FAKULTÄT FÜR !NFORMATIK

Recognizing Degraded Handwritten Characters

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Computergraphik & Digitale Bildverarbeitung (Visual Computing)

eingereicht von

Markus Diem
Matrikelnummer 0226595

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuer: Univ. Prof. Dr. Robert Sablatnig

Wien, 16.02.2010

(Unterschrift Verfasser) (Unterschrift Betreuer)

Markus Diem, Michelstr. 2, 6850 Dornbirn

“Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.”

Wien, 16.02.2010 (Unterschrift)

Danksagung

An dieser Stelle möchte ich mich bei all jenen Menschen, die mich in meinem Leben begleitet und unterstützt haben, bedanken. Sie haben wesentlich dazu beigetragen, dass ich meinen Traum von einem Hochschulstudium und diese Diplomarbeit verwirklichen konnte.

Ein Dank für die fachliche Unterstützung und die interessanten Diskussionen gilt meinen Arbeitskollegen am PRIP. Speziell möchte ich mich bei Phillip Blauensteiner, Angelika Garz, Adrian Ion, Florian Kleber, Martin Lettner, Julian Stöttinger, Maria Vill und Sebastian Zambanini bedanken. Sie haben, unter anderem, mein Interesse an der Bildverarbeitung gefördert und den Büroalltag spannend gemacht.

Ein spezieller Dank gilt auch meinem Betreuer Robert Sablatnig, der mich im 4. Semester im Rahmen einer Übung für die Bildverarbeitung begeistern konnte. Weiters ließ er mich einen Teil dieser Arbeit im Rahmen des FWF-Projekts *"The Sinaitic Glagolitic Sacramentary (Euchologium) Fragments"* (FWF: P19608-G12) durchführen und hat mir Veröffentlichungen in internationalen wissenschaftlichen Konferenzen ermöglicht.

Diese Diplomarbeit widme ich meinen Eltern und meiner Freundin Babsi. Meine Eltern haben mich von je her unterstützt und mein Studium finanziert. Sie schenkten mir in all den Jahren Kraft und Zuversicht. Babsi, du bringst mich jeden Morgen wieder zum Lächeln.
ⰎⰊⰅⰁ ⰎⰊ

Danke

Abstract

In this thesis, a character recognition system is proposed that handles degraded manuscript documents which were discovered at the St. Catherine's Monastery. In contrast to state-of-the-art Ocr systems, no early decision, namely the image binarization, needs to be performed. Thus, an object recognition methodology is adapted for the recognition of ancient manuscripts. Therefore, interest points are extracted which allow for the computation of local descriptors. These are directly classified using a Svm with one against all tests.

In order to localize characters, interest points that represent characters are found by means of a scale distribution histogram. Then, the remaining interest points are clustered using a k-means which is initialized with the previously selected interest points. Finally a voting scheme is applied where the local descriptors' class probabilities are accumulated to a probability histogram for each character cluster. This histogram does not solely allow for a hard decision, but can be presented to human experts who can decide the character class for hardly readable characters according to the probabilities obtained.

The system was evaluated on three different datasets, namely a synthetic with Latin script, degraded characters and real world data. The system achieves a $F_{0.5}$ score of 0.77 on the last dataset mentioned.

Kurzfassung

In dieser Diplomarbeit wird ein neues Character Recognition System für schlecht erhaltene Manuskripte vorgestellt. Im Gegensatz zu aktuellen OCR Systemen, welche Information durch eine frühzeitige Binarisierung verwerfen, wird eine Methodik implementiert, die sich an aktuellen Objekterkennungs-Algorithmen orientiert. Um die Bildinformation aufzubereiten, werden Interest Points berechnet, die Bildbereiche markieren, welche Struktur enthalten. Mit Hilfe von Interest Points können dann lokale Deskriptoren, sozusagen hochdimensionale Feature Vektoren, berechnet werden. Eine SVM klassifiziert die lokalen Deskriptoren.

Mit dieser Methodik werden auch ausgebleichte Buchstaben erkannt. Die Lokalisierung der Buchstaben muss aufgrund der nicht durchgeführten Binarisierung durch die Interest Points realisiert werden. Dafür werden Interest Points, die ganze Buchstaben beschreiben durch ein Scale Distribution Histogram segmentiert. Diese Interest Points dienen zur Initialisierung eines k-means Clusterings, welches lokale Deskriptoren eines Buchstabens gruppiert. Für die endgültige Klassifizierung der Buchstaben werden die Wahrscheinlichkeiten aller lokaler Deskriptoren eines Clusters, welche mit der SVM bestimmt wurden, durch ein Voting Schema akkumuliert.

Das System wurde mit drei Datensätzen evaluiert: generierte lateinische Buchstaben, schlecht erhaltene glagolitische Buchstaben und Dokumentseiten des *Cod. Sin. Slav.* 5N. Auf ganzen Dokumentseiten wird ein $F_{0.5}$ score von 0.77 erreicht.

Contents

Chapter 1

Introduction

The St. Catherine's Monastery on Mount Sinai, Egypt, which is the oldest continuously existing Christian monastery in the world, features a great collection of Slavonic manuscripts containing approximately 43 Slavic codices [MGK+08]. In 1975, another 42 items were found in a bricked chamber of the monastery. This finding contains six Glagolitic codices which were written between the 10^{th} and 12^{th} century [MGK+08].

The Glagolica was created in 862 by Konstantin-Kyrill who is famous for creating the Cyrillic alphabet [Mik00]. It is based upon the Greek alphabet and is today known as Church Slavonic. The Glagolitic alphabet initially consisted of 36 characters.

The six Glagolitic codices are called *Codd. Sin. slav* 1N - 5N. They represent a monastic collection comprising liturgical genres, books of canon law, ascetic and apocryphic miscellanies [MGK+08]. While the Psalterium Demetrii (*Cod. Sin. slav.* 3N) is preserved in its entirety, other codices such as the *Cod. Sin. Slav.* 5N are partially destroyed because of bad storage conditions. In Figure 1.1 (left), a typical page from the *Cod. Sin. Slav.* 5N is illustrated. It can be seen that the parchment's border are disrupted, parts of text lines are faded out and background clutter is present. The methods discussed in this thesis are developed with respect to the *Cod. Sin. Slav.* 5N.

In September 2007, a scientific team traveled to the St. Catherine's Monastery in order to digitize the *Cod. Sin. Slav.* 5N and the *Cod. Sin. slav.* 3N. For the acquisition of the manuscripts, a Hamamatsu C9300-124 camera was used. It records images with a resolution of 4000×2672 px and a spectral response between 330 and 1000 nm. A lighting system provides the required Infra-Red (IR), VIS and Ultra-Violet (UV) illumination. In order to speed-up the acquisition process, software was developed which controls the Hamamatsu camera and the automatic filter wheel that is fixed on its object lens. Thus, the user can specify which optical filters to use and camera parameters such as exposure time. Having specified all parameters, the software takes the spectral images and stores them on the hard disk [KS08].

Low-pass, band-pass and short-pass filters are used to select specific spectral ranges. The near UV (320 nm - 440 nm) excites, in conjunction with specific inorganic and organic substances, visible fluorescence light [Mai03]. UV reflectography is used to visualize retouching, damages and changes through e.g. luminescence. Therefore the visible range of light has to be excluded in order to concentrate on the long wave UV light. This is achieved by applying short-pass filters and using exclusively UV light sources. Addition-

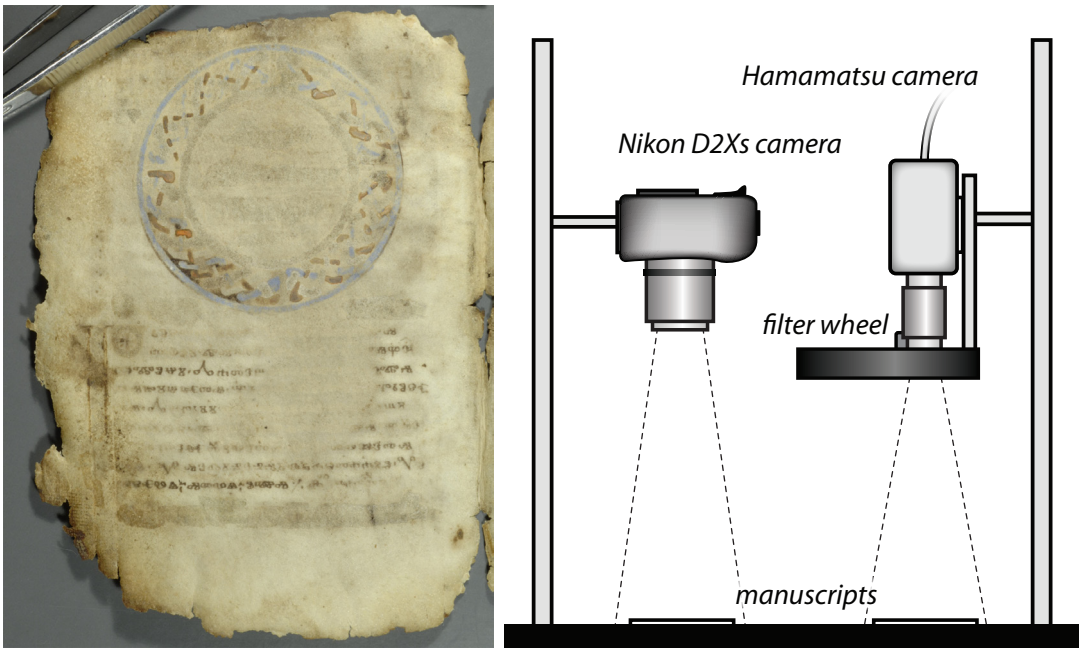

Figure 1.1: Page 20 verso (left) and the acquisition system (right) used for digitizing the manuscript pages.

ally, a RGB color image and a UV fluorescence image of each manuscript page are taken using a Nikon D2Xs camera. Figure 1.1 (right) shows the acquisition system where the Hamamatsu camera is used to capture seven spectral images (grayscale). Having acquired the spectral images, the manuscript pages need to be moved in order to capture RGB images with the Nikon camera [DLS07].

1.1 Motivation

It was illustrated in Figure 1.1 that the dataset investigated consists of ancient manuscripts which are degraded resulting from their storage conditions. The principal concept of this thesis is to develop a system that assists human experts when reading degraded manuscript pages. State-of-the-art OCR methods which are further detailed in Chapter 2 binarize images before extracting features for the recognition process. However, if the detail in Figure 1.2 (a) is considered, strokes and parts of faded-out characters are missed when applying a state-of-the-art binarization on manuscript images of the investigated dataset. As can be seen in Figure 1.2 (c), a global threshold, namely Otsu's method [Ots79], falsely detects background clutter. As a consequence of the image's low dynamic range, character holes (e.g. 𐎀 in the second row) which are useful for feature extraction[1], are not found correctly. Applying a local binarization (see Figure 1.2 (d)) improves the character extraction. However, background clutter still results in false objects. If the 𐎌 or the 𐎚 of the last text line is considered, it can be seen that even Sauvola's [SP00] method cannot extract faded-out characters correctly. These two characters are correctly

[1]Holes are topographic features of a character.

recognized with the system proposed. Figure 1.2 (b) shows the classification results of the same manuscript page when the proposed system is applied. Green characters with the corresponding character overlaid indicate correctly recognized characters. On the other hand red highlights marked with an $\times$ illustrate false classification results.

If we regard Figure 1.2, it can be seen that, despite the improvements of document binarization in the last decades, still challenging datasets exist. Intuitively, the task of image binarization is easy for human observers: mark all parts of characters and leave remaining image parts blank. A human observer who is not illiterate does not solely regard differences of gray-values but takes the document's context into account. However, if degraded manuscripts are to be considered, binarization-based on local gray values does not lead to correct results since gray scale information is ambiguous for degraded characters. Hence, the same gray values are parts of characters and background clutter. A solution for improving the binarization results is to take the image context into account. But solving the binarization using context would solve character recognition at the same time. Fischer et al. [FWL+09] call the segmentation of characters in cursive handwritings a *"chicken-and-egg"* problem, since characters can be reliably segmented, if they are recognized, but state-of-the-art recognition systems require a correct character segmentation. The same applies for ancient manuscripts if the binarization is considered.

In the last two decades, a paradigm shift – namely replacing blob features by local features[2] – took place in the object recognition community (see Chapter 2). Object recognition systems where initially similar to Ocr systems. Therefore images were binarized based on the intensity, then binary features where computed for each object present. If a bicycle or a car in a real world scene is considered, it is obvious that a binarization based on intensities cannot correctly segment these objects. That is why features such as local descriptors which are directly computed on the input signal achieved success for image retrieval and object recognition tasks [Low04, MS05]. Recently, methods were developed that localize objects by means of probabilistic models [MLS06], sliding windows [FFJS08] or sub windows [LBH09].

As previously mentioned, modern binarization techniques are not applicable for the dataset investigated. That is why a system is designed that is inspired by modern object recognition system. This allows for a late classification decision meaning that no information is initially lost owing to image binarization.

1.1.1 Scope of Discussion

This thesis focuses on character recognition for ancient manuscripts. However, a complete Ocr was not developed in this context. Thus the scientific question is: *Can state-of-the-art object recognition methods be applied for recognizing degraded characters?*

In order to answer this question, state-of-the-art interest points and local descriptors where compared on the investigated dataset being synthetically affine distorted. According to tests further detailed in Chapter 4, the best performing, namely DoG and Sift, where chosen for the feature extraction. Since all current Ocr systems binarize images,

[2] *blob features* are based on binary images, while *local features* are computed on color or grayscale images

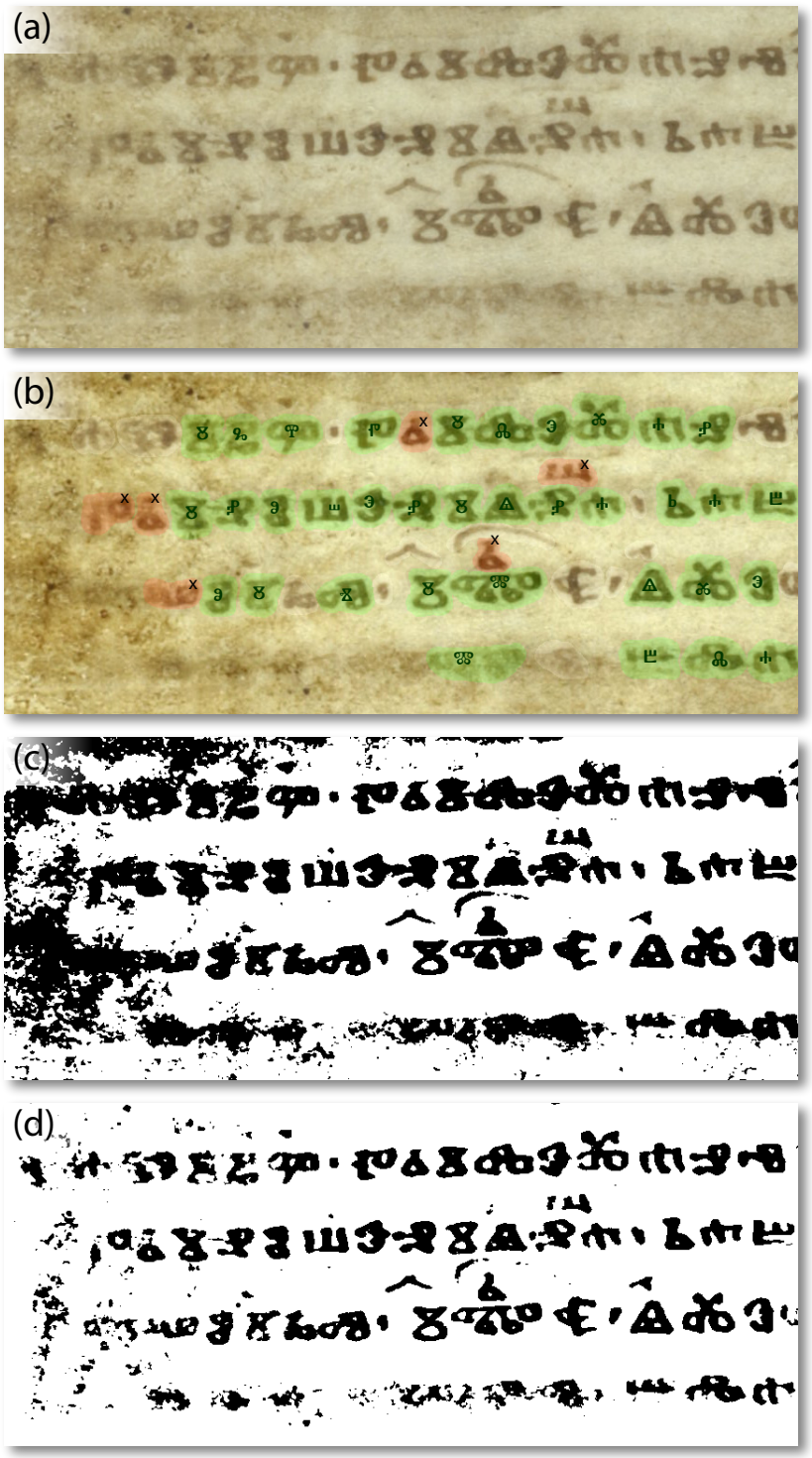

Figure 1.2: A manuscript page's detail (a), results of the system proposed (b), binarization of the page using Otsu's method (c) and the Sauvola binarization (d).

characters or words are implicitly localized. However, accurate object localization is a current issue in the object recognition community [MLS06, LBH09]. That is why a character localization based on clustering interest points was developed for the system proposed.

As previously mentioned it is not intended to build a complete OCR system. This is on the one hand because a dictionary, which is important to improve the recognition process, does not exist for the Glagolica. In addition, words are not separated by spaces in this script, which complicates the localization of words that are needed for dictionaries. On the other hand, a text does not need to be transcribed in order to evaluate a character recognition system. Hence, the proposed system is evaluated by directly groundtruthing document images.

As a consequence of the dataset investigated, the system is not compared to current state-of-the-art OCR systems. Nevertheless, the classification performance which is further discussed in Chapter 4 can be compared to results gained by current systems on degraded manuscripts which are further detailed in Chapter 2.

In addition to Glagolica, the system was evaluated on modern computer fonts in order to show its flexibility. This test additionally shows if a system based on local information is applicable for general OCR tasks.

1.1.2 Objective

Reliably recognizing characters of scanned machine printed documents is possible if current commercial OCR systems (e.g. TypeReader[3], FineReader[4], OmniPage[5], Tesseract[6]) are regarded. However, recognizing manuscripts and especially degraded manuscripts is still a challenging task which is further discussed in Chapter 2. Especially the binarization of faded-out characters in presence of background clutter is a current issue [FWL+09].

Another issue arising when manuscript characters are recognized is the class diversity. In other words, the classification task needs to differentiate in our case 36 characters which are even more for other scripts. However, the problem is not solely the number of classes, but also the characters' shapes vary according to the scribe, neighboring characters and writing materials. In addition to this, noise such as faded-out ink or mold degrades the documents which results in a challenging character recognition task. At the same time, characters such as ⰹ, ⰻ and ⰾ exist that have a similar shape. Figure 1.3 shows two different characters having a similar shape. Additionally, the character variation of one scribe is shown. The last row illustrates faded-out characters and stains present in the background.

The previously mentioned class diversity can be handled by training the system with all currently available characters. Yet, the human effort should be kept as low as possible in order to guarantee that the system can be applied to other scripts. That is why a classifier needs to be incorporated that maximizes the prediction when only few (e.g. 20) samples per character are presented to the system. An additional intention is to design a system that keeps probabilities throughout the processing. Thus, human observers are

[3]ExperVision: `http://www.expervision.com/`
[4]ABBYY: `http://www.abbyy.com/`
[5]Nuance Communications: `http://www.nuance.com/`
[6]Hewlett-Packard & Google: `http://code.google.com/p/tesseract-ocr/`

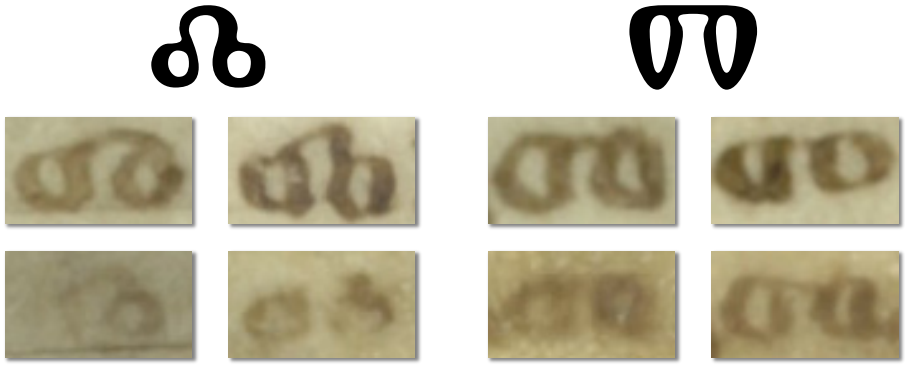

Figure 1.3: Four variations of a ⰱ$_{\text{a}}$ (left) and four variations of ⱆⱆ (right).

provided not solely character predictions but also with probabilities of a character for belonging to a given class.

If document images are not binarized, the object localization is an issue. It is known that a distinct part of the image is most probable belonging to a character (e.g. ⱍⱍ). Still it is not known if this part fits the whole character or more than the actual character. Additionally, combining the information of different local descriptors improves the final prediction. That is why a character localization method needs to be developed which is based on gray scale information so as to guarantee that faded-out characters are still recognized.

1.1.3 Main Contribution

The objective of this thesis is not to develop a complete OCR system but to discuss a case study on new methods for recognizing characters of ancient manuscripts. Thus, the main contribution of this thesis is to introduce object recognition methodologies to the character recognition community. For this purpose, a character recognition system was designed that incorporates state-of-the-art local features. An evaluation of local descriptors on ancient manuscripts is given in Chapter 3 and in [DS09].

