human-centered approach to enhance operators' capabilities and competencies within the new smart factory context," *Computers & Industrial Engineering*, vol. 113, pp. 144–159, 2017.

[3] T. Keller, C. Bayer, P. Bausch, and J. Metternich, "Benefit evaluation of digital assistance systems for assembly workstations," *Procedia CIRP*, vol. 81, pp. 441–446, 2019.

[4] M. Fellmann, S. Robert, S. Büttner, H. Mucha, and C. Röcker, "Towards a Framework for Assistance Systems to Support Work Processes in Smart Factories," in *Machine Learning and Knowledge Extraction*, ser. Lecture Notes in Computer Science, A. Holzinger, P. Kieseberg, A. M. Tjoa, and E. Weippl, Eds. Cham: Springer International Publishing, 2017, vol. 10410, pp. 59–68.

[5] B. G. Mark, L. Gualtieri, E. Rauch, R. Rojas, D. Buakum, and D. T. Matt, "Analysis of User Groups for Assistance Systems in Production 4.0," in *2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*. IEEE, 2019, pp. 1260–1264.

[6] M. Funk, T. Dingler, J. Cooper, and A. Schmidt, "Stop helping me - I'm bored!" in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers - UbiComp '15*, K. Mase, M. Langheinrich, D. Gatica-Perez, H. Gellersen, T. Choudhury, and K. Yatani, Eds. New York, New York, USA: ACM Press, 2015, pp. 1269–1273.

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.

[7] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, "Smart Factory of Industry 4.0: Key Technologies, Application Case, and Challenges," *IEEE Access*, vol. 6, pp. 6505–6519, 2018.

[8] N. Padoy, "Machine and deep learning for workflow recognition during surgery," *Minimally invasive therapy & allied technologies : MITAT : official journal of the Society for Minimally Invasive Therapy*, vol. 28, no. 2, pp. 82–90, 2019.

[9] J. Mathieu and S. Lehmann, *Eigenarten der industriellen Mehrstellenarbeit*, ser. Forschungsberichte des Landes Nordrhein-Westfalen. Wiesbaden and s.l.: VS Verlag für Sozialwissenschaften, 1963, vol. 1180.

[10] Z. Akkus, J. Cai, A. Boonrod, A. Zeinoddini, A. D. Weston, K. A. Philbrick, and B. J. Erickson, "A Survey of Deep-Learning Applications in Ultrasound: Artificial Intelligence-Powered Ultrasound for Improving Clinical Workflow," *Journal of the American College of Radiology : JACR*, vol. 16, no. 9 Pt B, pp. 1318–1328, 2019.

[11] S. Richardson, "Affective computing in the modern workplace," *Business Information Review*, vol. 37, no. 2, pp. 78–85, 2020.

[12] A. R. Pathak, M. Pandey, and S. Rautaray, "Application of Deep Learning for Object Detection," *Procedia Computer Science*, vol. 132, pp. 1706–1717, 2018.

[13] D. Lucke, C. Constantinescu, and E. Westkämper, "Smart Factory - A Step towards the Next Generation of Manufacturing," in *Manufacturing Systems and Technologies for the New Frontier*, M. Mitsuishi, K. Ueda, and F. Kimura, Eds. London: Springer London, 2008, pp. 115–118.

[14] Y.-R. Shiue, K.-C. Lee, and C.-T. Su, "Real-time scheduling for a smart factory using a reinforcement learning approach," *Computers & Industrial Engineering*, vol. 125, pp. 604–614, 2018.

[15] T. Kotsiopoulos, P. Sarigiannidis, D. Ioannidis, and D. Tzovaras, "Machine Learning and Deep Learning in smart manufacturing: The Smart Grid paradigm," *Computer Science Review*, vol. 40, p. 100341, 2021.

[16] J. Lee, B. Bagheri, and H.-A. Kao, "A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems," *Manufacturing Letters*, vol. 3, pp. 18–23, 2015.

[17] B. G. Mark, E. Rauch, and D. T. Matt, "Worker assistance systems in manufacturing: A review of the state of the art and future directions," *Journal of Manufacturing Systems*, vol. 59, pp. 228–250, 2021.

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.

[18] P. Renner, "Prompting Techniques for Guidance and Action Assistance Using Augmented-Reality Smart-Glasses," in *25th IEEE Conference on Virtual Reality and 3D User Interfaces*, K. Kiyokawa, Ed. Piscataway, NJ: IEEE, 2018, pp. 820–822.

[19] A. Khurshid, S. Cleger, and R. Grunitzki, "A Scene Classification Approach for Augmented Reality Devices," in *HCI International 2020 – Late Breaking Papers: Virtual and Augmented Reality*, ser. Lecture Notes in Computer Science, C. Stephanidis, J. Y. C. Chen, and G. Fragomeni, Eds. Cham: Springer International Publishing, 2020, vol. 12428, pp. 164–177.