Another issue solved in this thesis is the localization of characters without the need for binarization. This is further discussed in Chapter 3 and in [DS10]. The character localization is based on the fact that every object produces one single interest point that describes the whole object. These interest points are detected using an adaptive scale selection threshold that is computed by means of a scale distribution. Subsequently, the interest points representing characters are obtained as seed points for a k-means clustering that groups all local descriptors of a given manuscript image.

In addition to the designed system and comparison of local descriptors, the system was applied to modern computer fonts (see Chapter 4). This evaluation allows for proofing the system's capability to be easily adapted to different writing systems. The synthetically generated characters additionally allowed for tests with artificial noise and to proof that the methodology does not only apply for Glagolica.

1.2 Definition of Terms

In this section, commonly used terms will be discussed. Before going into details on definitions, a general remark on the notation of Glagolitic characters needs to be done. Figure 1.4 shows two Glagolitic ⰱ's and ⱆ's. Since the LaTeX font does not support the different shapes of these characters, they are defined as $ⰱ_a$ and $ⰱ_b$ where $ⰱ_a$ marks the initial which consists of two circles connected by an arrowhead and $ⰱ_b$ denotes the character that is represented by the font. For the ⱆ the same notation is used.

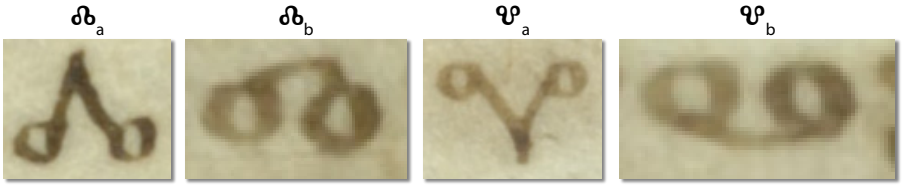

Figure 1.4: Definition of ⰱ and ⱆ

Subsequently, a list with definitions of commonly used abbreviations will be given.

DoG **Difference-of-Gaussian**: An approximation to the LoG which is computed by successively differencing images that were previously smoothed with Gaussians having different scale parameters σ [Low04]. This method allows for finding blob-like structures of different scales in images (see Section 3.1).

Fast **Features from Accelerated Segment Test**: A corner detector which extracts corners of a single scale. Therefore, Bresenham circles around each pixel are considered. Corners are classified according to previously learned rules [RD06] (see Section 2.2).

Gloh **Gradient Location-Orientation Histogram**: A local descriptor that was first proposed by Mikolajczyk et al. [MS05]. It is similar to Sift but exploits the Pca for dimensionality reduction (see Section 2.3).

k-nn ***k*-Nearest Neighbor**: A simple classifier. It predicts classes by finding a sample's k nearest neighbors and accumulating their labels [DHS00].

LoG **Laplacian-of-Gaussians**: A scale-space that is computed by repeatedly applying a LoG filter. The filter which is illustrated in Figure 2.6 is a robust high-pass filter [Lin94].

Mser **Maximally Stable Extremal Regions**: An interest Point detector which finds image regions by means of a watershed-like segmentation. It was proposed by Matas et al. [MCUP04] and proved to be the most stable interest point detector in studies by Mikolajczyk et al. [MTS+05] (see Section 3.1).

Ocr **Optical Character Recognition**: The process of transcribing documents from digital images by recognizing characters [RK09].

PCA **Principal Component Analysis**: A statistical method that allows for dimensionality reduction. This is achieved by computing the eigenvectors of the data's covariance matrix. Thus, the feature space is transformed to a new vector basis where the dimensions can be sorted according to their importance [DHS00, Jol02].

SIFT **Scale Invariant Feature Transform**: A local descriptor which was first proposed by Lowe [Low04]. It is based on accumulating gradients according to their orientation and location into a high dimensional feature vector (see Section 3.2).

SURF **Speeded Up Robust Features**: A local descriptor similar to SIFT proposed by Bay et al. [BTG06]. It can be computed faster than SIFT since integral images are exploited for the interest point detection and the feature vector's construction (see Section 2.3).

SUSAN **Smallest Univalue Segment Assimilating Nucleus**: A corner detector that is based on non-linear filtering. It extracts corners in a single scale [SB97] (see Section 2.2).

SVM **Support Vector Machine**: A classifier which allows for classifying high dimensional features by solving a dual optimization problem. It is based on risk minimization rather than error minimization which is known for tending to over fit the training data [VC74] (see Section 3.2).

1.3 Results

In order to choose the best performing local descriptors, state-of-the-art methods where compared on the dataset investigated. For these experiments affine transformations where applied to the document images. It turned out that SIFT, in combination with the DoG, is most robust with respect to image transformations such as scale, rotation and projective distortions.

The system proposed was evaluated using synthetic data, degraded characters and real world data. For the test with synthetic data, character images with different fonts (e.g. Times New Roman, Arial) where generated. For undistorted data, the system's precision is 0.96. The single false predictions are i and j when written with Arial. This is because solely small corners with different directions are recognized. In order to simulate partially visible characters, the characters in the synthetic images where occluded. If 50 % of the characters are occluded, the system's precision is 0.75. The precision is 0.904, if Gaussian noise with zero mean and $\sigma = 0.008$ is added to the initial data.

In the second experiment, degraded characters were extracted from the *Cod. Sin. Slav.* 5N. On the degraded test set the precision was 0.789 compared to 0.981 if characters with a high dynamic range are evaluated. Considering 25 different characters, a precision of 0.717 is achieved when partially visible and faded-out characters need to be recognized.

Finally, a test on real world data including 1055 characters was performed. Aside from the evaluation of crucial parameters, a comparison between no clustering and with

the character localization proposed was done. The $F_{0.5}$-score, which is a weighted mean between precision and recall, is 0.804 if characters are localized using synthetic clustering. In contrast, the $F_{0.5}$-score decreases to 0.772 if characters are localized with the proposed interest point clustering. A remarkable fact is that the precision does not significantly change (0.005) between these experiments, but the performance decrease can be traced back to the recall which decreases from 0.748 to 0.673. This can be attributed to characters which are missed if clustering errors occur.

1.4 Thesis Structure

Having previously discussed the motivation for this thesis, the related work will be subsequently given in Chapter 2. There, the state of the art for degraded character recognition is described in the first part. The second part details related work on object recognition focusing on interest point detectors and local descriptors.

In Chapter 3 the interest point namely DoG and the local descriptor (SIFT) is described in detail. Additionally, comparisons of different interest point detectors and local descriptors are discussed in this chapter. The SVM and methods used for properly training the system are discussed accordingly. The chapter's final section addresses the character localization which was especially designed for manuscript images.

Chapter 4 details experiments and the system's results on the dataset investigated. In order to show the system's performance, three experiments were carried out using different datasets. The first of which evaluates the system for Latin script where its behavior is tested if artificial noise is being introduced. In the second experiment, images containing single characters are used to compare the system's performance on degraded and well preserved characters. Finally, a test on real world data is carried out that allows for computing the precision and recall on degraded document images.

At the thesis' end, a conclusion is given in Chapter 5, which discusses advantages and disadvantages of the system proposed. Additionally, future developments are depicted that may improve the character recognition system.

Chapter 2

Related Work

In this chapter, an overview of state-of-the-art methods is given. The objective is to show the current progress of document analysis and object recognition. The chapter's first part – which deals with Ocr – demonstrates the current frontiers in character recognition of ancient documents. The second part aims at introducing common object recognition methodologies with the history of local features in particular.

It is not intended to give an exhaustive survey about object recognition methodologies, but to give a short overview mapping important concepts and ideas. A more detailed explanation of the methods used in this thesis is given in Chapter 3. Additionally, the respectively cited papers particularize the discussed topics.

First, Ocr systems are discussed in Section 2.1 focusing on off-line character recognition applied to degraded documents. In addition to general document pre-processing methods, new developments in document binarization are detailed in Section 2.1.1. Since this thesis is geared to object recognition, its state of the art is discussed in the Sections 2.2 and 2.3. The first of which deals with the progress of interest point detection. The latter gives an overview of remarkable local descriptors and their performance evaluated in [MS05].

2.1 Optical Character Recognition Systems

It is reported in [AYV01] that the first character recognition system was developed in 1900 by Tyuring who aimed at assisting visually impaired people. However, Handel patented a so-called *Statistical Machine* which was able to optically recognize characters in 1933 [Han33]. Similarly, the Austrian inventor Tauschek developed an analog optical reading device [Tau35]. In Figure 2.1 the illustration of Tauschek's *Reading Machine* is given. He recognized characters by means of templates which are projected onto the document. If a template matches the character (number) hardly any light is backscattered. Thus, a photo sensor can recognize if a template matches the currently observed character.

Beginning in 1940, the first digital Ocr systems were developed. At that time, scientists focused on machine printed Latin documents. For that task, simple template matching algorithms were designed that matched each character present in a document with a set of predefined character images.

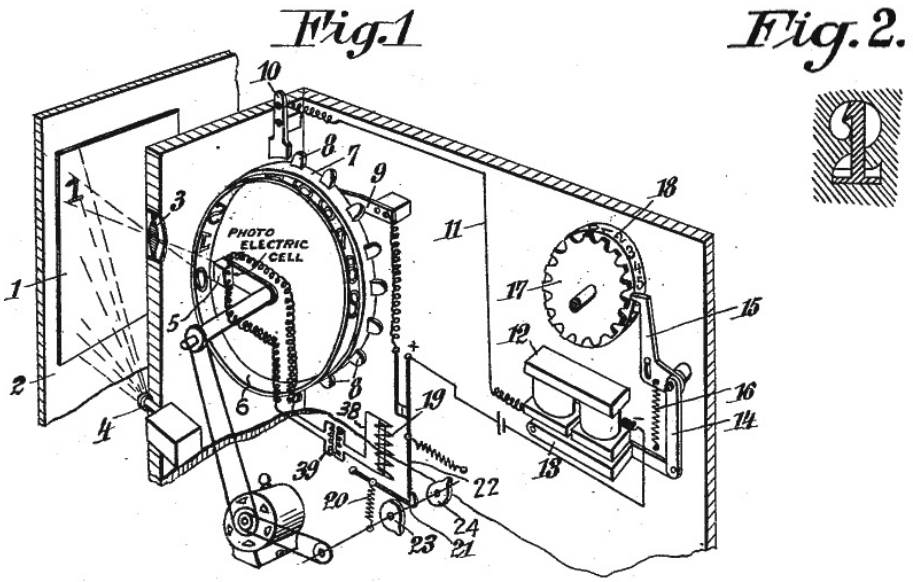

Figure 2.1: Tauschek's analog *Reading Machine* [Tau35].

Modern Ocr systems can be divided according to their input data. A principal difference is on-line versus off-line Ocr. The first of which deals with the recognition of words written on a digital device such as a Pda. The latter analyzes digitized manuscript or machine printed pages. In on-line Ocr systems, the input data does not require pre-processing. Thus, the data is already binarized and thinned as a result of the input device. Additionally, the signal is time dependent meaning that the time is known when strokes where written. Therefore, the writing direction of each stroke is known. A survey on on-line and off-line handwriting recognition is given in [PS00, Vin02]. Off-line Ocr systems are further discussed in Section 2.1.2. Figure 2.2 illustrates the classification of Ocr systems. This illustration does not show the difference between constrained and unconstrained Ocr. The first of which are systems having a constrained vocabulary (e.g. postal address recognition, geographical names).

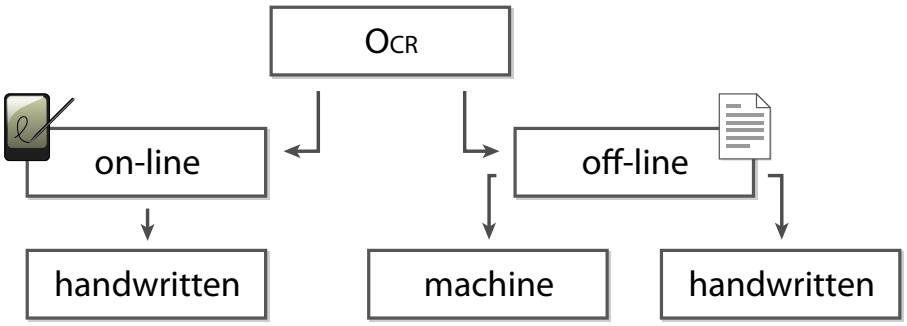

Figure 2.2: Classification of Ocr systems.

2.1.1 Document Analysis

State-of-the-art OCR systems need a preceding document analysis in order to recognize the characters. Typical document analysis steps include layout analysis [DKS09, MEE+09], skew estimation [Hul98, vBSB09], text line extraction [KSGM08, LSZT07] and binarization [GNP09]. In this section, related work on binarization will be detailed. A survey about character segmentation is given in [CL96].

In Figure 2.3, established image binarization methods of the 20th century are applied to the investigated dataset. For visualization purposes, objects are set to 0 (black) and background is set to 1 (white). It can be seen that the global binarization method proposed by Otsu [Ots79] is not capable to correctly segment the characters. As a result of background clutter, the method segments background in the left image part. In addition, faded-out ink causes a low dynamic range which results in filled character holes. In addition character holes are filled because of faded-out characters which results in a low dynamic range. The Sauvola [SP00] binarization method performs visually better on this test image. However, for this result the parameters (especially $k = 0.2$) had to be tuned which is crucial if a varying dataset is investigated. Degraded characters such as the ⊔ in the last text line cannot be extracted correctly. Similarly to the Otsu binarization, background clutter is segmented in the left image region.

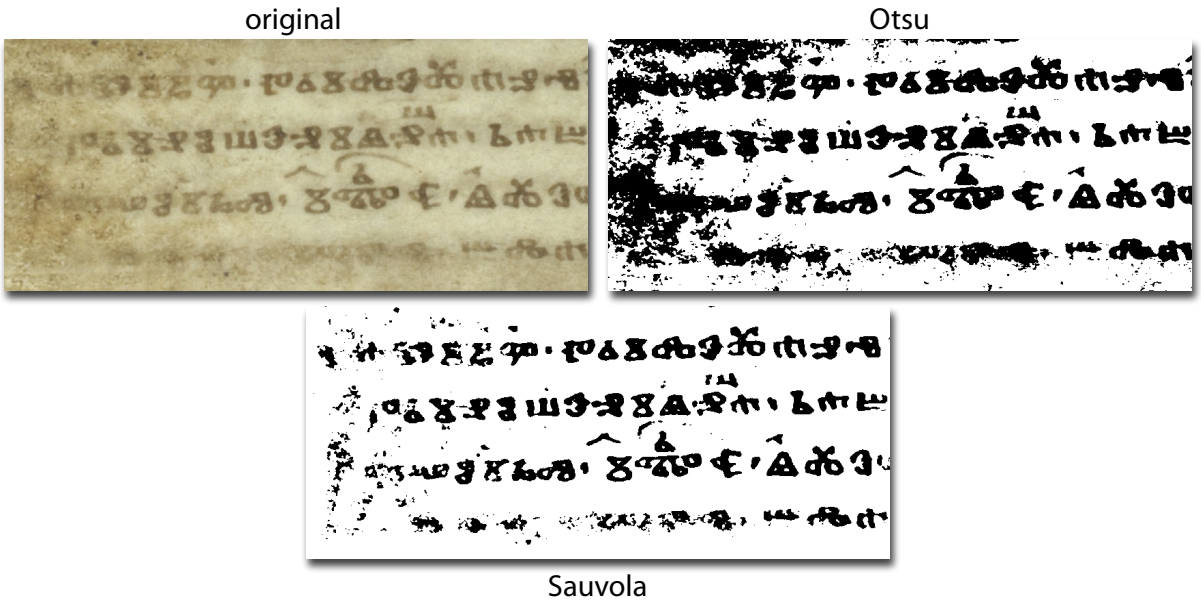

Figure 2.3: Comparison of two binarization methods on the investigated dataset.

Otsu [Ots79] proposed in 1979 a global thresholding approach that considers the class variances. It takes into account a gray scale image's histogram and assumes that all gray-values belong to two classes: foreground and background. In order to find the best global threshold, the intra-class variance is minimized while at the same time maximizing the inter-class variance. Even though this method was not designed especially for document image binarization, it proved to perform perfectly if printed scanned documents are considered. However, when for example photographed documents with changing illumination

need to be processed, a global thresholding approach fails.

That is why Niblack [Nib90] proposes a local thresholding approach based on the local mean value and standard deviation. Sauvola [SP00] further improves this method by adaptively amplifying the standard deviation. The mean and standard deviation of a local region are computed for each pixel. This can be calculated efficiently if integral images are exploited. Then, a threshold is assigned to each pixel according to:

$$T(x,y) = m(x,y)\left[1 + k\left(\frac{s(x,y)}{R} - 1\right)\right] \tag{2.1}$$

where $T(x,y)$ is the resulting threshold for each pixel, $m(x,y)$ and $s(x,y)$ are the local region's mean and standard deviation respectively. The standard deviation's dynamic range is given by R and $k > 0$ is used to control the influence of $s(x,y)$. This local adaptive thresholding method is capable of binarizing document images of poor quality. It can especially handle changing illuminations and background clutter which arises from repeatedly copying the same page. However, considering ancient or medieval manuscripts, this method fails particularly if the character size varies, homogeneous background is present and characters are faded-out.

Bukhari et al. [BSB09] propose an improved document binarization method based on Sauvola's methodology. For this purpose, ridges are detected by means of multi-scale anisotropic Gaussian smoothing and the Hessian matrix. Instead of using a constant k in Equation 2.1, they suggest to vary $k(x,y)$ according to ridges previously detected. If $k = 0.05$ for foreground regions and $k = 0.2$ for homogeneous background regions, this method outperforms Otsu's and Sauvola's thresholding approach.

A similar approach that is based on Sauvola's thresholding method is proposed by Tanaka [Tan09]. He detects homogeneous background by extracting a flatness measure of local regions. This allows for a noise reduction which arises from Sauvola's binarization. Additionally, if more than two gray value classes are present in a local region, the current window is shifted away. This improves the segmentation of lines which are close to characters and have a different gray value.

Text binarization methods that focus on uneven lighting conditions are proposed by Lu et al. [LT07] and Kuk et al. [KC09]. The latter of which propose to initially estimate the shading by means of a Gaussian convolution having a large kernel. Then a descriptor is established which is based on mean filters and allows to classify pixels into Text Region (TR), Near Text Region (NTR) and Background Region (BR). Finally, pixels belonging to TR and NTR are relabeled by means of a graph cut method.

In contrast, Lu et al. [LT07] – winner of the Document Image Binarization Contest 2009 [GNP09] – developed a document binarization method based on a global Savitzky-Golay filter. In more detail, the shading is estimated by fitting a least square polynomial surface to a given document image. Combining the pixels' gray-values and the polynomial surface allows them to directly threshold the observed image. Their method outperforms Otsu's and Sauvola's thresholding method on the investigated dataset.

Yosef [Yos05] proposes a binarization method focusing on degraded manuscript images. Therefore, a global threshold (e.g. Otsu) is applied to the manuscript image. According to CC statistics such as the aspect ratio of the CC's bounding box, characters connected with

background clutter in the binary image are detected. Noisy characters are then converted into seed regions for a growing process that finds the final character form.

Ntirogianis et al. [NGP09] developed a binarization method which handles printed and handwritten degraded document images. Therefore, a local binarization method such as the Sauvola threshold is initially applied to the document image. Computing the skeleton and the outer contour of each CC allows for a stroke's width estimation. An adaptive parameter is then applied to the local thresholding method that considers the likelihood of a background pixel for belonging to a character according to the previously computed stroke width. An additional method for binarizing degraded document images is proposed by Xi et al. [XCL+07]. They combine two local thresholding methods namely Niblack's [Nib90] and Palumbo's [PSS86] which is based on local contrast information.

In addition to these binarization methods, a work from Ramanan [RS06, Ram06] is discussed which deals with localizing objects. He proposes to train deformable models that estimate the pose of an object in order to get a fuzzy segmentation. This allows for a localization of objects. Considering that handwritten characters are deformed prototypes, one could think of adapting this approach for localizing characters that cannot be binarized correctly.

2.1.2 Recognizing Characters of Degraded Documents

In this section, state-of-the-art OCR systems for degraded documents are presented. Current OCR systems have three basic steps in common which are shown in Figure 2.4. First, document pre-processing, which was given in the previous section, is performed. There the document's skew is estimated, the text layout is extracted and the document image is binarized. Subsequently, binary features, which will be further discussed in this section, are extracted. These features are then classified by means of a Neural Networks (NN) or a SVM. Some OCR systems have an additional step which is not illustrated in Figure 2.4. They use a dictionary in order to correct spelling mistakes caused by character classification errors. Finally, each character gets assigned a corresponding class label.

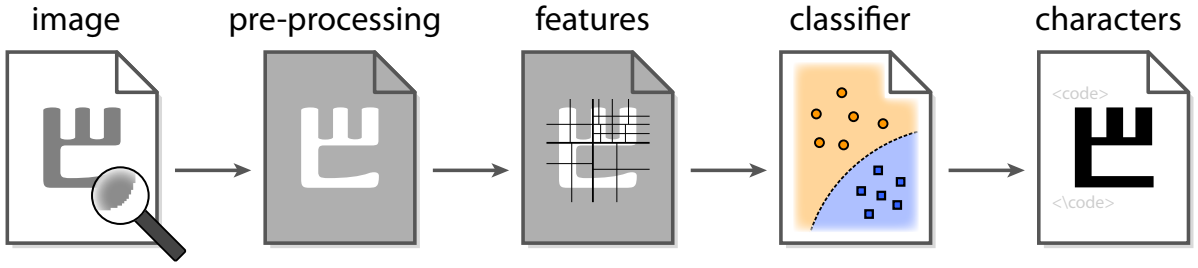

Figure 2.4: General OCR system design.

The approaches subsequently presented differ according to the data investigated. Thus, three general data sets are differentiated: typewritten documents, cursive handwritten documents and handwritten documents. Figure 2.5 illustrates documents of the particular datasets. The Georg Washington document in Figure 2.5 (middle) can be correctly binarized since the background is homogeneous. However, a correct character segmentation is hard as stated in [LRM04] because of the cursive script. On the contrary,

the Hebrew manuscript in Figure 2.5 contains background clutter because of the ink on the reverse bleeding through.

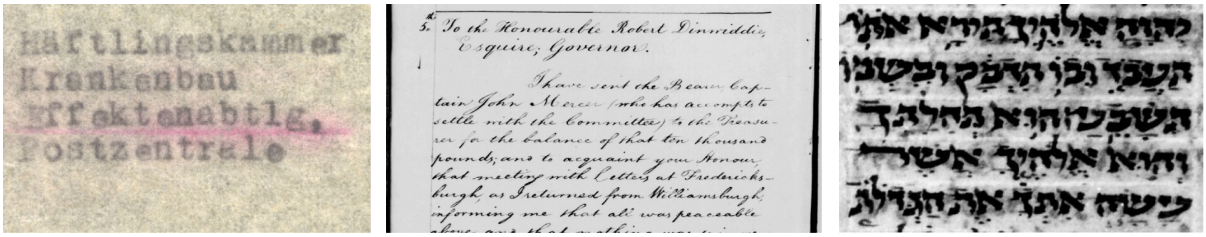

Figure 2.5: Degraded typewritten document (left), cursive handwritten document from the George Washington collection (middle) and Hebrew manuscripts (right). Courtesy by Pletschacher [PHA09], Lavrenko [LRM04] and Yosef [Yos05] (from left to right).

A framework for recognizing degraded typewritten documents from the 19$^{\text{th}}$ century is proposed by Pletschacher et al. [PHA09]. They propose to train the classifier using a semi-supervised clustering approach. Therefore, binary features such as the normalized height or aspect ratio are extracted. Based on the feature's information, glyphs are clustered such that the same characters are grouped together. Human feedback allows to label and correct the automatically found glyph clusters.

Lavrenko et al. [LRM04] directly recognize words from the George Washington collection. Hence, previously segmented words need to be normalized according to the slant, skew and baseline. Then, scalar features such as the word's width or aspect ratio and profile-based features (e.g. projection profiles) are computed on the normalized word images. A Hidden Markov Model (HMM) with hidden states that represent words is used to classify the words. Lavrenko reports a precision on the George Washington collection of 0.603. This technique was later improved by Rath et al. [RM07] who propose to use dynamic time warping in order to compensate non-linear variations present in manuscripts.

Similar to the previously mentioned methods, a word recognition system is proposed by Frinken et al. [FB09]. They compute statistical moments from sliding windows that are applied to normalized word images. A NN with one hidden layer is constructed for the classification. In addition, the a priori data distribution is trained by means of semi-supervised learning that is fed with labeled and unlabeled data. Frinken et al. [FPF$^+$09] additionally combine this methodology with HMM's in order to improve the word recognition.

Contrary to the word recognition methods, Alirezaee et al. [AAFF05] developed a character recognition system for medieval Persian manuscripts. They extract statistical features such as Pseudo-Zernike moments from previously binarized document images. In order to find features that are discriminately, the Fisher Linear Discriminant is used, which transforms the data such that the inter-class variance is maximized. The resulting weight function is used for character classification.

Arrivault et al. [ARFMB05] propose a combined statistical and structural character recognition approach for ancient Greek and Egyptian documents. Therefore, two statistical features namely Fourier moments and Zernike moments are extracted from binary document images. According to the dictionary's size, a Bayes or k-NN classifier is used to

label characters according to statistical features. Structural features such as attributed graphs are computed and classified for characters which are rejected during the classification of statistical features.

Another approach that aims at recognizing historical Greek characters is published by Vamvakas et al. [VGSP08]. Having binarized the image and segmented individual characters, zone features and character profile features are calculated. The first of which are constructed by tiling the character image into zones and accumulating the character pixel density to the normalized zone image. Unlabeled character features are then clustered according to the features extracted. In a manual step, labels are assigned to the clusters and clustering errors can be corrected. Finally, a SVM is applied for character classification.

In 2007, Ntzios [NGP+07] developed a so-called segmentation-free character recognition system applicable for the same documents. He extracts geometrical features from binarized images in combination with a watershed-like algorithm that fills cavities. A decision tree is used for the character classification. Since the decision tree and the feature extraction are highly script dependent, the approach does not show promising for generally recognizing ancient manuscripts.

2.2 Interest Point Detectors

In this section, an overview of state-of-the-art interest point detectors is given. The detection of interest points is a crucial task, since the results of the subsequent feature matching is directly related to its performance. Hence, if an interest point detector is chosen which has a low repeatability against certain geometrical distortions (e.g. scale change) that are present in the observed images, the feature matching performs poorly. This is because interest points which are found in one image are not detected in another image because of the detector's low repeatability. As a consequence, interest points with no corresponding partner in the other image cannot be matched at all, since the same interest points need to be selected in both images in order to match them. Due to the previously mentioned importance of the interest point detection, it is a well investigated but still active research topic (see [Mor81, HS88, MS01, Low04, BTG06]). This section does not cover all interest point detectors, but gives an overview of important concepts. A more detailed explanation of the topic is given in [Mik02].

All interest point detectors presented are based upon derivatives or their approximations since derivatives allow for extracting structures invariant to global illumination. Figure 2.6 shows the first and second partial derivatives of a 2D Gaussian. The y derivatives (g_y, g_{yy}) are the same as the transposed x derivatives. In addition to the Gaussian derivatives, the LoG is illustrated in Figure 2.6.

2.2.1 Corner Detectors

Local interest points for stereo image matching tasks were first introduced by Moravec [Mor81] in 1981. He proposes to compute features at image locations which possess corners in order to minimize the number of wrong matches. Therefore, the directional variance is measured using squared sums of adjacent pixel differences in four directions.

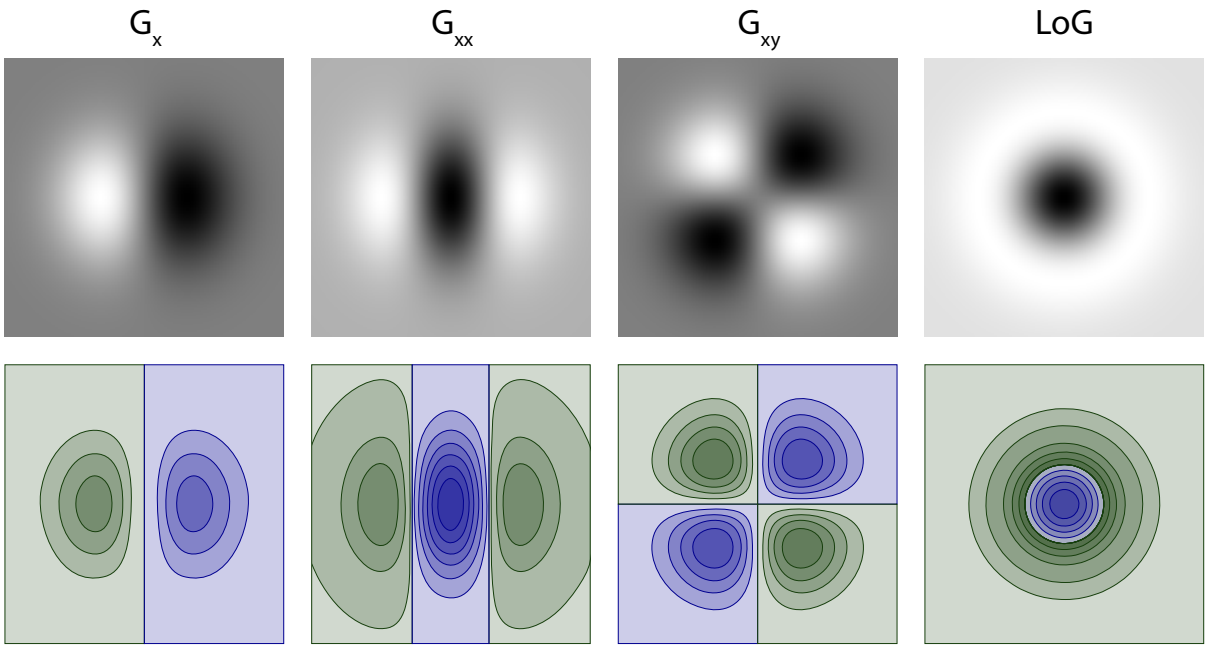

Figure 2.6: Gaussian derivative kernels and the LoG kernel which are commonly used for interest point detection.

The window's interest measure is subsequently calculated by the minimum of these sums. Moravec suggests locating features at local maxima of the interest measure.

Harris and Stephens [HS88] improves the repeatability of the Moravec detector using the second moment matrix (autocorrelation-matrix). The so-called Harris corner detector extracts feature points at locations of corners and image regions which have large gradients in all directions. A drawback of the Harris corner detector is its sensitivity to scale changes. Thus, feature points can solely be extracted at a predefined scale.

In order to compensate the lack of scale-invariance, Mikolajczyk et al. [MS01] combines the Harris corner detector with a Laplacian. Thus, the features are spatially located using the Harris function. Afterwards, the characteristic scale is found by the maximum of the Laplacian in a scale-space introduced by Lindeberg [Lin94]. By this means, it is possible to detect regions of interest which have a high (80 % for a scale factor of 1.2) repeatability with respect to scale changes.

Mikolajczyk [Mik02] additionally exploits the Hessian matrix for interest point detection. The Hessian matrix – a square matrix consisting of second-order partial derivatives – is used to select the dominant scale of an interest point by selecting the maxima of the determinant. Even though the trace of the Hessian matrix is the same as the Laplacian, the scale selection is more robust with respect to illumination changes and noise [Mik02]. Recently, a Fast-Hessian detector was presented by Bay et al. [BTG06]. There, the Gaussian second-order derivatives are approximated by box filters accelerating the computation in combination with integral images.

Smith and Brady [SB97] proposes SUSAN which is a fast corner and edge detector based on non-linear filtering. Therefore, a mask is defined which compares each pixel

within the given mask with the current center (nucleus) of the mask. Afterwards, pixels with a similar brightness to the nucleus define an area which is used to find the local structure.

Another method that focuses on real-time corner detection rather than finding corners accurately invariant to a given set of distortions was recently proposed by Rosten and Drummond [RD06] and is called FAST. This method is similar to SUSAN but considers a Bresenham circle around each pixel. The pixels are classified into corner and non-corner pixels respectively with a machine learning algorithm. The method is further optimized with the ID3 algorithm which minimizes the access rate per pixel. Finally, a non maxima suppression is performed in order to guarantee that no real corner produces more than one detected corners.

2.2.2 Blob Detectors

Lowe [Low99] first introduced SIFT in 1999. He recognizes objects using scale and rotationally invariant features. In contrast to the previously mentioned methods, the features are not localized with the Harris function but by computing the DoG which detects blob-like image regions. In order to localize features spatially and in scale, local extrema of the DoG function are computed. Mikolajczyk [Mik02] showed with experimental comparisons, that the most stable features are produced by extrema of the LoG (see Figure 2.6). Since the DoG is an approximation of the LoG – for the sake of computational effort – the results of both methods are similar.

2.2.3 Other Techniques

A summary of other commonly used interest point detectors is given consecutively. Kadir and Brady [KB01] compute saliency regions by measuring the entropy of pixel intensity histograms which are computed for elliptical regions. In order to select the scale of the detected interest points, they search for the maximum in the scale-space of each feature. In 2002, Matas et al. [MCUP04] introduced MSER which are extracted with a segmentation algorithm that is similar to the watershed segmentation. Later, Mikolajczyk et al. [MTS+05] demonstrate that MSER are robust with respect to viewpoint changes, but have low repeatability under increasing blur and scale changes compared to other well-known interest point detectors.

Carboneto et al. [CDS+06] propose to combine different interest point detectors in order to improve the results of object recognition. More precisely, they combine the Harris-Laplace, Kadir-Brady and LoG detectors and conclude that the image classification could be improved over the Harris-Laplace detector. Nevertheless, the interest point detection using a combination of detectors does not significantly outperform the Kadir-Brady detector in combination with their dataset but is more computationally expensive [CDS+06].

2.3 Local Descriptors

This section gives a brief overview of current research on local descriptors. The principle of local descriptors is to find distinctive image regions such as corners and to analytically describe these regions independent of a predefined set of transformations (e.g. affine transformations). A remarkable advantage of local descriptors compared to global methods is their robustness with respect to occlusions and global non-linear distortions present in images [Low04, Mik02]. Thus, local descriptors are capable of recognizing objects even if parts (see Section 4.1) of the objects are occluded because solely local information is computed to establish the correspondence between the objects.

While at the beginning, image matching on the basis of local features was solely used in stereo vision tasks, Schmid and Mohr [SM97] proposed to use feature matching for image retrieval tasks. Therefore, they build a method which uses the Harris corner detector for feature localization and compute a feature vector by means of Gaussian derivatives which are called *"local jet"* [KvD87]. Schmid and Mohr show that matching local features outperforms previous global methods for image retrieval tasks. Currently, the methods are applied to solve general image processing tasks such as wide baseline stereo vision [MCUP04], shape matching [BMP02] and object recognition [FPZ03], object localization [CDS$^+$06, MLS06, LBH09].

An intuitive local descriptor would be to take n pixel intensities in a predefined region around the localized interest point and convert them into an n-dimensional vector. Obviously, this descriptor would fail if an affine transformation such as a rotation with an angle $\theta > \varepsilon$ was applied to the image. Another drawback of such a descriptor would be its dependency to photometric transformations (e.g. intensity changes) caused by changing illuminations or sensor noise. The matching of such a descriptor can be done with the normalized cross correlation in order to obtain matching results independent of intensity changes. Nevertheless, the high dimensionality which results in a high computational complexity and its sensitivity to affine transformations limit the applications of such a descriptor. That is why local descriptors are designed to be robust with respect to geometrical and photometric transformations of a given dataset.

2.3.1 Distribution-Based Descriptors

In contrast to simple descriptors, distribution-based descriptors use a histogram of locally measured data in order to represent the local appearance. Johnson and Hebert [JH97] proposes a distribution-based local descriptor for 3D object recognition on the basis of oriented points. Therefore, they compute the position of other points with respect to the selected point. Lazebnik et al. [LSP03] adapted this approach to 2D images by taking into account the intensity values and the distance between neighboring pixels and the reference point.

Another descriptor called *shape context* which is based on point distributions is proposed by Belongie et al. [BMP02] for shape matching and object recognition tasks. For this purpose, the canny edge detector [Can86] is computed and the interest points are uniformly sampled on the edge of objects. Afterwards, a log-polar histogram containing the relative distances to all $n - 1$ remaining interest points is constructed. The log-polar

space guarantees that nearby interest points are emphasized.

As previously mentioned, Lowe [Low04] proposes SIFT for object recognition tasks. There, each interest point is represented by a three dimensional histogram of the gradient magnitudes' distribution weighted by their orientation. In more detail, eight orientation planes consisting of 4×4 bins are constructed which results in a 128–dimensional feature vector. The scales of interest points are determined by computing a DoG scale-space. Invariance to changes in rotation is achieved by transforming the coordinate system with respect to the dominant direction which is found by the global maximum of a histogram over all orientations.

Mikolajczyk and Schmid [MS05] extended the SIFT approach in order to gain more robustness and distinctiveness. To achieve this, they use a log-polar location grid instead of a Cartesian grid. For the so-called GLOH they take into account different radii and gradient orientations which results in 272 dimensions. Since the performance of the matching process decreases with increasing dimensionality, Mickolajczyk proposes to compute the PCA in order to reduce the dimension of each descriptor to 128. The covariance matrix of the PCA was estimated using 47000 image patches. Nevertheless, the PCA may perform poorly for specific datasets as a result of the estimation process.

Ke and Sukthankar [KS04] recently improved the SIFT descriptor. Therefore, they take into account a 41×41 image patch at each interest point detected. Having computed a 3 042 dimensional SIFT descriptor, the PCA is calculated to reduce the vector's dimensionality. The PCA was applied to the covariance matrix of 21000 image patches. Afterwards, the eigenvectors are sorted according to their importance and the top n are taken into account. Ke proposes – based on empirical studies – to take the first 20 eigenvectors. Despite the low dimensionality compared to the SIFT descriptor, the authors show experiments where PCA-SIFT performs better than the original SIFT algorithm. They trace this effect back to the fact that eliminating the lower components of the PCA removes unmodeled distortions.

Due to the fact that distribution, based high-dimensional descriptors exhibit the best performance on general object recognition tasks [MS05], Bay et al. [BTG06] designed a new descriptor called SURF for on-line applications focusing on computational speed. Similar to the SIFT descriptor they obtain rotation invariance by normalizing the descriptor with its dominant orientation. To achieve this, the Haar-wavelet responses are computed in x and y direction using integral images. Then, the dominant orientation is determined by calculating the sum of all responses within a sliding window. Finally, the 64-dimensional descriptor is constructed by summing the Haar-wavelet responses in x and y direction and their absolute values in 4 sub regions around the interest point.

2.3.2 Other Techniques

In contrast to distribution-based descriptors, the interest point neighborhood is approximated in differential descriptors by derivatives of a given order. Koendrik and van Doorn [KvD87] were the first to investigate local derivatives, called *local jet*. The derivatives are computed by convolution with Gaussian derivatives (see Figure 2.6). Since this approach is not rotationally invariant, Florack et al. [FHRKV94] proposes to compute invariants which are combinations of *local jet* components and additionally reduce the dimension

of the feature vector. A further approach to gain rotational invariance of differential descriptors is to use steerable filters investigated by Freeman and Adelson [FA91]. There, the derivatives are steered in the gradient direction.

Another method describing local context is to compute central moments up to a given order [GMU96]. Then, invariants are calculated that describe the shape and intensity distribution within a defined region.

2.3.3 Performance of Local Descriptors

Mikolajczyk and Schmid [MS05] evaluate the performance of ten different local descriptors (amongst others: SIFT, GLOH, PCA-SIFT, *shape context*). They compare the precision/recall of each descriptor on a database[1] that contains real images with different geometric and photometric transformations such as rotation, viewpoint change or JPEG compression. They conclude that GLOH performs best for object matching and object recognition tasks. Nevertheless, the performance of GLOH is not significantly better than the performance of SIFT throughout their tests, but it is computationally more expensive than SIFT. Similar results are obtained by *shape context*. However, the performance of *shape context* decreases significantly if edges in the images are not reliable. PCA-SIFT performs worse than the high-dimensional descriptors. Mikolajczyk and Schmid take 36 eigenvectors into account which showed – empirically evaluated on their database – the best results for low dimensional descriptors. They do not mention if the projection matrix is trained for their database or if they apply the proposed one. The best performance of low-dimensional descriptors is achieved by gradient moments and steerable filters.

Summary

In this section, related work about character recognition and object recognition was depicted. According to the discussed state-of-the-art OCR systems, it can be assumed that recognizing characters in ancient and degraded manuscripts has still not reached the final frontier. It was additionally shown that all current systems extract their features from binary images. This can be traced back to the fact that character recognition systems where developed since the beginning of the 20^{th} century, a time when object recognition was not feasible because of hardware constraints. However, in the last two decades, object recognition systems have become powerful tools in Computer Vision (CV). That is why it is proposed in this thesis to use object recognition methodologies for character recognition in order to overcome challenges that arise when degraded manuscripts are observed.

In addition to current OCR systems presented, related work in the field of object recognition was described. For that purpose, an overview of the last two decades was given. The interest point detectors and local descriptors explained will be further compared and discussed in the subsequent Chapter which details the design of the character recognition system proposed.

[1] available at: `http://www.robots.ox.ac.uk/~vgg/research/affine`

Chapter 3

Methodology

In contrast to state-of-the-art systems, the system proposed has a fundamentally new architecture which is designed to compensate the drawbacks that arise when dealing with ancient manuscripts. Instead of applying a binarization so as to compute features, they are directly extracted from the gray-scale image.

The system is divided into two major tasks: classification and localization. Both tasks are based upon the extraction of interest points which are computed by means of the DoG. This interest point detector extracts blob-like regions at different scales of an observed manuscript image. Thus, the x, y coordinates as well as the scale s are provided for each region of interest.

Exploiting this information, local descriptors are calculated which describe the respective regions by means of gradient vectors that rely on the pixels' gray-scale values. These local descriptors are directly classified using a multi-kernel SVM. Having classified all extracted image regions, one character consists of multiple pre-classified points.

In order to assign one class label to each character present in an image, the interest points need to be clustered. Therefore, character center estimation is performed, which exploits the fact that each character produces a single interest point at a specific scale (see Section 3.4). This estimation is used for an improved initialization of the k-means clustering which groups the interest points according to the subjacent characters. Finally, the information gained by the classification and localization steps is merged together. The so-called interest point voting weights the class probabilities of all local descriptors belonging to the same cluster and assigns the final class label to each character.

In this chapter, the methods for character recognition are detailed. Figure 3.1 illustrates the two major tasks and gives an overview of the core methods. As can be seen, the character localization and the classification are based on interest points. Both tasks are computed in parallel as they do not depend on each other. Finally, a voting scheme merges the information gained by localization and classification and predicts character labels. Section 3.1 presents the interest point extraction. The local descriptors based on SIFT are described in Section 3.2. Their classification accomplished by a SVM is detailed in Section 3.3. Whereas in Section 3.4 the character localization, which is needed to group the local descriptors, is illustrated. Finally, the descriptor voting is given in Section 3.5.