[20] S. Dutta, "An overview on the evolution and adoption of deep learning applications used in the industry," *WIREs Data Mining and Knowledge Discovery*, vol. 8, no. 4, 2018.

[21] A. Schmidt, "Implicit human computer interaction through context," *Personal Technologies*, vol. 4, no. 2-3, pp. 191–199, 2000.

[22] D. Kuhner, L. Fiederer, J. Aldinger, F. Burget, M. Völker, R. T. Schirrmeister, C. Do, J. Boedecker, B. Nebel, T. Ball, and W. Burgard, "A service assistant combining autonomous robotics, flexible goal formulation, and deep-learning-based brain–computer interfacing," *Robotics and Autonomous Systems*, vol. 116, pp. 98–113, 2019.

[23] A. K. Dey, G. D. Abowd, and D. Salber, "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications," *Human–Computer Interaction*, vol. 16, no. 2-4, pp. 97–166, 2001.

[24] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, ser. Adaptive computation and machine learning. Cambridge, Massachusetts and London, England: The MIT Press, 2016.

[25] A. Koul, S. Ganju, and M. Kasam, *Practical deep learning for cloud, mobile, and edge: Real-world AI and computer vision projects using Python, Keras and TensorFlow*, 1st ed. Beijing and Boston and Farnham and Sebastopol and Tokyo: O'Reilly, October 2019.

[26] J. Brownlee, *Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python*. Machine Learning Mastery, 2019.

[27] T. M. Mitchell, *Machine learning*, ser. McGraw-Hill series in computer science. Boston, Mass.: WCB/McGraw-Hill, 1997.

[28] S. Weidman, *Deep learning from Scratch: Building with Python from first principles*, 1st ed. Beijing and Boston and Farnham and Sebastopol and Tokyo: O'Reilly, Septemberg 2019.

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.

[29] T. Hastie, R. Tibshirani, and J. H. Friedman, *The elements of statistical learning: Data mining, inference, and prediction*, 2nd ed., ser. Springer Series in Statistics. New York, NY: Springer, 2017.

[30] H. Wu, Q. Liu, and X. Liu, "A Review on Deep Learning Approaches to Image Classification and Object Segmentation," *Computers, Materials & Continua*, vol. 60, no. 2, pp. 575–597, 2019.

[31] D. Zeng, M. Liao, M. Tavakolian, Y. Guo, B. Zhou, D. Hu, M. Pietikäinen, and L. Liu, "Deep Learning for Scene Classification: A Survey," *CoRR*, 2021. [Online]. Available: http://arxiv.org/pdf/2101.10531v2

[32] W. Ballard, *Hands-On Deep Learning for Images with TensorFlow: Build Intelligent Computer Vision Applications Using TensorFlow and Keras.* Birmingham: Packt Publishing Ltd, 2018.

[33] J. A. Yacim and D. G. B. Boshoff, "Impact of Artificial Neural Networks Training Algorithms on Accurate Prediction of Property Values," *Journal of Real Estate Research*, vol. 40, no. 3, pp. 375–418, 2018.

[34] G. L. Team, "Types of Neural Networks and Definition of Neural Network." [Online]. Available: https://www.mygreatlearning.com/blog/types-of-neural-networks/

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[36] Nitish Srivastava and Geoffrey E. Hinton and Alex Krizhevsky and Ilya Sutskever and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, no. 15, pp. 1929–1958, 2014.

[37] T. D. Akinosho, L. O. Oyedele, M. Bilal, A. O. Ajayi, M. D. Delgado, O. O. Akinade, and A. A. Ahmed, "Deep learning in the construction industry: A review of present status and future innovations," *Journal of Building Engineering*, vol. 32, p. 101827, 2020.

[38] K. Xia, J. Huang, and H. Wang, "LSTM-CNN Architecture for Human Activity Recognition," *IEEE Access*, vol. 8, pp. 56 855–56 866, 2020.

[39] S. Trinks and C. Felden, "Image Mining for Real Time Fault Detection within the Smart Factory," in *2019 IEEE 21st Conference on Business Informatics (CBI)*. IEEE, 2019, pp. 584–593.

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.
TU Bibliothek Your knowledge hub WIEN

[40] A. Matei, A. Glavan, and E. Talavera, "Deep learning for scene recognition from visual data: a survey," in *International Conference on Hybrid Artificial Intelligence Systems*, 2020, pp. 763–773.

[41] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 248–255.

[42] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 413–420.

[43] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "SUN database: Large-scale scene recognition from abbey to zoo," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 3485–3492.

[44] P. Viswanathan, T. Southey, J. Little, and A. Mackworth, "Place Classification Using Visual Object Categorization and Global Information," in *2011 Canadian Conference on Computer and Robot Vision*. IEEE, 2011, pp. 1–7.

[45] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning Deep Features for Scene Recognition using Places Database," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, Eds., vol. 27. Curran Associates, Inc, 2014.

[46] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 Million Image Database for Scene Recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 6, pp. 1452–1464, 2018.