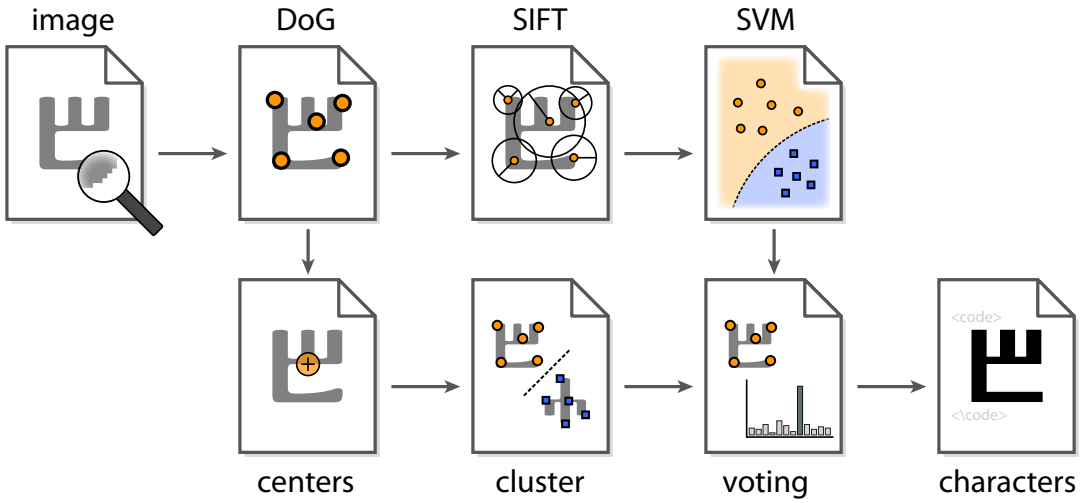

Figure 3.1: The system proposed consisting of two major tasks: classification (top) and character localization (bottom).

3.1 Interest Point Detector

In Section 2.2, an overview of state-of-the-art interest point detectors was given. Additionally, advantages and drawbacks were depicted for each method. For the system proposed in this thesis, the DoG detector is used for the localization of image regions where local descriptors are computed. It was chosen by reason of the consecutively enumerated advantages which were gathered by studies of Mikolajczyk [MTS+05, MS05], Lowe [Low04] and comparisons of interest point detectors on the investigated dataset (see Section 3.1.2). Thus, the main advantages of the DoG are subsequently given.

- Blobs are detected in a scale-space. That is why features can be extracted in a scale invariant manner.
- The scale-space is computed by convolving an image with Gaussians having an increasing σ. As a consequence, the DoG is robust with respect to noise caused by e.g. the camera sensor or JPEG compression.
- The DoG is computationally faster than the LoG but produces similar results.
- The DoG detects more interest points[1] than other approaches such as MSER or FAST. Thus, a character is described with more details ($\approx 70.8\,\%$) which results in a higher reliability of the descriptor classification.
- Mikolajczyk [Mik02] states that the DoG has a higher repeatability for viewpoint changes below $50\,^\circ$ than Harris based interest point detectors.

[1] The DoG detects 1997 interest points for a sample image having 474×616 px where MSER detects 584 and FAST detects 1057 interest points.

3.1.1 Interest Point Localization

In order to detect interest points invariant to scale changes of the image, a scale-space which was exhaustively studied by Lindeberg [Lin94] is constructed. The scale-space $L(x, y, \sigma)$ of an image $f(x, y)$ is constructed by convolving the image with Gaussians $G(x, y, \sigma)$ having a varying scale parameter:

$$L(x, y, \sigma) = G(x, y, \sigma) * f(x, y) \tag{3.1}$$

where $*$ denotes the convolution in x and y direction and σ is the scale parameter. The Gaussian filter kernel is defined by:

$$G(x, y, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp^{-(x^2+y^2)/2\sigma^2} \tag{3.2}$$

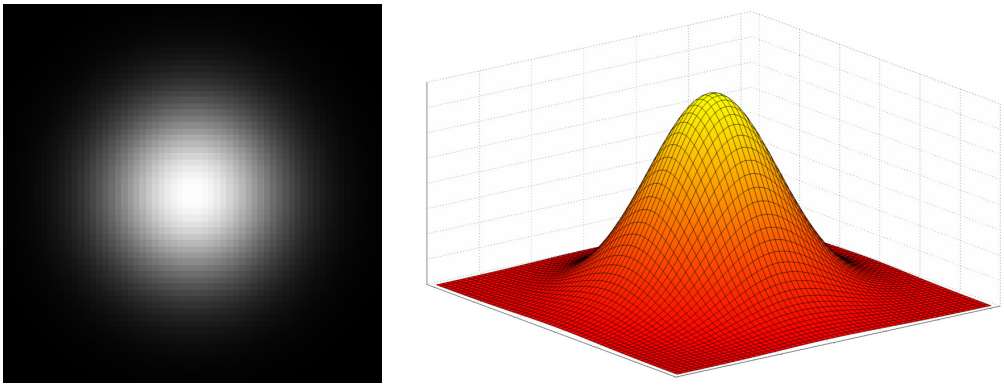

Figure 3.2: A Gaussian low-pass filter kernel with $\sigma = 10$ visualized as image (left) and as a function of x and y (right).

Figure 3.2 shows the Gaussian filter kernel which is a 2D representation of the well-known normal probability curve. Lindeberg [Lin94] proved that the Gaussian kernel is the only low-pass filter which can be used to compute a scale-space owing to its linearity and spatial shift invariance. This arises from the fact that each pixel of a finer scale contributes equally to a pixel of a coarser scale. Hence, structures of a coarse scale represent simplified structures of the finer scale levels and do not possess new structures generated by smoothing.

A convolution using a 2D symmetric filter kernel is equivalent to convolving the image with the same 1D kernel successively. Therefore, the scale space is computed, according to:

$$\begin{aligned} L(x, y, \sigma) &= G(x, \sigma)^T * (G(x, \sigma) * f(x, y)) \\ G(x, \sigma) &= \frac{1}{\sqrt{2\pi\sigma}} \exp^{-x^2/2\sigma^2} \end{aligned} \tag{3.3}$$

where $G(x, \sigma)^T$ denotes the transposed 1D Gaussian. This method accelerates the scale space computation since the convolution with a 2D kernel results in $O(HW \cdot M^2)$ multiplications and additions where HW are the image height and width respectively and M

is the kernel's size. Convolving an image with two 1D Gaussians results in $O(HW \cdot 2M)$ multiplications and additions which dramatically reduces the computational effort considering that M is at least 3 and in our case: $\sigma = \sqrt{2} \Rightarrow M = 9$.

The scale-space allows extracting structures of an image at different levels of details. In order to speed-up the computation of the scale-space, the images are resampled after σ has doubled which is called octave. Thus, the image size decreases exponentially with each octave. Due to the resampling, subsequent processing steps can be implemented efficiently. The Gaussian filter kernel additionally suspends noise introduced by e.g. the camera sensor or image compression.

Having constructed the scale-space, regions of interest are extracted at every scale level by means of the DoG $D(x, y, \sigma)$. It is computed by differencing images of two nearby scale levels which are separated by a constant factor k:

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * f(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \tag{3.4}$$

Since the scale-space – which is computationally intensive – needs to be computed anyway in order to gain scale invariance of the features, the DoG can be computed simply by subtracting the images which represent scale levels.

As mentioned in Section 2.2.2, the DoG is a close approximation to the LoG. Since the Laplacian, which is denoted by ∇^2 is a differential operator, structures such as edges and corners – more generally blobs – have strong negative or positive responses while flat regions become zero. Figure 3.3 illustrates the pyramid representation of a Gaussian scale-space and its corresponding DoG scale-space. Note the increasing smoothness of the image as σ is increased.

Extrema Detection

Having computed the DoG, interest points can be located simply by finding the positive and negative extrema of each scale level of a given image. Therefore, each pixel value $D(x, y, \sigma)$ is compared to the values of its 8-connected neighborhood. If the observed pixel represents a spatial local extremum within one scale level, it is compared with its 18 neighbors of the lower and the higher scale level. Solely pixel values which are local extrema spatially and in scale are chosen as possible interest point candidates. More precisely:

$$\begin{aligned} D(x, y, \sigma) \; &> D(x - i, y - j, (k - l)\sigma) \\ &\forall\, i, j, l \;\in\; \{-1, 0, 1\} \wedge (i \wedge j \wedge l \neq 0) \end{aligned} \tag{3.5}$$

where $D(x, y, \sigma)$ represents a scale-space level and k is a constant factor multiplied to σ in order to select different scale levels. The indices i, j, l are defined between $[-1, 0, 1]$.

Currently, the interest points are located at pixel coordinates. However, Lowe [Low04] established that the performance of feature extraction can be improved if the interest points are not placed at the central sample point. Therefore, a 3D quadratic function is fitted to the local function in order to determine the interpolated position spatially and in scale. At the same time, points are rejected, which have a low contrast and are unstable.

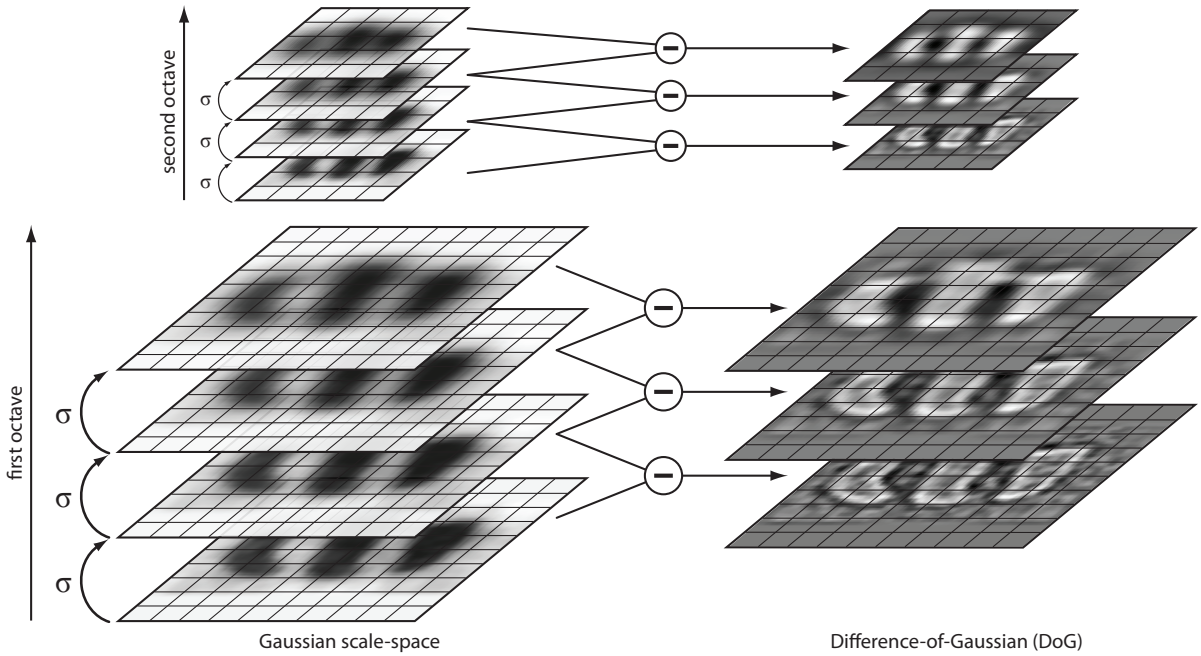

Figure 3.3: The first and the second octave of a Gaussian scale-space (left). Consider the increasing smoothness of the images within one octave as σ increases. The corresponding DoG is illustrated on the right side. There, edges and corners become black or white while flat regions are gray (zero).

In order to reject weak interest points, the quadratic function value is thresholded (*thresh*) at the extremum. This threshold is further studied in Section 4.3.1.

In addition to the mentioned weak interest points caused by noise, those located at edges have a poor localization along the edge. In order to detect such interest points the 2×2 Hessian matrix $\mathbf{H}$ is computed at their location. The Hessian matrix is defined by:

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{3.6}$$

where D_{ij} denotes the second partial derivatives by x and y respectively. The underlying idea of computing the Hessian matrix is to determine whether the principal curvature is large compared to the perpendicular curvature which is characteristically for interest points located at edges. Lowe introduced a measure which allows comparing the curvatures and therefore to find out if a point is weakly located on the edge without having to compute the eigenvalues of the Hessian matrix. This measure is defined by:

$$\frac{\mathrm{Tr}(\mathbf{H})}{\det(\mathbf{H})} < \frac{(r+1)^2}{r} \tag{3.7}$$

where r is a threshold, $\det(\mathbf{H})$ is the determinant of the Hessian matrix and $\mathrm{Tr}(\mathbf{H})$ is defined by:

$$\mathrm{Tr}(\mathbf{H}) = D_{xx} + D_{yy} \tag{3.8}$$

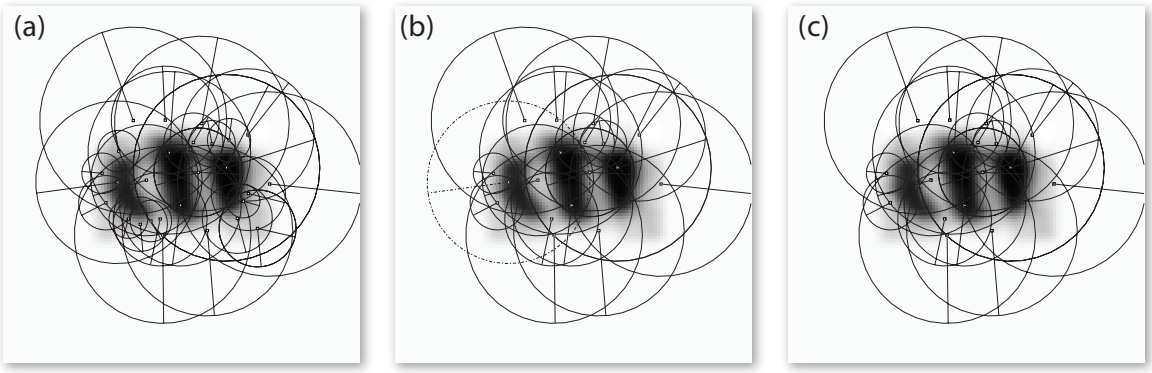

Figure 3.4: The images show a Glagolitic ⰻ with interest points. The threshold (*thresh*) of (a) is set to 0.007, in (b) it is 0.01 and in (c) r is set to 10.

Figure 3.4 shows three images of a Glagolitic ⰻ where the black circles indicate the scale of each interest point. Figure 3.4 (a) shows the interest points with a low threshold (*thresh*)(0.007). In Figure 3.4 (b) a threshold of 0.01 is applied which rejects interest points of lower scale levels since they are most likely caused by noise. This threshold is optimal for the given problem (see Section 4.3.1). Figure 3.4 (c) is computed with the same threshold as Figure 3.4 (b) but r is set to 10. In this case, one interest point is rejected which is located on the left vertical stroke of the character (illustrated with a dashed line in Figure 3.4 (b)).

3.1.2 Comparison of Interest Point Detectors

In order to emphasize the advantages of the DoG, different detectors are tested on the investigated dataset. Therefore, four state-of-the-art interest point detectors (namely: DoG, FAST, MSER, SUSAN) are evaluated on the dataset given. These detectors were selected since they (DoG, MSER) outperformed other detectors (see [MS05]) or they are fast (FAST) and not considered in previous performance evaluations (SUSAN).

An overview of the interest point detectors compared is given in Section 2.2. The interest point detectors' robustness to three relevant types of affine image transformations (scale, rotation, projective), which are illustrated in Figure 3.5, is evaluated.

These transformations arise when document images are not scanned but digitized using a camera which is the case when books or ancient manuscripts are considered. To exemplify, the scale-changes result from different resolutions of digital cameras, changing object lenses or changing the distance between the camera and the object imaged. Rotation variations arise from rotations of the manuscript pages as well as non parallel text lines (local rotations). Projective transformations occur when documents are imaged without a controlled environment and, therefore, the camera is not positioned normal to the document's surface.

The robustness is evaluated with four test panels, containing 84 characters, are synthetically distorted according to the defined image transformations. Four test panels were chosen since three turned out to be not statistically significant. On the other side, more

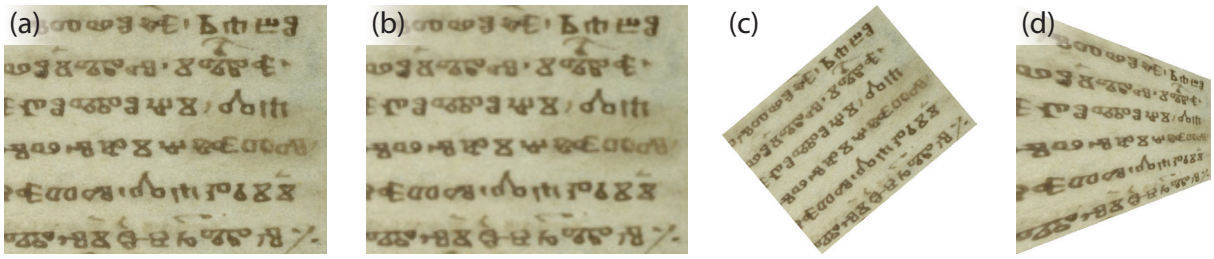

Figure 3.5: The synthetic transformations which are used to test the robustness of the detectors. Original test panel (a), scale test with 30 % of the original image size (b), rotation with an angle of 40 ° (c) and affine distortion (d).

than four test panels would slow down the time consuming evaluation. The performance of each interest point detector is computed by means of the precision which is evaluated using manually tagged ground truth data. Hence, 84 characters are used in order to compute the performance.

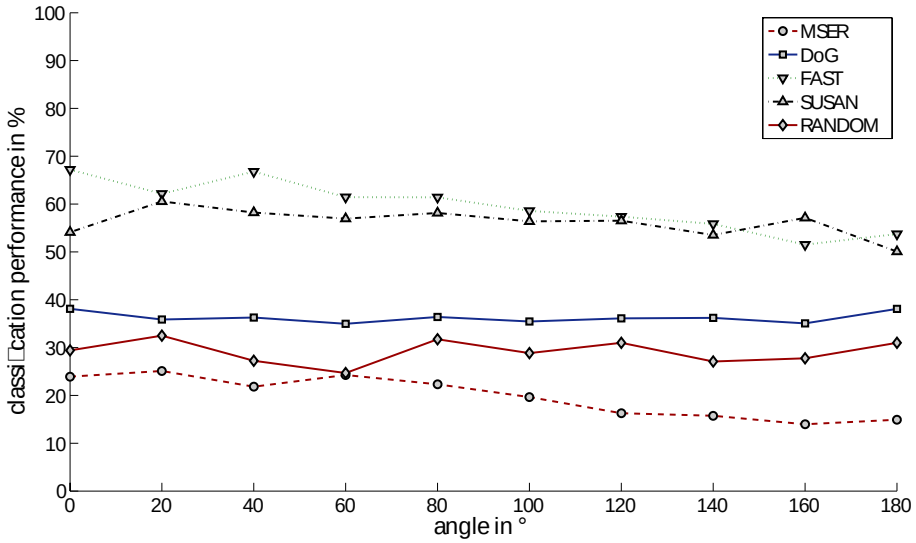

Figure 3.6: Comparison of different interest point detectors with varying rotation angle between 0° and 180°.

Rotation

The interest point detectors' invariance to rotation was tested by rotating each test panel from $0° - 180°$. The step size was chosen to be 20° so that image degradations caused by interpolations are minimized. The step size being 20 ° is a trade-off between the experiment's precision and computational performance. Figure 3.6 shows the precision of each interest point detector tested with increasing rotation angles. All interest point detectors where compared without the modifications described in this thesis.

The FAST detector (Figure 3.6) closely followed by the SUSAN detector outperforms the other interest point detectors. Nevertheless, the performance of FAST decreases with increasing angles (max: 67.2 % at 0 ° and min: 51.5 % at 160 °). The mean performance

of the DoG, which is ø = 36.3 %, is weaker than those of FAST and SUSAN. This can be traced back to the fact that the DoG is computed with a scale-space where features of a coarse scale level are mistaken for features of a fine scale level. In detail, a whole text line has the similar shape at a coarse scale-level as a horizontal stroke of a character. As illustrated in Figure 3.6, the DoG is more stable with respect to rotation than the detectors compared. The MSER has a weak performance since it locates fewer interest points on the characters than the other detectors which results in a worse training of the classifier (see Table 3.1). Notice, that the performance of MSER decreases similar to FAST as the angle increases since they are not robust with respect to rotational changes. Additionally, randomly sampled interest points were computed. They perform better than MSER due to the previously mentioned fact that more samples per training image are used to train the classifier.

DETECTOR	# IP	MEAN	STD (σ)	MIN
MSER	**124**	19.8 %	4.25 %	14.0 %
DoG	289	36.3 %	**1.09** %	35.0 %
FAST	249	59.6 %	5.19 %	51.5 %
SUSAN	200	56.2 %	2.93 %	50.1 %
RANDOM	216	29.1 %	2.47 %	24.7 %

Table 3.1: Number of interest points (# IP) per test panel, mean, standard deviation and minimal precision of all compared interest point detectors, if a test image is rotated. The precisions are averaged on all test panels.

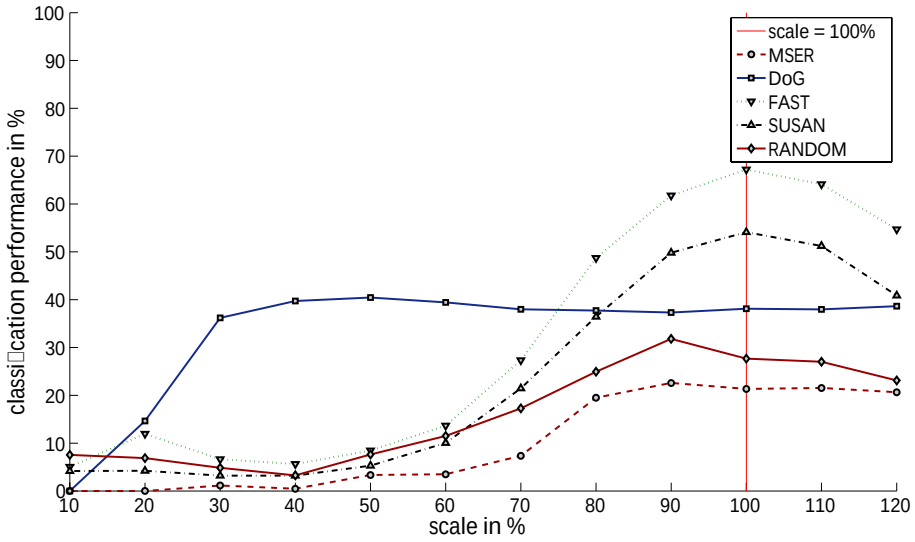

Figure 3.7: Comparison of the performance of four different interest point detectors with respect to varying image size (10 % – 120 % of the original image size). The vertical line marks the location where the test panels have the scale which is used to train the classifier.

Scale

This test setup was arranged similar to the rotation invariance test. Each test panel was resampled 12 times with a step size being 10 % of the original image size. It can be seen that FAST performs best around 100 % of the original image size. In this test setup, SUSAN performs significantly worse than FAST. Since, except for the DoG, the interest point detectors are not computed in a scale invariant manner, it outperforms all other methods. The DoG has a constant performance for image sizes larger than 30 % of the original image. Although the radius of the FAST detector could be changed, it cannot be used to extract features in a scale invariant manner [RD06]. By contrast MSER can be computed invariant to scale changes. Nevertheless, the DoG outperforms MSER on the investigated dataset because of the previously mentioned drawbacks of MSER.

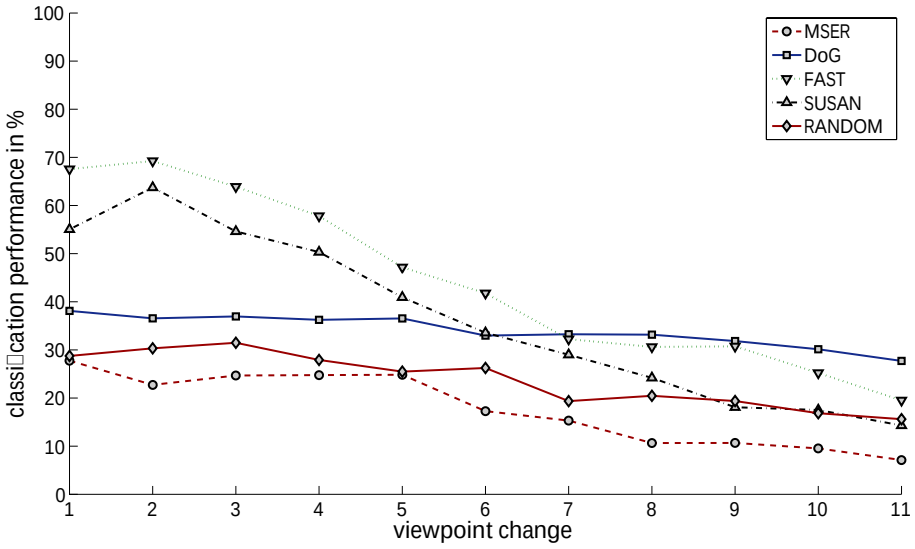

Figure 3.8: Comparison of the performance of four different interest point detectors with respect to increasing projective distortions of the investigated dataset.

Viewpoint Change

In addition to comparing the stability of different feature detectors with respect to scale and rotation, their robustness under viewpoint changes was evaluated. The viewpoint change of an image was simulated by applying an affine transformation where the horizontal and vertical axes on one side were shortened. This results in a distortion similar to changing the viewpoint angle (e.g. walking around an object). Once again, FAST and SUSAN outperform the compared detectors if the affine transformation is below 6. But while the angle of the viewpoint increases, the performance of both detectors decreases significantly faster than that of the compared detectors. Even randomly sampled interest points are more robust with regard to affine distortions than the two mentioned detectors. This can be observed by comparing the standard deviation of all detectors in Table 3.2. The DoG is the most stable detector of the methods evaluated if an image undergoes projective distortions. Its precision decreases slightly.

Detector	# ip	Mean	Std (σ)	Min
Mser	124	17.8 %	7.48 %	7.1 %
DoG	289	34.0 %	**3.24** %	27.7 %
Fast	249	44.2 %	18.00 %	19.5 %
Susan	200	36.5 %	17.41 %	14.3 %
Random	221	23.8 %	5.63 %	15.6 %

Table 3.2: Number of interest points (# ip) per test panel, mean, standard deviation and minimal precision of all compared interest point detectors, if a viewpoint change is simulated on the test panels. The given precisions are averaged on all test panels.

3.2 Local Descriptor

For each interest point detected by the DoG, a descriptor is computed which considers the structure of the neighborhood of a given interest point. The size of the neighborhood considered depends on the scale factor σ which is determined by the scale selection. The aim of a local descriptor is to maximize its distinctiveness, while at the same time maximizing its robustness to a certain set of image distortions. Obviously, the distinctiveness of a descriptor decreases, when increasing its robustness with respect to image transformations. Consider for example a Latin **d** which is similar to a Latin **p** rotated by 180 °. If the descriptor is invariant to rotation, the feature vectors of **d** and **p** would be the same.

A comprehensive test of state-of-the-art local descriptors is performed on the investigated dataset in order to choose the best performing one. According to studies of Mikolajczyk and Lowe [MS05, Low04] and to the evaluation of local descriptors, which is further explained in Section 3.2.3, Sift was chosen. It turned out that Sift performs best for the given task due to the subsequently enumerated advantages.

- Sift is a high-dimensional descriptor which leads to a high distinctiveness.
- It is robust regarding common transformations of manuscript images (e.g. rotation, scale, illumination changes).
- The distribution of gradient magnitudes and their orientation is considered, which are reliable features for recognizing characters.
- The computational effort of Sift descriptors is lower compared to similar descriptors (e.g. Gloh, Pca-Sift).
- Sift was successfully used for miscellaneous recognition tasks (e.g. [DS03, CDS+06, QMO+05]).

3.2.1 SIFT

Sift was first introduced by Lowe in 1999 [Low99] for matching different camera views of one object. He did not try to classify the feature vectors but to match features of different images. In order to find correspondences between arbitrary images of one object

the features primary need to be scale and rotation invariant. By weighting the considered image region with a 2D Gaussian, the features are additionally robust with respect to affine distortions and poorly localized interest points. They are additionally robust with respect to non-linear illumination changes, by extracting information using gradients.

The local descriptor's design was inspired by a model based on biological vision [EIP97]. Complex neurons in the primary visual cortex respond to gradients of a specific orientation and spatial frequency. But, their locations may shift within a so-called receptive field without loss of information. Using this model for a descriptor increases its robustness with respect to 3D viewpoint changes and non-rigid deformations.

Orientation Normalization

In order to achieve rotation invariance, the orientation – computed by local pixel properties – is assigned to each descriptor. This allows representing the features relative to the estimated orientation rather than computing each feature in a rotation invariant manner (e.g. *local jet*).

The orientation estimation is based on the computation of the gradient magnitude $m(x, y)$ and the gradient orientation $\theta(x, y)$ within the local neighborhood. For these computations, the smoothed image $L(x, y)$ closest to the given scale, is chosen. Additionally, a 2D Gaussian window having a σ of 1.5 times the interest point's scale, is multiplied to the gradient magnitude so that the influence of border pixels is decreased. This increases the descriptors robustness with respect to affine distortions and small variations of poorly localized interest points. Both, the gradient magnitude and the orientation, are calculated using pixel differences:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 \; + \; (L(x, y+1) - L(x, y-1))^2} \tag{3.9}$$

$$\theta(x, y) = tan^{-1} \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \tag{3.10}$$

The orientations $\theta(x, y)$ are assigned to an orientation histogram which consists of 36 bins corresponding to 360 ° (see Figure 3.10). Each orientation is weighted by the corresponding gradient magnitude $m(x, y)$ since the gradient magnitude can be seen as an information content measure of a given pixel. The orientation histogram is then smoothed by means of a 1D Gaussian kernel in order to increase the robustness of the estimation against noise. The highest bin of the orientation histogram indicates the estimated dominant orientation of a given local region. In addition, each orientation that is greater than 80 % of the dominant direction is taken as dominant rotation of a novel descriptor. Hence, if any other orientation bin lies within 80 % of the global maximum, more than one descriptor is computed at the position of the given interest point. This heuristic increases the generalization performance of the local descriptors. Finally, a 2$^{\text{nd}}$ order polynomial is fit through the 3 bins of the local maximum's neighborhood in order to accurately map the dominant orientation.

Figure 3.9 (a,b) shows the local region of a given interest point, extracted at the top of a Glagolitic Ⱁ. Calculating the gradient magnitude of the given region results in Figure 3.9 (c). The orientation is shown in Figure 3.9 (d), note the noise in the right upper corner. The resulting smoothed orientation histogram is shown in Figure 3.10. The red upper line marks the global maximum which is in this case 0.0079. The black line at 0.0063 marks

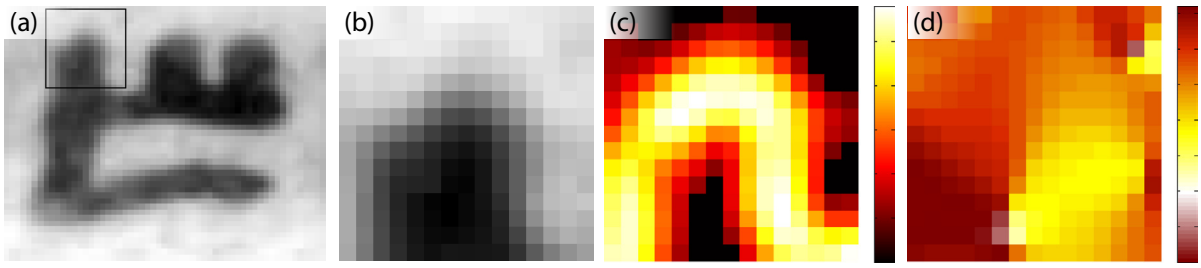

Figure 3.9: The gradient magnitude (c) and the orientation (d) of a local image region which are used for the estimation of the local orientation. The image region is taken from an image showing a Glagolitic 𞀀.

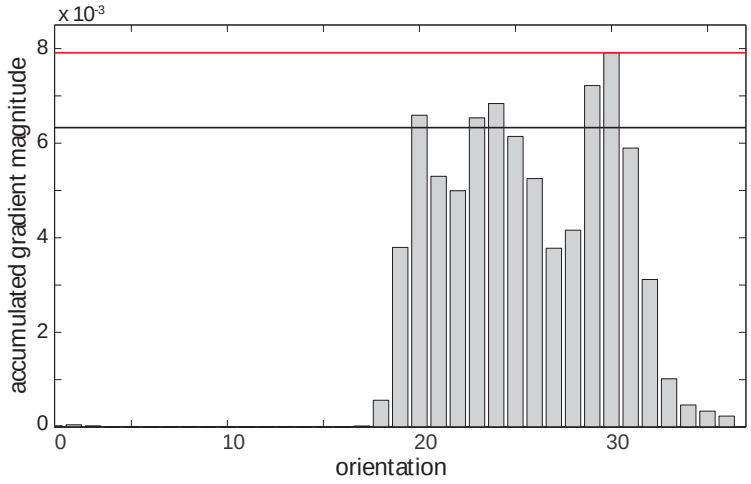

Figure 3.10: An orientation histogram with the 80 % interval denoted by the black line.

the 80 % interval where additional descriptors are created. In this particular case three descriptors having different dominant orientations are created.

Descriptor Computation

The previous sections introduced the computation of image location, scale and orientation of a given interest point. Thus, a 2D coordinate system is created which is invariant to these parameters.

The descriptor is constructed by means of the gradient magnitude $m(x, y)$ and the gradient orientation $\theta(x, y)$. First, the coordinates of a local region are rotated relatively to the orientation of the interest point. The gradient magnitudes of a local region are again weighted by a 2D Gaussian function in order to decrease the effect of gradient magnitudes at the region's border which change if the interest point is poorly localized.

Each descriptor consists of eight 4×4 orientation histograms which yield to a 128 dimensional feature vector. The orientation planes correspond to 8 different gradient orientations $(0\,°, 45\,°, 90\,°, ..., 315\,°)$. Each orientation plane has 4×4 bins which approximate the spatial distribution of the given gradient magnitudes. Then, the gradient magnitudes of a local region are trilinearly interpolated in order to avoid boundary effects. In detail,

a gradient magnitude is spatially interpolated according to its Euclidean distance to the 4 nearest bin centers. In addition, it is interpolated between the two nearest orientation histograms determined by the gradient orientation.

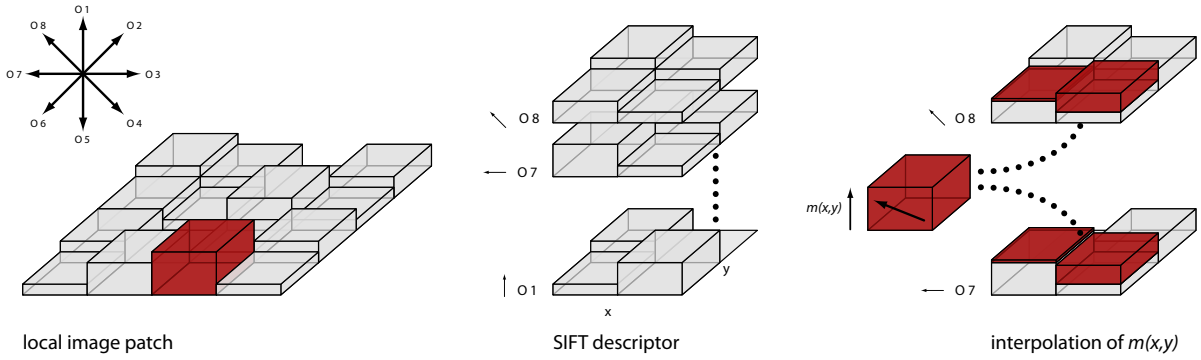

Figure 3.11: The computation of a SIFT descriptor. The cubes illustrate different gradient magnitudes. In this case eight 2×2 orientation histograms are used as feature vector. The right illustration shows the trilinear interpolation of a sample having a gradient orientation of 292.5 °.

Figure 3.11 illustrates the computation of a SIFT descriptor for a given local image patch. In the local image patch, the gradient magnitudes $m(x, y)$ at each pixel position are represented by cubes, where the height of each cube indicates the magnitude of the gradient. The descriptor consists – for the sake of simplicity – of eight 2×2 orientation planes. In this case, the cubes represent histogram bins accumulated by the local image patch. The orientation planes are labeled (O 1, O 2, ... O 8) which correspond to the orientations ($\mathsf{O} = 0\,°, 45\,°, ..., 315\,°$). Figure 3.11 (right) shows the trilinear interpolation of a gradient magnitude which is marked red in the local image patch. The sample gradient orientation is 292.5 °. Thus, it is added to the 7^{th} (270 °) and 8^{th} (315 °) orientation plane. The weights[2] for the orientation interpolation are 0.5 in this case. Furthermore, the sample is spatially interpolated with the weights 0.25 (left bin) and 0.75 (right bin).

The gradient magnitudes are not sensitive to global brightness changes since they are computed by means of pixel differences. Nevertheless, they are not robust with respect to varying illuminations of an object. In order to gain invariance to affine illumination changes, the feature vector is normalized. However, the descriptors are then still not resistant to non-linear altering illumination arising from camera saturation, or shading variations of 3D objects. This dependence is in fact reduced by thresholding large gradient magnitudes with an empirically found limit but can be neglected in character recognition applications.

3.2.2 Modifications of SIFT

The modifications of SIFT and the DoG subsequently introduced, are motivated by experiments discussed in Section 4.3.1. The threshold displayed in Figure 3.4 is set to 0.01

[2]The weights are determined by the Euclidean distance of the sample orientation to the nearest orientation planes: $1 - \frac{\sqrt{(270-292.5)^2}}{360/8} = 0.5$

compared to the proposed value 0.03. Thus, more interest points are detected in an image, which improves the probability histogram defined in Section 3.5.

In contrast to the original implementation, we do care about the difference of local maxima and local minima in the DoG space. Since local maxima represent characters (by trend black) whereas local minima are located between lines or characters. The character center estimation (see Section 3.4.1) is improved if solely local maxima are regarded in the DoG space.

Lowe [Low04] proposed to subsample the original image (double its size) in order to get interest points corresponding to the highest spatial frequencies present in an image. However, it turned out that especially these interest points corresponding to a small local descriptor are unstable throughout the classification and that they adulterate the final classification performance. This modification improves the recognition while at the same time reducing the memory consumption of the final software.

Furthermore, the rotation invariance of SIFT is disabled up to 180 °. Thus, the same structure rotated by 180 ° results in a different descriptor which increases its distinctiveness (see Section 3.2). The dependence on rotation is achieved by:

$$\theta = \theta - \pi \qquad \forall \theta > \pi \tag{3.11}$$

where θ is the main orientation of a given local descriptor. If additionally $\frac{\pi}{2}$ would be subtracted, the local descriptor would be sensitive to rotational changes up to 90 °. However, tests showed that the performance is decreased then. In Figure 3.12 a Glagolitic ⰹ is illustrated, which is a Glagolitic ⰻ rotated by 180 °. As can be seen, the interest points are located in the center of circles, at corners and at junctions. The highlighted local descriptor is once computed rotationally invariant and once with a rotational dependence up to 180 °. The histograms in the second row are down-sampled local descriptors for a more intuitive visualization. It can be seen in the second row of Figure 3.12 that the descriptors are similar[3] to each other if the features are computed rotationally invariant. In contrast, when the rotational invariance is discarded, the same local descriptor produces a mirrored vector[4] for the ⰹ and ⰻ. This is why the rotational dependence improves the system's performance.

3.2.3 Comparison of Local Descriptors

Similar to the interest point detectors, described in Section 3.1, the performance of five state-of-the-art local descriptors (namely: SIFT, SURF, GLOH, PCA-SIFT and gradient moments) is evaluated on the investigated dataset. These local descriptors where chosen since they performed best in Mikolajczyk's performance evaluation [MS05]. Except for SURF which was selected to demonstrate the performance of approximated high dimensional features. It was developed in 2006 which is after the performance evaluation of local descriptors.

This evaluation incorporates the same test setup as the comparison of interest point detectors described in Section 3.1.2. Again, the robustness of the local descriptors with

[3] Absolute difference: 0.155, $r = 0\,°$

[4] Absolute difference: 10.39, $r = 180\,°$

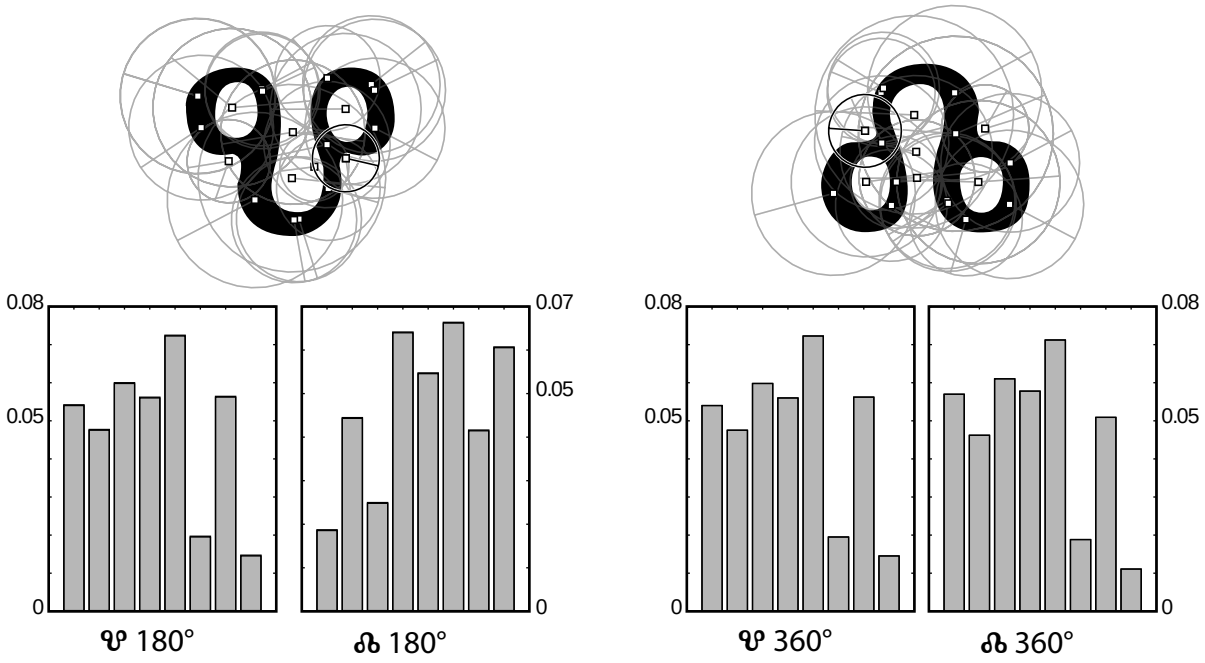

Figure 3.12: A Glagolitic ⰹ and ⰻ with their local descriptors (first row). The down sampled features computed rotationally invariant (right) and with rotational dependence up to 180 ° (left).

regard to a certain set of affine transformations (particularly: scale, rotation, projective) is evaluated. In order to demonstrate the effect of the feature vector on the classification performance, the interest points were computed using the DoG detector. Since training and testing for all local descriptors is done with the same interest points, the varying results can be traced back to the weaknesses and strengths of the different local descriptors. Indeed, the classifier could possibly influence the results. In order to minimize this effect, all tests are carried out with the same SVM having one RBF kernel. The classifier's parameters (γ, C) are estimated individually by means of a three-fold cross-validation.

When classifying features, the vector's dimensionality needs to be considered. Hence, the higher the feature dimension, the more training samples are needed to guarantee a generalization of the classifier (see Section 3.3). But then, high-dimensional feature vectors have a higher distinctiveness than low-dimensional features [MS05]. However, SVMs are based on statistical learning theory rather than empirical risk minimization. That is why they have a generalization even if they are trained with few samples of high-dimensional classification problems. To demonstrate this fact, PCA-SIFT is tested using the first 128 eigenvalues and with the first 36 eigenvalues as proposed by Ke and Sukthankar [KS04]. Additionally, gradient moments[5] are evaluated to show the performance of low-dimensional features. The high-dimensional descriptors are chosen on the one hand because they are new (SURF) and on the other hand due to their good results (SIFT and GLOH) in Mikolajczyk's performance evaluation [MS05].

[5]Gradient moments performed best, of all low-dimensional features, in [MS05].