[47] V. Singh , D. Girish , A. Ralescu, "Image Understanding-a Brief Review of Scene Classification and Recognition," in *Proceedings of the Modern Artifical Intelligence and Cognitive Science (MAIS)*, 2017, pp. 85–91. [Online]. Available: http://ceur-ws.org/vol-1964/ml2.pdf

[48] N. Jmour, S. Zayen, and A. Abdelkrim, "Convolutional neural networks for image classification," in *2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET)*. IEEE, 2018, pp. 397–402.

[49] L. Xiao, Q. Yan, and S. Deng, "Scene classification with improved AlexNet model," in *Proceedings of the 2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)*, T. Li, Ed. Piscataway, NJ: IEEE, 2017, pp. 1–6.

[50] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

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.
TU Bibliothek Your knowledge hub WIEN

[51] A. Shah and K. Rana, "Scene Classification Using Deep Learning Technique," in *Proceedings of First International Conference on Computing, Communications, and Cyber-Security (IC4S 2019)*, ser. Lecture Notes in Networks and Systems, P. K. Singh, W. Pawłowski, S. Tanwar, N. Kumar, J. J. P. C. Rodrigues, and M. S. Obaidat, Eds. Singapore: Springer Singapore, 2020, vol. 121, pp. 575–587.

[52] M. Afif, R. Ayachi, Y. Said, and M. Atri, "Deep Learning Based Application for Indoor Scene Recognition," *Neural Processing Letters*, vol. 51, no. 3, pp. 2827–2837, 2020.

[53] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of Tricks for Image Classification with Convolutional Neural Networks," in *In Proceedings of the IEEE Conference on Computer Vision and Pattern*, Computer Vision Foundation/IEEE, Ed., 2019, pp. 558–567. [Online]. Available: http://arxiv.org/pdf/1812.01187v2

[54] W. Ge and Y. Yu, "Borrowing Treasures from the Wealthy: Deep Transfer Learning through Selective Joint Fine-tuning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1086–1095.

[55] L. Xie, F. Lee, L. Liu, K. Kotani, and Q. Chen, "Scene recognition: A comprehensive survey," *Pattern Recognition*, vol. 102, p. 107205, 2020.

[56] A. López-Cifuentes, M. Escudero-Viñolo, J. Bescós, and Á. García-Martín, "Semantic-Aware Scene Recognition," *Pattern Recognition*, vol. 102, no. 6, p. 107256, 2020. [Online]. Available: http://arxiv.org/pdf/1909.02410v3

[57] J. Yao, S. Fidler, and R. Urtasun, "Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 702–709.

[58] O. Moutik, S. Tigani, R. Saadane, and A. Chehri, "Hybrid Deep Learning Vision-based Models for Human Object Interaction Detection by Knowledge Distillation," *Procedia Computer Science*, vol. 192, pp. 5093–5103, 2021.

[59] G. Yengera, D. Mutter, J. Marescaux, and N. Padoy, "Less is More: Surgical Phase Recognition with Less Annotations through Self-Supervised Pre-training of CNN-LSTM Networks," *CoRR*, 2018. [Online]. Available: http://arxiv.org/abs/1805.08569

[60] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Temporal Residual Networks for Dynamic Scene Recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 7435–7444.

[61] Y. Gu, H. Liu, T. Wang, S. Li, and G. Gao, "Deep feature extraction and motion representation for satellite video scene classification," *Science China Information Sciences*, vol. 63, no. 4, 2020.

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.
TU Bibliothek Your knowledge hub WIEN

[62] T. Zheng, C. Liu, B. Liu, M. Wang, Y. Li, P. Wang, X. Qin, and Y. Guo, "Scene Recognition Model in Underground Mines Based on CNN-LSTM and Spatial-Temporal Attention Mechanism," in *2020 International Symposium on Computer, Consumer and Control (IS3C)*. IEEE, 2020, pp. 513–516.

[63] S. Jaiswal and T. Jaiswal, "Deep Learning Approaches for Object Detection," *Artificial Intelligence Evolution*, pp. 123–145, 2020.

[64] X. Wu, D. Sahoo, and S. C. Hoi, "Recent advances in deep learning for object detection," *Neurocomputing*, vol. 396, pp. 39–64, 2020.

[65] H. Harzallah, F. Jurie, and C. Schmid, "Combining efficient object localization and image classification," in *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 237–244.

[66] S. Wang, J. Wan, Di Li, and C. Liu, "Knowledge Reasoning with Semantic Data for Real-Time Data Processing in Smart Factory," *Sensors (Basel, Switzerland)*, vol. 18, no. 2, 2018.

[67] L. Lazar, "Neural Networks on Microsoft HoloLens 2," *Universität Stuttgart*, 2021. [Online]. Available: http://dx.doi.org/10.18419/opus-11558

[68] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, 2020. [Online]. Available: http://arxiv.org/pdf/1610.02391v4

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.