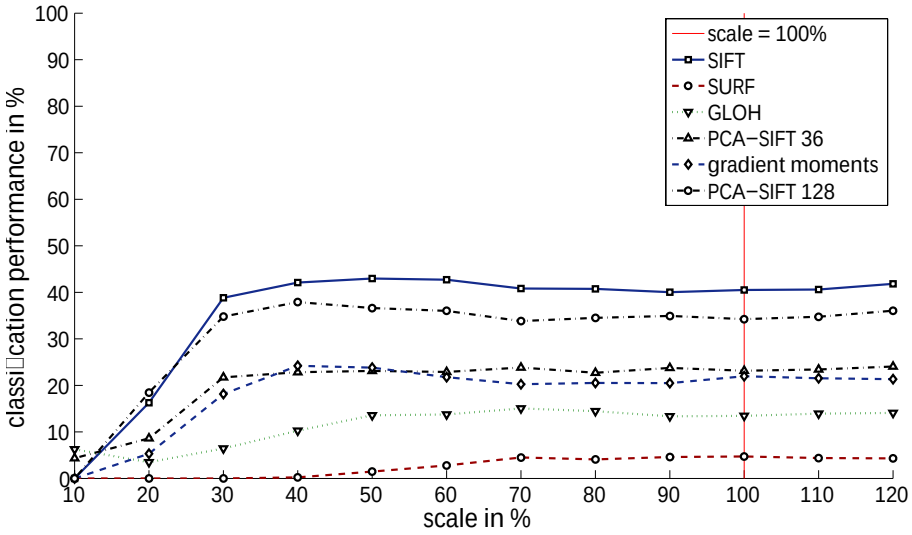

Figure 3.13: Comparison of four different local descriptors with respect to varying image size (10 % – 120 % of the original image size). The vertical line marks the image scale which was used for training the classifier.

Scale

The abscissa of Figure 3.13 shows the changing image scales in 10 % steps. The red vertical line marks the image scale used for training the classifier. In general, all descriptors have a similar robustness with respect to scale changes because it mainly depends on the scale selection scheme which is implemented in the interest point detector algorithm.

It can be seen that SIFT has the highest precision which is $\overline{x} = 35.6\,\%$. The 128 dimensional PCA-SIFT descriptor performs similarly. In contrast, the 36 dimensional PCA-SIFT has a worse performance ($\overline{x} = 20.4\,\%$) which is not significantly different to the second low-dimensional local descriptor (gradient moments). The performance of GLOH increases slower with respect to scale changes and reaches its mean performance at 50 % of the original scale regarding the other descriptors which reach the mean performance at 30 %. The worst results on the investigated dataset are obtained by SURF. This can be ascribed to the fact that the descriptor is highly dependent on the proposed Fast-Hessian detector [BTG06] as it performs significantly better if this detector is used instead to the DoG (see Section 3.2.4).

Rotation & Affine

Since the robustness regarding rotation and affine transformations depends more on the interest point detector than on the descriptor, just a brief summary of these test results is given below. Figure 3.14 (left) shows that all evaluated descriptors are invariant to rotation (maximum standard deviation: $\sigma = 1.01\,\%$). The ranking of the mean classification performance is headed by SIFT ($\overline{x} = 38.8\,\%$) and PCA-SIFT 128 ($\overline{x} = 33.7\,\%$). In the center span, the low-dimensional descriptors gradient moments ($\overline{x} = 23.2\,\%$) and PCA-SIFT 36 ($\overline{x} = 22.6\,\%$) are located. Contrary to the expectations, GLOH performs poorly within the proposed system and in combination with the DoG, having a mean classification performance of ($\overline{x} = 15.0\,\%$). But the worst results are achieved with SURF

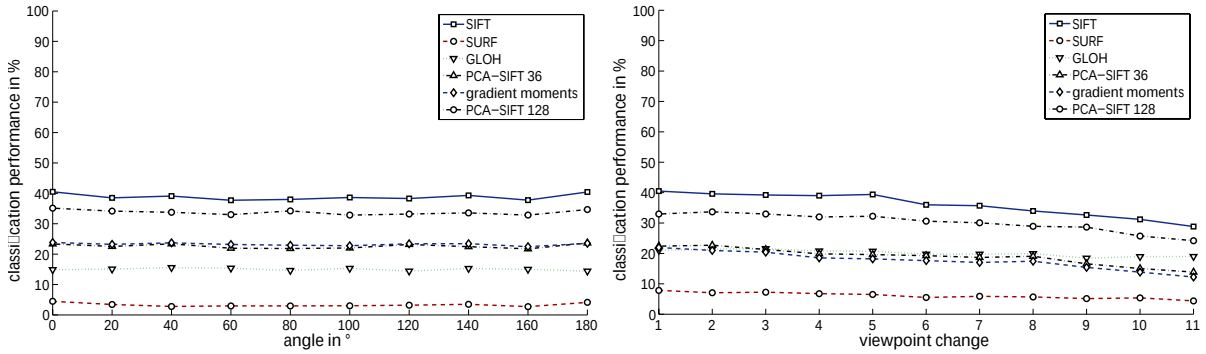

Figure 3.14: Comparison of different local descriptors with varying rotation (left) and projective distortions (right).

($\overline{x} = 3.3\,\%$) in combination with the DoG.

Testing the robustness of local descriptors with respect to affine distortions results in a similar ranking as the rotation evaluation (see Figure 3.14 (right)). The only remarkable result is here obtained by GLOH which was especially designed to handle affine distortions [Mik02]. Table 3.3 shows that GLOH has the smallest standard deviation ($\sigma = 1.05\,\%$) which supports the conclusion that the descriptor is in fact more robust with respect to affine transformations than the other descriptors evaluated. In these tests, SURF has a similarly small standard deviation. However, this number results from the poor performance of SURF and cannot be used to draw any conclusion about the characteristics of the descriptors.

DESCRIPTOR	# IP	MEAN	STD (σ)	MIN
SIFT	1600	36.0 %	3.92 %	28.8 %
SURF	1600	6.1 %	1.04 %	4.4 %
GLOH	1600	20.1 %	**1.05%**	18.4 %
PCA-SIFT 36	2161	18.9 %	2.82 %	13.9 %
gradient moments	2161	17.6 %	2.96 %	12.2 %
PCA-SIFT 128	2161	30.2 %	3.08 %	24.2 %

Table 3.3: Number of interest points (IP) evaluated, mean, standard deviation and minimal precision of all local descriptors compared, in respect of affine transformations. The precisions are averaged on all test panels.

3.2.4 Comparison of Local Feature Systems

The previous evaluation was setup to precisely show the characteristics of different local descriptors if embedded in the proposed system. Due to the strong dependence of some descriptors (e.g. SURF) to the proposed interest point detectors, an additional evaluation was done, where the whole systems – proposed by the respective authors – are tested on Glagolitic manuscript images. Table 3.4 shows the local descriptors tested, the corresponding interest point detectors and the papers which first introduced the systems.

Descriptor	Detector	Reference
Sift	DoG	[Low99]
Surf	Fast-Hessian	[BTG06]
Gloh	Harris-Laplace	[Mik02]
Pca-Sift	DoG	[KS04]
gradient moments	Harris-Hessian-Laplace	[GMU96]

Table 3.4: Local Descriptor, corresponding interest point detector and the respective paper of the local descriptor systems evaluated on the investigated dataset.

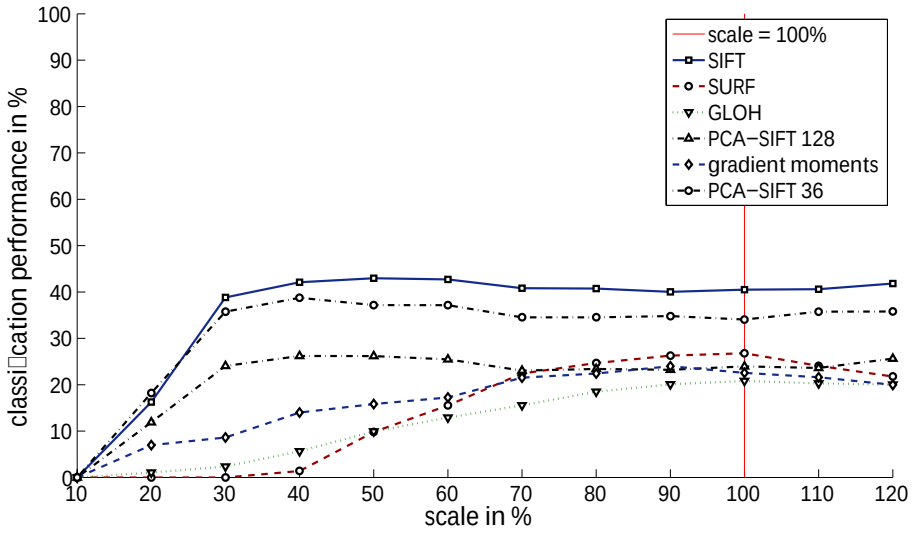

Figure 3.15: Comparison of six different local descriptor systems with varying image size (10 % – 120 % of the original image size). The vertical line indicates the image scale used for training the classifier.

Scale

Even though the local descriptors are computed with their particular interest point detector, Sift still performs best on this dataset. Gloh and gradient moments have – in combination with their detector – even a lower scale adaption (at $\approx 70\,\%$) compared to the previous tests carried out using the DoG. On the contrary, Gloh performs better at scales nearby the trained scale (max: 20.82 % compared to max: 15.05 % in the previous test). As mentioned before, Surf performs significantly better (up to 22.07 %) in combination with the proposed Fast-Hessian detector. Due to the approximations made (e.g. integral image) the Fast-Hessian detector is not scale invariant, but robust regarding scale changes. This is clearly illustrated in Figure 3.15, since the performance is about 0 % between $0 \prec$ and 40 % of the original scale, where other descriptors such as Sift have already fully adapted to the changing scale.

Rotation & Affine

Regarding Figure 3.16 (left), it can again be observed that the interest point detector is the most important factor for the feature's robustness regarding image transformations.

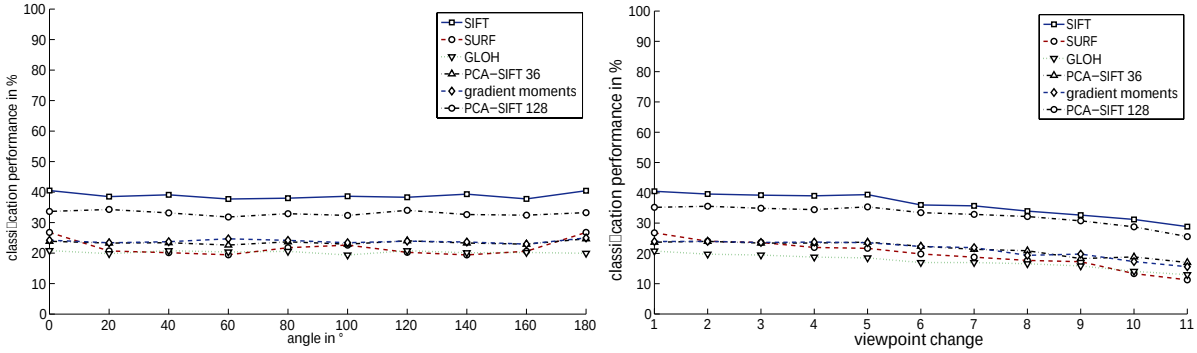

Figure 3.16: Comparison of six different local descriptor systems with varying rotation (left) and projective distortions (right).

In contrast to this illustration, the classification performance is almost constant in Figure 3.14 where the descriptors were evaluated using the same interest point detector. The Fast-Hessian, which is used for the computation of SURF, is solely invariant for orthogonal angles ($0\,°, 90\,°, 180\,°$...). Therefore, the classification performance decreases significantly when applying image rotations with different angles. The other descriptors have the same performance as in the previous tests, except for GLOH which has – beside SURF – an improvement in performance of $\overline{x} = 5.27\,\%$ on average.

The evaluation of the descriptor system's robustness with regard to projective transformations is given in Figure 3.16 (right). Analogous to the previous test, the Fast-Hessian detector has the highest performance decrease (standard deviation $\sigma = 4.64\,\%$) as the affine distortions are increased. The Harris-Laplace detector (without affine adaption) is less robust than the DoG. This can be observed when the standard deviation of the previous test ($\sigma = 1.05\,\%$) is opposed with that of the current test ($\sigma = 2.41\,\%$).

The performance tests introduced on Glagolitic manuscript images show, that SIFT performs best for the given task and is robust with respect to common image transformations that need to be considered when recognizing characters. This can be attributed to the fact that SIFT is high-dimensional – therefore highly distinctive – and that new approaches such as SURF focus on computational speed, not accuracy.

3.3 Classification

Having computed the local descriptors of a given manuscript image, each non-planar image region is described by a high-dimensional feature vector. For the character recognition, the local descriptors are classified by means of a one-against-all scheme. Thus, one classifier is trained per character class, resulting in 25 classifiers. Additionally to the labels predicted for local descriptors, a probability is assigned to the descriptors by each classifier resulting in a probability histogram. This strategy has two major advantages. On the one hand, the classifier is not too sensitive to noise in the training data as the criterion function is less complex when two class labels are assigned (e.g. ·|·, **not** ·|·). On the other hand, probabilities – needed for the subsequent voting – can be solely computed for two classes.

For that purpose, a simple k-NN or a Bayes classifier could be considered. However, both classifiers have a major drawback. The k-NN classifier is capable of classifying a dataset which is not linearly separable by assigning the most probable class label to an unknown data point according to the labels of the k nearest neighbors in the training set. Hence, it is suitable for handling complex input data with few training data. Nevertheless, it is a well-known fact that the k-NN tends to overfit the training data. The Bayes classifier, on the other hand, finds an optimal solution for a given classification problem by maximizing the a-posteriori probability of an unknown sample. This is equivalent to minimizing the classification error on the training set. Yet, if the training set does not well approximate the true data, both listed classifiers fail. By contrast, the SVM rather minimizes the overall risk than the overall error of a training set, which results in a good generalization performance even for high-dimensional features.

3.3.1 Support Vector Machine

The Support Vector Machine was introduced by Vapnik and Chervonenkis in 1974 [VC74]. As previously mentioned, the SVM is based on statistical learning theory which considers the difference between the empirical risk and the true overall risk. Thus, the size of the training data and the model complexity are incorporated.

Compared to a Perceptron, the SVM does not search for any solution (separating hyperplane) of a given problem. It rather finds the optimal hyperplane having a maximal margin to both classes. The margin $1/\|w\|$ is defined as the minimum distance of a feature vector to the separating hyperplane. This formulation leads to a dual optimization problem. Since the optimization criteria are convex, they can efficiently be solved by Lagrange multipliers. To solve the optimization problem, solely support vectors – generally a small subset of the input data – need to be considered. Support vectors are feature vectors which are located on the margin or – in case of non-linear separability – on the wrong side of the hyperplane.

Figure 3.17 shows a SVM for a 2D training set consisting of two classes. The optimal margins are illustrated with dashed lines, additionally, the support vectors are marked by a dark circle.

3.3.2 Radial Basis Function

The linear SVM was extended to a non-linear classifier by Boser et al. [BGV92] in 1992. Therefore, the dot product of the feature vectors $(x_i^T x_j)$ in the criterion function are replaced by kernel functions. Thus, the feature space is transformed to a higher dimension. There, a hyperplane is computed according to the previously mentioned scheme and then, the feature space is re-transformed to the input space. This results in a non-linear classifier in the input space, as the transformation was non-linear. Due to the kernel trick which was proposed by Aizermann et al. [ABR64], the higher dimensional space does not need to be evaluated explicitly, since the inner product can directly be computed as a function.

The RBF kernel is defined by:

$$k(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2} \quad \gamma > 0 \tag{3.12}$$

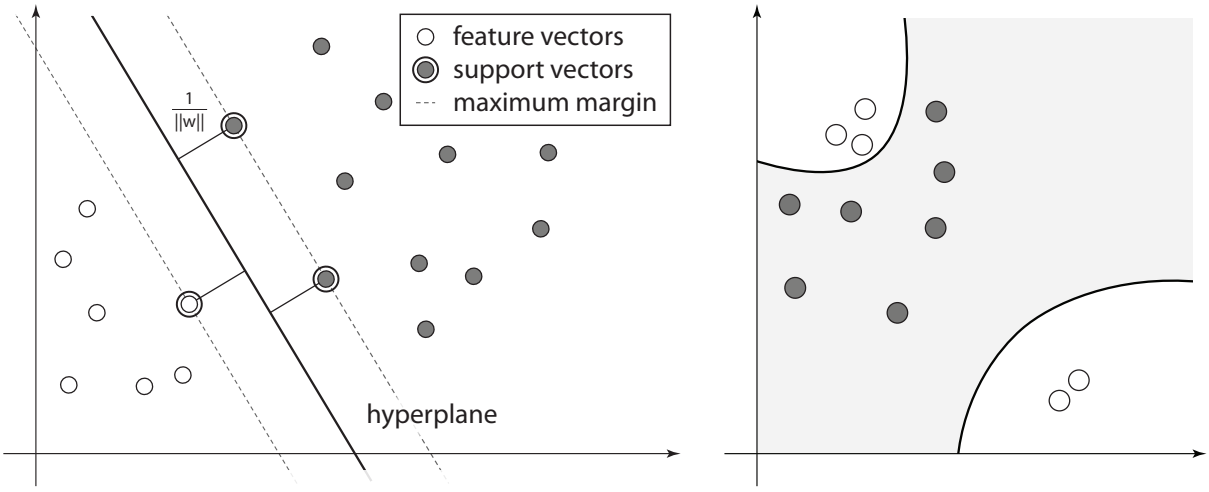

Figure 3.17: Linear SVM (left) with the optimal hyperplane (black line) and the maximal margin (dashed lines). SVM (right) with an RBF kernel.

where x_i and x_j are feature vectors and γ is a parameter which needs to be determined using cross-validation. The RBF, kernel has the advantage, that solely one parameter needs to be determined while at the same time being flexible enough to handle complex training sets.

In Figure 3.17 the hyperplane of a SVM which has an RBF kernel is shown. For this illustration, γ was chosen to be 5.

3.3.3 Training

The supervised learning is carried out with 20 sample images per character class which were manually extracted from the codex and tagged. The parameters γ, C are determined for each character class individually by means of 3 fold cross validations. The parameter γ introduced in Equation 3.12 controls the sensitivity of the kernel function. The cost parameter C controls the flexibility of the classifier. If it is set too high, the model perfectly fits the training data which reduces its generalization performance.

For the three-fold cross-validation, the training set is split into 3 subsets. Then, the classifier is trained on one of the respective subsets and validated with the remaining subsets. This process is carried out for all subsets and classifiers with changing parameters γ, C. Finally, a grid is obtained with classification performances for each tuple. Then the classifier is trained on the whole training set using parameters which maximize the three-fold cross-validation. This algorithm guarantees, that the RBF kernel is properly adapted to the given classification problem.

In Figure 3.18 (left) the classification performance of the cross validation is given for varying parameters γ, C. For this kernel, the maximum, being 96.98 %, is achieved for the tuple $\langle \gamma, C \rangle = \langle 2.6, 8 \rangle$ which is used for finally training this SVM. It can be seen, that the performance decreases with γ (e.g $\gamma = 0.1$). This results from the fact that the decision boundary gets more rigid when γ is decreased. In addition, the number of training features influences the decision boundary when γ is set to a small value. If for

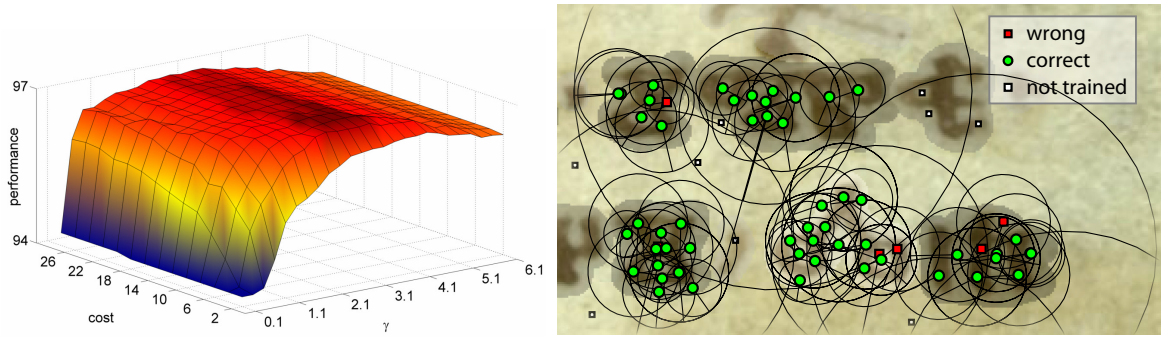

Figure 3.18: Cross-validation of the SVM kernel for the character Ɔ (left). The maximum performance for this kernel is achieved with $\langle \gamma, C \rangle = \langle 2.6, 8 \rangle$. Classified local descriptors (right). Note that most false classified descriptors have a large or a small scale.

example 0.1 % of the features are ⊦ and the rest is **not** ⊦, then the SVM would classify all samples as not ⊦.

A five- or seven-fold cross-validation can be used to determine the classifier's parameters. Tests showed that, indeed, the absolute classification performance increases if the training set is split more than 3 times. This arises from the fact that more samples are presented to the classifier as the splitting is increased. Nevertheless, the relative performance over all parameter tests does not change which results in the same local maxima and, therefore, the same values for both parameters.

Figure 3.18 (right) shows a test panel which was manually tagged (gray blobs). After the classification step, a label is assigned to each local descriptor according to the highest probability. The figure shows correctly classified (green circles) and false (red rectangles) local descriptors. Additionally, the scale of each descriptor is illustrated by a black circle. In this case, 25 character classes were trained and the classification performance on this test panel is 76.9 %.

3.4 Character Localization

For traditional OCR engines, the characters or words are localized implicitly in the binarization step. If handwriting OCR engines are considered, an additional character segmentation step needs to be performed in order to detect concatenated characters. In contrast, the system proposed does not incorporate information about the positions of characters in a given image to the point of feature classification. Indeed, the positions of the classified features are known, but as a feature does not necessarily represent a whole character, its position and size is unknown.

The character localization is based on clustering the interest points. This approach benefits from the fact, that degraded characters are detected with local descriptors but not considered when the image is binarized. Thus, even degraded characters can be localized. Another advantage is the low computational complexity, since solely the interest points are considered (e.g. for a 436 $px \times 992$ px image that has a total of 432512 px, 1543 interest points are detected).

3.4.1 Character Center Estimation

The k-means clustering cannot estimate the number of clusters k. In order to determine the number of clusters, which is in this case equivalent to the number of characters, a cluster validity index can be used [BP98, HBV01]. However, experiments showed that the combination of different cluster validity indexes does not work for this task. This arises from the fact that the text line spacing is greater than the between-character spacing. Hence, a cluster analysis would group lines, not characters.

Scale Estimation

To overcome this problem, the scales of interest points are exploited. There exists at least one interest point that represents a whole character. In other words, each character produces a single local maximum in a certain scale level. In order to remove interest points that represent lines, solely interest points resulting from positive local maxima in the DoG scale space are considered (characters are generally darker than background). Extracting this information, the parameter k of the k-means can be estimated and, at the same time, initial cluster positions are obtained that improve convergence. However, the scale levels representing a character need to be extracted in a scale invariant manner.

In order to find the minimum scale level of interest points that represent a whole character, the scale distribution of all interest points in a given image is regarded. Figure 3.19 shows the interest point's scale distribution. There, the abscissa represents increasing scales, particularly the radius of interest points, measured in px. The ordinate gives the number of interest points corresponding to the scale interval. It can be seen that most interest points are detected in scale levels below 30 px. This results, on the one hand, from the higher resolution which decreases with respect to increasing scale and on the other hand from the fact that manuscripts have high frequency features such as endings, junctions and corners. The scale levels corresponding to characters – which we are interested in – are within the second peak between 30 px and 80 px. The third and last peak corresponds to interest points that represent text lines or low frequency features such as illumination changes or stains.

Indeed, the intervals are fuzzy, which precludes the use of a sharp threshold. The interest point's scale of a small character is the same as the scale of an interest point that represents a structure of a larger character.

Since the interest points' scale distribution is similar for all manuscript images, independent of the image resolution, the number and localization of characters can be obtained by a simple algorithm. First, the scale distribution is normalized and smoothed by a Gaussian filter kernel ($\sigma = 3$) that removes noise. Afterwards, the first peak is located by means of the second derivation:

$$s'_\sigma(x) = \operatorname{sgn}(s_\sigma(x) - s_\sigma(x-1)) \tag{3.13}$$

$$s''_\sigma(x) = s'_\sigma(x) - s'_\sigma(x-1) \tag{3.14}$$

where s_σ represents the smoothed scale distribution and sgn is the Signum function. Thus, the first peak p_s is obtained by:

$$p_s = \min\{\ x \mid s''_\sigma(x) = 2\ \} \tag{3.15}$$

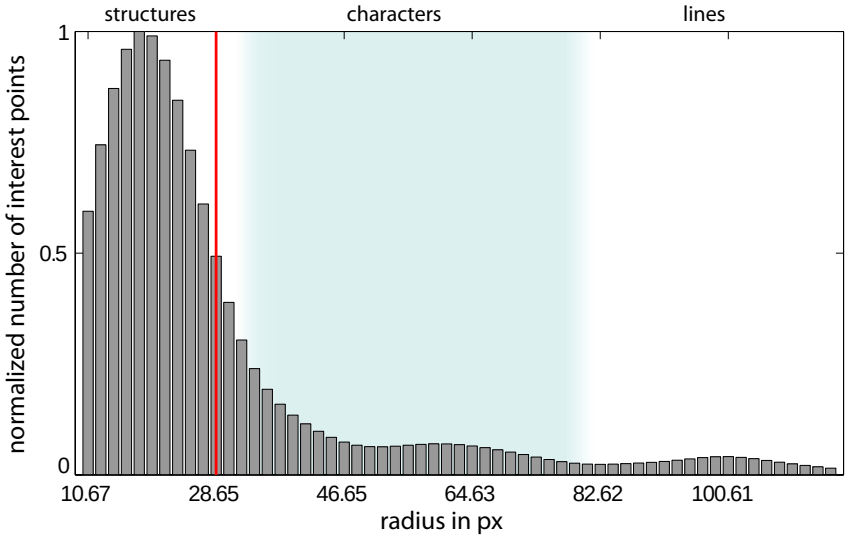

Figure 3.19: Interest points' scale distribution of a manuscript image. Three characteristic peaks can be seen, which represent high frequency structures, characters and lines. The red vertical line marks the minimum scale level of an interest point in order to be selected for the cluster initialization.

Then, the minimum scale level (rendered as a red vertical line in Figure 3.19) is defined as the first bin having a higher index than the peak that is below a given threshold s_t. The threshold is evaluated on the dataset, which is further explained in Section 4.3.1.

This algorithm guarantees that even small characters[6] are localized for the k-means initialization. Generally, more interest points are selected than characters are present in an image. This relies on the fact that background clutter – which produces interest points – is clustered together with characters, if too few initial cluster centers are obtained.

Cluster Center Refinement

The interest points that represent characters are now selected. However, more than one interest point still represent one large character or more than one interest point is at the same location according to changing main orientations. In order to overcome these erroneous localizations, a heuristic was developed that exploits the area of influence. Each interest point's region of influence is estimated by a circle having a radius that corresponds to the point's scale. Thus, interest points having a smaller Euclidean distance to their nearest neighbor than the radius of the smallest scale selected in the scale estimation process are regarded. They are erroneous due to the causes afore mentioned and could therefore simply be deleted. But since a correct character localization method significantly improves the k-means, it proved to be better if the erroneous interest points are linearly interpolated.

One could think about changing the Euclidean metric to one that weights the distance by a determined orientation. This would guarantee that interest points are not interpolated across text lines. But the manuscript page's orientation had to be estimated.

[6] 27 $px \times 34$ px compared to other characters which have 93 $px \times 33$ px

Figure 3.20 shows the initial cluster centers (white rectangles). Multiple interest points representing one character are denoted by red circles and the corresponding interpolated points are white circles. As can be seen, the interpolation solely needs to be done for large characters such as the Glagolitic ⰑⰓ. Interpolated interest points with no erroneous points nearby, are those with multiple orientations. Note that this algorithm does not detect all characters at the image border (e.g. ⰱ in the last text line). This results from the border effect of the convolution which especially discards interest points having a high scale level ($> 30px$).

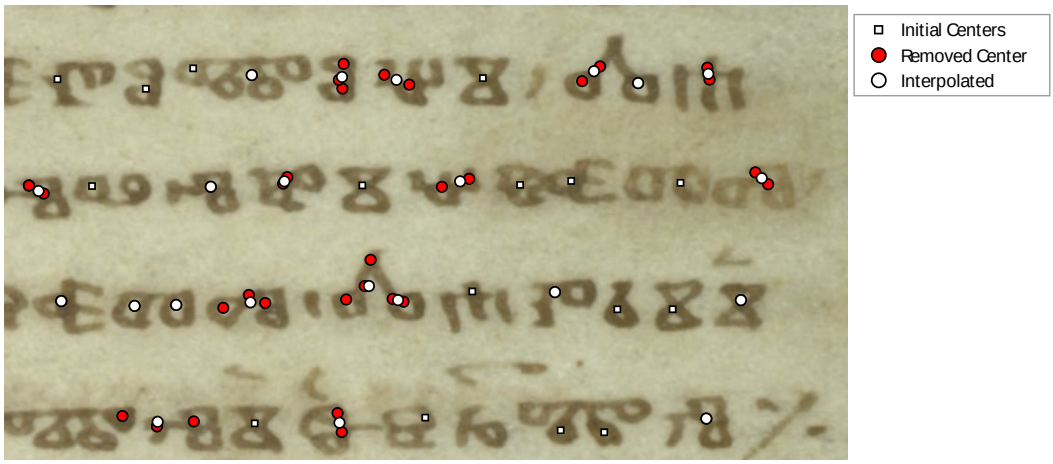

Figure 3.20: Estimated cluster centers. Removed centers are displayed as red circles, interpolated are white circles

3.4.2 Interest Point Clustering

As mentioned before, the interest points are clustered using k-means clustering. This method was first introduced by Stuart P. Lloyd [Llo82] and further studied by J. MacQueen [Mac67].

First, cluster centers are initialized by randomly choosing k vectors of the dataset. David Arthur [AV07] showed that the k-means can be improved if the seeding points are not chosen randomly. In the system proposed, the seeding points are chosen according to the method explained in Section 3.4.1.

The problem of k-means is to minimize the potential function:

$$\Phi = \sum_{x=0}^{n} \min_{c} \|x - c\|^2 \tag{3.16}$$

for k centers c where x are samples. Having found those centers, samples are grouped according to their minimum distance to the cluster centers. The solution of this problem is NP-hard. Thus, the k-means is a local search method that does not guarantee to find the optimal solution. The heuristic algorithm consists of two steps that are altered until convergence:

1. Initialize the centers c_i for $i = 1...k$.
2. **Assign** C_i all samples that are closer to c_i than to c_j for $\forall j \neq i$.
3. **Update** c_i to be the center of mass of all points in C_i.
4. Repeat step 2 and 3 until convergence

where C_i denotes a cluster (group of samples) and c_i the cluster center. The k-means converges when no cluster center c_i changes its position in the **update** step. By this means, the interest points are grouped together so that they represent the characters. Particularly, the position and an approximate size of the character are determined. Having grouped the previously classified interest points, a simple voting scheme can be performed to finally assign the character labels.

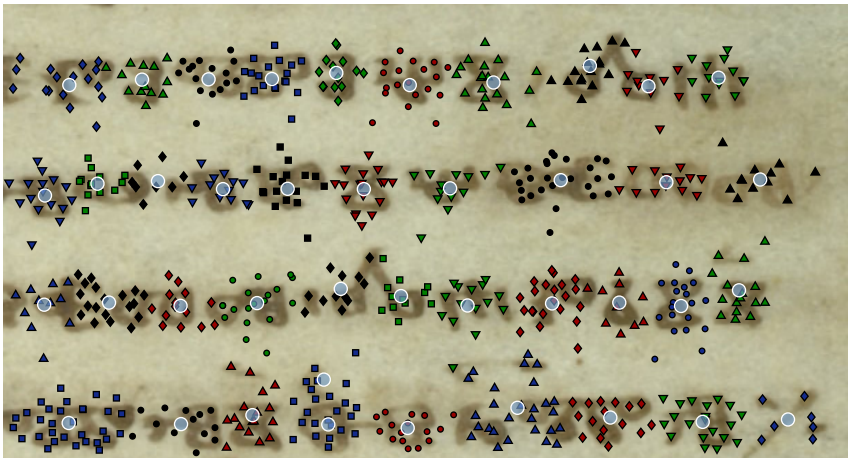

Figure 3.21: Interest point clustering. The shape and color of interest points denotes their belonging. Blue circles having a white contour represent the final cluster centers.

In Figure 3.21, the interest point clustering is displayed. The blue circles with white contours represent the final cluster centers. The markers' shape and color indicate the interest points' clusters. It can be seen that some large characters like the Glagolitic ⰳ (third character of the first line) have more than one cluster. As a result of the border effects, mentioned in Section 3.4.1, one character (ⰱ) is fused with its neighbor (ⰳ) in the last line.

3.5 Feature Voting

For the final character classification, a voting scheme is applied. Therefore, all local descriptors of a cluster C_i are considered. Each descriptor was previously classified (see Section 3.3). Hence, a probability histogram exists that indicates the class likelihood of each descriptor in the cluster. Accumulating these histograms, the maximum bin indicates the most probable class label.

Figure 3.22 shows the final probability histogram of two degraded characters. Each bin of the histograms represents one of the previously trained character classes. The bin's

height indicates the probability of a character belonging to the respective class. The left character is classified correctly, having a significantly high class probability. In contrast, the probability histogram of a false classification is given in Figure 3.22 (right). There, three class probabilities are similarly high.

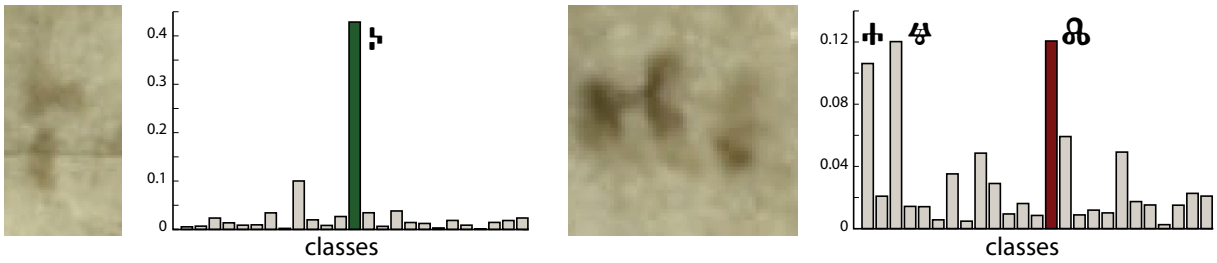

Figure 3.22: Probability histogram of two character clusters. A correct classification (left) and a false classification (right). There, the probability is similarly high for three classes, one of which is the correct (ᚼ).

Descriptor Weighting

Directly averaging the descriptors' probabilities has drawbacks. First, descriptors which are larger than a character describe the structure of more than one character. Additionally, descriptors of background clutter are falsely clustered to characters. These incorrect descriptors adulterate the performance if direct averaging is applied. That is why a weighting function is developed that regards these observations.

According to the previously mentioned observations, descriptors that are larger than characters should have a low weight. That is why a weight is established, that linearly depends on the descriptor's scale:

$$w_i = 1 - \frac{s_i}{\max\limits_{j=0\ldots n}(s_j + c)} \tag{3.17}$$

where s_i is the ith descriptor's scale and w_i is the final weight. The constant $c > 0$ guarantees that the weight w_i is > 0 for all descriptors. Similarly, the descriptors are weighted according to their distribution within the character cluster. Instead of the scale s_i, the descriptor's distance d_i to the cluster center is regarded. It turned out, that a robust cluster center improves the weighting compared to the default center-of-mass. This is because the robust center penalizes outliers (the center-of-mass shifts towards outliers). The robust cluster center is defined as the median of all x, y coordinates in a particular cluster.

Detecting Weak Clusters

In addition to the classification, the probability histogram can be used to estimate the weakness of a character cluster. Therefore, the maximum class bin m_b is considered. If another bin exists that has a higher probability than $m_b = 0.875$, it can be assumed, that the character cluster is weak (e.g. background clutter, false classification). This method allows improving the precision and therefore the F_1-score. The false classification

in Figure 3.22 (right) would be rejected since two bins are greater than 87.5 % of the maximum bin (see Section 4.3.1).

Summary

In this chapter, the methodology was discussed in detail. A character recognition system consisting of two major steps namely localization and classification was introduced. This system is especially designed for ancient manuscripts, as binarization does not need to be performed. The features which are extracted in a scale invariant manner are computed by means of the image's gray value information. In order to choose the best performing interest point detector and local descriptor, state-of-the-art methods were compared on the investigated dataset. In addition, the training and the validation of the classifier was discussed. Since there solely exist character localization methods based on binarization, a new method was introduced that allows for localizing characters by means of the interest points extracted. Finally, a voting scheme that is able to cope with uncertainty was proposed.

The subsequent section shows experiments that were carried out on the *Cod. Sin. Slav.* 5N. It is intended to show the strengths and the weaknesses of the approach proposed.

Chapter 4

Results

In this chapter, the system introduced is evaluated. It is intended to empirically evaluate the system by manually annotated real world data and synthetically generated data. The subsequent experiments show the strengths and drawbacks of the new character recognition methodology proposed in this thesis. Three different experiments were carried out in order to analyze certain aspects which are detailed subsequently. Figure 4.1 illustrates the three test setups. The number of characters for training and testing as well as the number of classes evaluated are shown for each dataset.

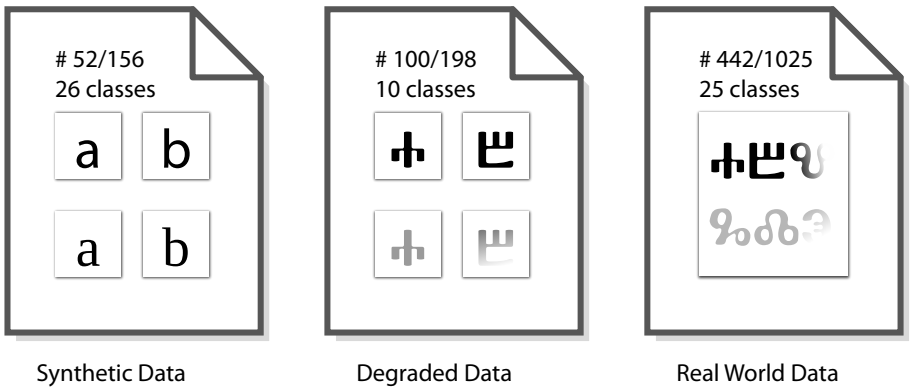

Figure 4.1: The three datasets used to evaluate the system with the number of characters for training/testing and the number of character classes.

The first experiment is performed using synthetic data. Therefore, Latin text is generated with varying fonts. In addition, noise is added to the synthetically generated images in order to show the system's robustness regarding image degradation. In one experiment, white Gaussian noise with varying standard deviation σ is added. The second test setup aims at analyzing the effect of partially faded-out characters. For that purpose, a gradient that removes the character's parts is introduced. It is shown in Section 4.1 that the system is capable of classifying characters correctly even if parts are occluded.

In addition to experiments on synthetic data, degraded characters were extracted from the investigated dataset. The evaluation discussed in Section 4.2 aims on the one hand at showing the system's performance when degraded characters are present in manuscripts. On the other hand an evaluation is given that solely considers the classification step. Thus,

errors introduced by the character localization are not considered in this experiment. In order to show the performance decrease resulting from degraded characters, a second data set is evaluated that contains intact characters similar to those used for training the SVM. A class confusion matrix that is computed on both datasets allows for analyzing the errors on the respective character classes. It shows which topological structures are likely to be mistaken.

In Section 4.3, the system is evaluated by means of manually annotated ground truth data. In these experiments, the parameters incorporated are evaluated on the test data. A discussion is then given about which parameters need to be adapted if the system is applied on different manuscripts or writing systems. Finally, results of the system on real world data are presented in Section 4.3.2. Using a synthetic character localization that was especially designed for the evaluation allows for an exact error computation on both major steps (classification, localization) individually. Additionally, statistics are given that show the character class occurrence and classification performance of different character classes.

4.1 Experiments on Synthetic Data

Before experiments are carried out on the challenging dataset investigated in this thesis, tests are performed on synthetic data. The data generated contains Latin fonts. This is done on the one hand, to demonstrate the system's capability of recognizing different writing systems. On the other hand, using Latin script allows for experiments with different fonts. Another reason for choosing Latin to generate synthetic data, is the fact that even though Glagolica is embedded in Unicode since version 4.1.0 (March 2005) [Aea07] the fonts available do not incorporate this standard. Hence, generating an a does not necessarily result in a Glagolitic a (ⰰ).

The training and test sets are generated by rendering TrueType fonts into images. This allows for generating test images with arbitrary fonts and at the same time to automatically annotate the ground truth data which minimizes the human effort. The system is trained using Times New Roman (regular) and Arial (regular). These fonts are chosen in order to guarantee that the system is trained on Serif fonts and Sans Serif fonts. In all subsequent experiments, 26 character classes (the English alphabet) are evaluated.

First, the system is tested with the training set so as to guarantee that the implementation is correct. If all 52 characters are considered, two characters are falsely classified, namely: i and j when generated with Arial. The i is confused with j while j with h. This can be traced back to the fact that Sans Serif characters such as i, j, l exclusively produce SIFT features that represent corners with changing orientations. However, all remaining characters (e.g. h) produce the same corners at stroke endings. That is why the SVM cannot be trained properly for Sans Serif fonts. Considering this experiment, one could think of joint probabilities being classified (see Section 5).

Figure 4.2 shows two results of the evaluation with synthetic data. Topologically complex characters (Figure 4.2 (left)) are easily recognized since they produce distinct local descriptors (note the probability interval is [0 1] in contrast to Figure 4.2 (right) [0 0.12]). On the opposite, the descriptors of i vote for j in Figure 4.2 (right). As can

be seen, all interest points are located at corners having different scales, which results in low prediction probabilities for all classes trained. The maximal probability, being 0.102, indicates that the decision made is uncertain.

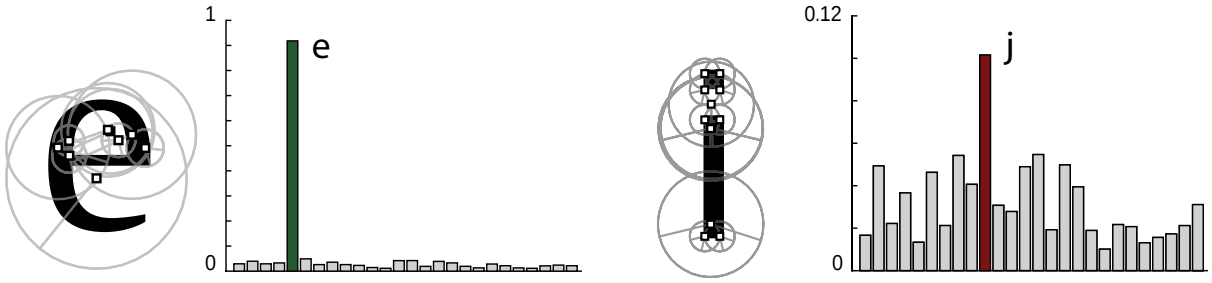

Figure 4.2: Two examples of the synthetic data set with their corresponding class probabilities. The i is classified falsely.

In addition to experiments on the training set, the system's performance is evaluated for new fonts presented. Therefore, a test set containing three Serif fonts (namely: Times New Roman, Georgia, Garamond) and three Sans Serif fonts (namely: Arial, Helvetica, Tahoma) is generated. This results in 156 sample characters, while the SVM is trained on 52 characters. In this experiment, a precision of 0.763 is achieved. If weak character clusters are rejected ($m_b = 0.85$), the precision increases to 0.865.

Experiments with Noise

Two further experiments are carried out on synthetic data in order to show the system's robustness with respect to certain degradations which are subsequently detailed. First, the system's robustness with regard to partially visible characters is regarded. Therefore, a gradient s_g is multiplied that occludes parts of the characters. Secondly, white Gaussian noise with zero mean and increasing standard deviation σ is added to the image.

Figure 4.3 shows the system's precision when varying the gradient's occlusion fraction, which is evaluated between 0.5 and 0.6. The upper sample shows an a occluded with a gradient set to 0.5. This corresponds to an occlusion fraction of two thirds, which means only one third of the character remains visible. Whereas the gradient is set to 0.6 for the lower sample image (the whole character is visible, however, it gradually fades out). The minimal precision, being 0.312, is achieved for $s_g = 0.5$. On opposite, the system achieves a precision of 0.923 when the gradient is set to 0.6. If one half of a character is occluded, the precision is 0.75. Thus, the system is capable for classifying partially visible characters as a consequence of the approach being based on local information.

The second experiment shows the system's behavior if white Gaussian noise is added. If the standard deviation σ of the Gaussian noise is set to 0.003, the precision is 0.923. Increasing the noise to $\sigma = 0.008$ decreases the system's precision to 0.904. Hence, the proposed system is robust with respect to Gaussian noise.

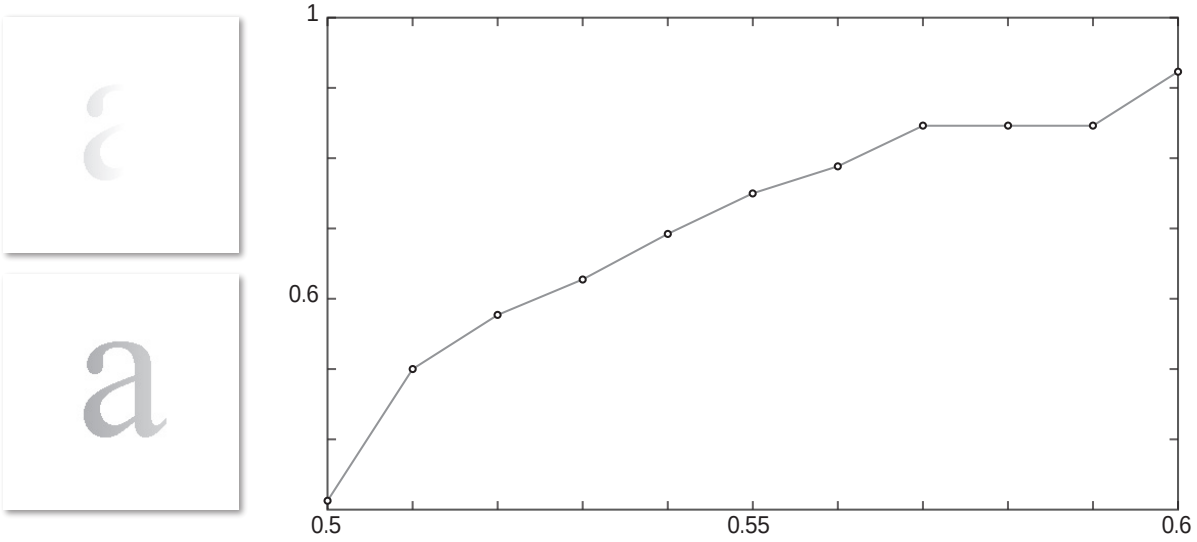

Figure 4.3: Synthetically degraded character if $s_g = 0.5$ (upper) and $s_g = 0.6$ (lower). The right plot shows the system's precision when varying s_g.

4.2 Character Evaluation

By extracting single characters, it is possible to solely evaluate the classification step illustrated in Figure 3.1. Therefore, two datasets are constructed that consist of single characters which are extracted from the *Cod. Sin. Slav.* 5Nand annotated.

The first dataset (SETA) consists of 10 classes having 10 – 12 samples each (totally 107) which are well preserved. This dataset is a reference for the evaluation with degraded characters. The second dataset, which is referred to as SETB, contains 25 character classes with about 9 characters per class (totally 198). Degraded or partially visible characters were extracted to construct this set. It is used to demonstrate the systems' behavior when degraded characters need to be recognized.

Figure 4.4 shows examples of both datasets. It can be seen that some characters such as ѻѣ$_b$, ѿ and ѱ$_b$ are similar to each other. The degraded characters in the second row differ strongly from those of SETA. They are hard to read for humans.

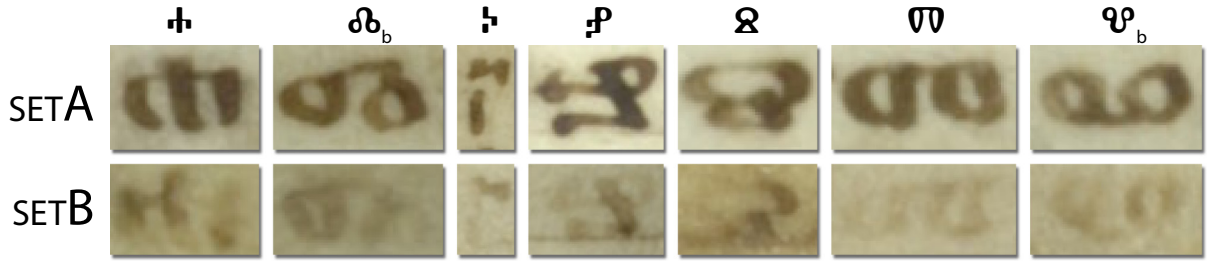

Figure 4.4: Examples of the datasets evaluated. The first row shows examples of SETA, whereas the second row shows the same characters from the degraded dataset.

4.2.1 Evaluation of Dataset A

The SETA is first evaluated in order to show the method's performance on noise-free data. As mentioned before, 10 SVM kernels are trained using 10 samples per class. Then all 107 test characters are evaluated. The voting is the same as described in Section 3.5 except for the fact, that clustering does not need to be performed. Another difference to the system described in Chapter 3 is the interest points' threshold. It is set to 0.009 instead of 0.01 in order to guarantee that highly degraded characters never remain without any descriptor being detected.

For the character classification, an overall precision of 98.13 % is achieved. Thus, solely two characters[1] out of 107 are falsely predicted. Both confused characters consist of two circles and a connecting stroke (see Figure 4.4, second and last column) which produce similar descriptors.

A confusion matrix of the local descriptors is given in Table 4.1 in order to show the class confusion. To construct this table, the highest probability of each descriptor being classified was taken into account. Totally, 1714 descriptors where detected in SETA while solely 60 % of them could be classified. In Table 4.1, the columns indicate the system's prediction of a local descriptor while the rows show its correct class. Hence, values in the principal diagonal (bold font) represent the precision of the particular class. The other values (e.g. 2.9 in the last column of the first row) indicate that 2.9 % of descriptors that belong to the class ⰰ are falsely predicted as belonging to ⱏ$_b$. The last column illustrates the total number of descriptors that belong to the particular class. In contrast, the last row gives the number of descriptors that where classified as the respective class.

The overall precision of the local descriptors is 79.83 %. Compared to the overall precision being 98.13 % it can be concluded, that the voting improves the character classification. This can be attributed to the fact, that false classifications are assumed to be noise with a given prior. Hence, if 10 descriptors of a character (e.g. ⰰ) vote for ⰰ the other 10 descriptors will not necessarily vote for one other class but for different classes.

prediction (columns); correct class (rows)

%	ⰰ	ⰱ	ⰲ$_a$	ⰲ$_b$	ⰵ	ⰺ	ⱂ	ⱄ	ⱆ	ⱏ$_b$	#
ⰰ	**74.3**	1.4	7.1	8.6	2.9	0.0	1.4	1.4	0.0	2.9	70
ⰱ	0.0	**92.2**	2.6	2.6	0.0	0.9	0.0	0.0	1.7	0.0	116
ⰲ$_a$	0.5	2.0	**85.9**	2.0	0.5	1.0	1.0	2.4	1.5	3.4	205
ⰲ$_b$	0.8	3.4	6.7	**65.5**	4.2	0.0	2.5	1.7	3.4	11.8	119
ⰵ	5.4	0.0	2.7	0.0	**81.1**	0.0	5.4	1.4	0.0	4.1	74
ⰺ	0.0	8.2	1.6	0.0	3.3	**70.5**	6.6	1.6	4.9	3.3	61
ⱂ	1.0	3.0	2.0	0.0	3.0	2.0	**87.9**	0.0	0.0	1.0	99
ⱄ	2.5	0.0	0.0	1.3	7.5	1.3	3.8	**81.3**	0.0	2.5	80
ⱆ	1.0	2.0	6.9	7.9	0.0	0.0	1.0	1.0	**74.3**	5.9	101
ⱏ$_b$	0.0	4.7	4.7	6.6	0.0	0.0	0.0	3.8	4.7	**75.5**	106
#	62	131	209	107	79	49	105	80	92	117	1031

Table 4.1: Confusion matrix of the local descriptors in SETA.

[1] A ⱏ$_b$ is mistaken with a ⰲ$_a$ and a ⰲ$_b$ is mistaken with a ⱏ$_b$.

As can be seen in Table 4.1, Ⰲ$_a$ produces nearly twice as much descriptors than the other characters do. This can be attributed to the fact that the character is larger than the other ones ($\approx 100 \times 80px$ compared to $\approx 60 \times 50px$). This ratio also applies to the training. But if the column Ⰲ$_a$ is examined it can be assumed that not significantly more descriptors are confused with this class than with other classes. Thus, the classifier is capable of handling classes having more training samples.

The worst classification result is obtained for Ⰲ$_b$ whose precision is 65.5 %. When regarding the last column of that row, it can be seen that Ⰲ$_b$ is most likely (11.8 %) confused with Ⰳ$_b$ which has the same shape rotated by 180 °. This class confusion also holds for Ⰳ$_b$ which is most likely classified as Ⰲ$_b$ with 6.6 %. However, this correlation does not hold in general. For example the character Ⰰ is confused with Ⰲ$_b$ in most cases (8.6 %), while solely 0.8 % of the Ⰲ$_b$ descriptors are falsely assumed to be those of Ⰰ.

In general, it can be observed that the class confusions are intuitive. In other words, the probability of a class being mistaken with another one is high when human observers consider these characters as similar.

4.2.2 Evaluation of Dataset B

For a direct comparison of both datasets, the same ten classes are chosen of setB. Certainly, the same classifier is used for both test setups. In contrast to setA, the degraded characters in the second dataset have a lower precision, which is 78.89 %. Additionally, the ratio between descriptors detected and those classified is lower which is in this case 39 % compared to 60 % in setA. These numbers indicate that it is harder for the system to classify degraded characters. On the other hand, the system copes with uncertainty which arises from the fact that fewer descriptors are classified in this case.

prediction

correct class	%	Ⰰ	Ⱎ	Ⰲ$_a$	Ⰲ$_b$	Ⰵ	Ⰹ	Ⱂ	Ⰿ	Ⱆ	Ⰳ$_b$	#
	Ⰰ	**56.1**	2.4	12.2	7.3	7.3	4.9	7.3	0.0	0.0	2.4	41
	Ⱎ	1.5	**69.2**	3.1	0.0	1.5	3.1	12.3	1.5	0.0	7.7	65
	Ⰲ$_a$	1.1	10.3	**67.8**	5.7	4.6	2.3	3.4	1.1	0.0	3.4	87
	Ⰲ$_b$	8.1	8.1	12.9	**33.9**	4.8	3.2	14.5	4.8	1.6	8.1	62
	Ⰵ	2.6	5.3	7.9	0.0	**60.5**	10.5	0.0	5.3	0.0	7.9	38
	Ⰹ	3.1	0.0	6.3	6.3	9.4	**56.3**	15.6	0.0	0.0	3.1	32
	Ⱂ	2.5	5.0	2.5	7.5	0.0	5.0	**67.5**	5.0	0.0	5.0	40
	Ⰿ	4.7	14.0	7.0	7.0	2.3	2.3	11.6	**44.2**	0.0	7.0	43
	Ⱆ	1.8	12.3	17.5	5.3	0.0	1.8	3.5	3.5	**29.8**	24.6	57
	Ⰳ$_b$	1.9	5.6	13.0	13.0	1.9	5.6	13.0	9.3	5.6	**31.5**	54
	#	37	80	100	47	39	37	69	35	21	54	519

Table 4.2: Confusion matrix of the local descriptors in setB.

Table 4.2 shows the confusion matrix of the local descriptors in setB. In general, the confused classes are similar to those in setA even though the overall precision is decreased. The Ⱆ stands out in this test, since hardly any descriptor (4/519) is falsely classified as Ⱆ. Although not as outstanding, this peculiarity can be observed in Table 4.1 too. One

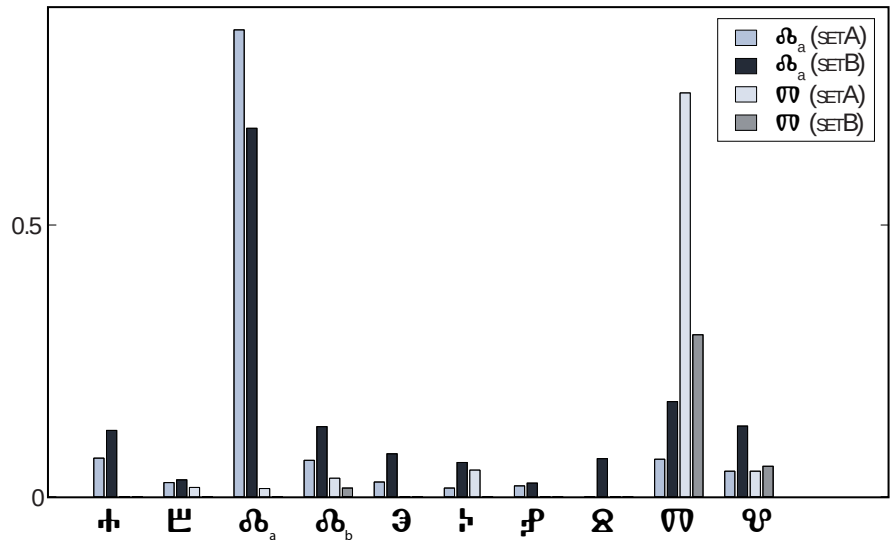

Figure 4.5: Prediction rates of all classes when two characters are evaluated from both datasets. ⰴa has most false predictions in SETA while ⰾⰾ has the least number of false predictions in SETB.

intuitive explanation to this observation is the fact that four out of ten characters are similar to the ⰾⰾ in these test setups. Thus, similar descriptors scale down the feature space where the SVM classifies a feature into this class. The observed phenomenon indicates one of the major disadvantages of the system proposed which is further discussed in Chapter 5.

Figure 4.6 compares the per-class precision of both datasets. As can be seen, the results of the degraded dataset are highly correlated[2] with those of SETA. This draws the conclusion that the interclass variations do not significantly change if characters are degraded. The performance decrease, when degraded characters are regarded, is on average $27.16\,\% \pm 11.24\,\%$.

Evaluation of All classes

In addition to the comparison of SETA and SETB, all 198 degraded characters where evaluated. Even though, 25 different classes are predicted in this evaluation (+15 classes), the precision decreases slightly by 7.17 %. Thus, the overall precision is 71.72 % when descriptor voting is applied on degraded characters. The ratio of detected descriptors and those classified is now 26 % which is decreased by 13 % compared to the previous test on the same dataset with 10 classes. Since the performance decrease is lower than the complexity increase, the system proofs to be capable for classifying degraded manuscripts. Table 4.3 gives an overview of all tests performed with single Glagolitic characters.

[2]correlation coefficient: 0.719

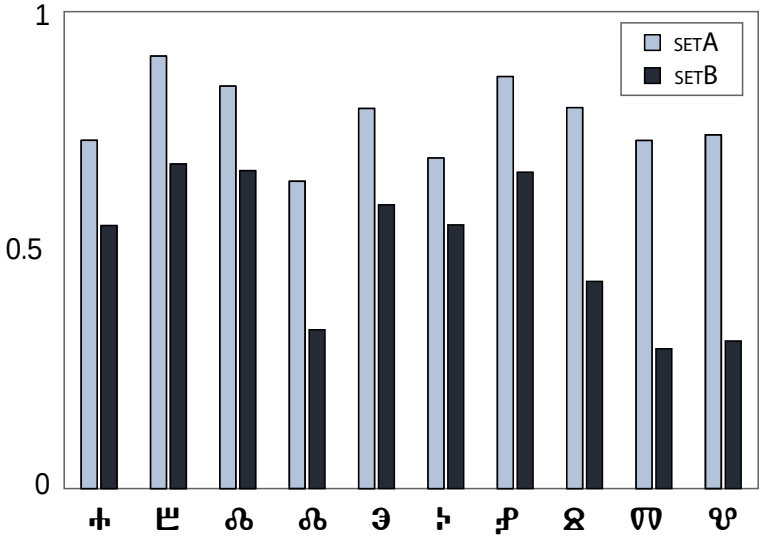

Figure 4.6: Comparison of the class precision between setA and setB. The precision is computed on the descriptor level (1550 descriptors where evaluated).

	#	# classes	precision
setA	107	10	0.981
setB	90	10	0.789
setB	198	25	0.717

Table 4.3: Dataset, number of samples, number of classes and the system's precision

4.3 System Evaluation

In this section, the evaluation of the system proposed is given. Beside the system's performance on the dataset, crucial parameters are studied. In order to evaluate the system, 15 different pages containing 1055 characters are extracted from the *Cod. Sin. Slav.* 5n. The pages were chosen randomly. It can be seen in Figure 4.7 that the pages contain faded-out ink, degraded characters and background noise. The groundtruth was annotated manually. Therefore, each character was brushed with a gray-value that corresponds to its class index. These indices correspond to the alphabetical order of the Glagolica and are given in Table 1. Figure 4.7 additionally shows that the annotation does not need to perfectly fit the subjacent character, since the system provides one coordinate per character. Thus, if the center of mass obtained by the clustering is located within the annotated blob, it is assumed to belong there. Furthermore, local descriptors are evaluated with this annotated test set. Since interest points which describe a part of a character may lie outside the character's border, one is well advised to tag more. Additionally to the groundtruth, characters where annotated according to their condition. In Figure 4.7, the gray border illustrates the good versus degraded annotation. All characters outside the border are annotated as being degraded. Certainly, this annotation highly depends on the operator. However, it is exclusively used to determine the performance difference between good and degraded characters which is compared to the results presented in Section 4.2.

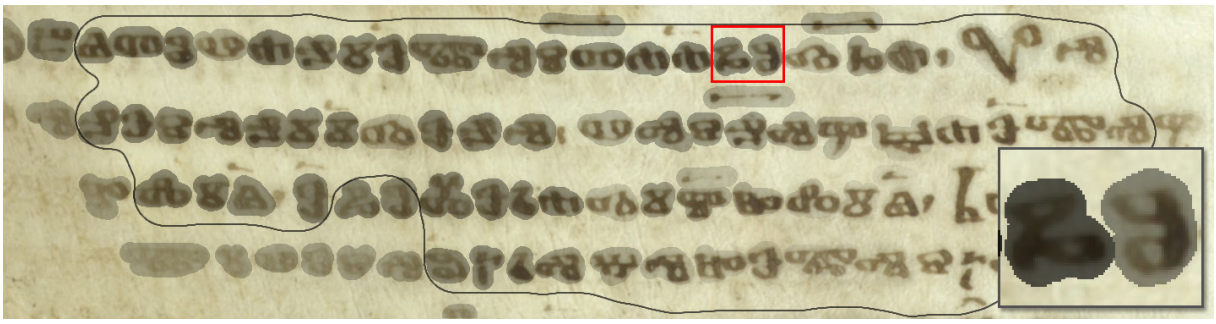

Figure 4.7: Manually tagged groundtruth extracted from page 38 verso of *Cod. Sin. Slav.* 5N.

Statistical Methods

All statistical methods used for evaluating the system are based upon three different values:

- **True Positive**: values that correspond with the groundtruth
- **False Positive**: correctly located values with false class labels
- **False Negative**: groundtruth values that are not detected by the system

Thus, a character is exclusively considered as a True Positive if all centers of mass that are within the tagged blob have the same class index as the blob. If at least on center of mass does not correspond to the tagged label, the character is considered as False Positive. On the opposite, characters that were not detected at all (e.g. if the ink is faded-out) are defined as False Negatives.

These values allow for computing the precision and recall. The former is defined as the sum of True Positives divided by the sum of retrieved values (True Positives + False Positives). The latter is the sum of True Positives divided by the total number of elements that exist (True Positives + False Positives + False Negatives). Thus, the precision indicates the percentage of correctly classified characters to those retrieved. Whereas the recall specifies the percentage of correctly classified characters to those present in an image.

The aim of a classification task is to maximize both, the precision and the recall. Therefore the F score is introduced, which is a weighted average between the precision and the recall:

$$F_\beta = \frac{(1+\beta^2)\mathsf{p}\cdot\mathsf{r}}{\beta^2\mathsf{p}+\mathsf{r}} \quad \Longleftrightarrow \quad F_\beta = \frac{(1+\beta^2)\mathsf{tp}}{(1+\beta^2)\mathsf{tp}+\beta^2\mathsf{fn}+\mathsf{fp}} \tag{4.1}$$

where r is the recall and p is the precision. The right equation expresses the F score in terms of True Positives/False Positives. There, tp stands for True Positives, fp are False Positives and fn is defined as False Negatives. The β allows weighting the precision or the recall. Thus, if β is set to 0.5, the precision is weighted twice as much as the recall. This value is user defined and depends on the particular classification task.

In our case, $\beta = 0.5$ since it is more important that correct results are retrieved than to detect all degraded characters. If a character is missed, the operator has the possibility to select this character. After that, the classification is performed on this individual character as it is done on SETB.

4.3.1 Parameter Evaluation

In this section, the system's parameters are evaluated. First, the parameters of the local descriptors are given, then, those of clustering and voting are presented. The classifier has two parameters that need to be adapted to a given training set. These parameters are found by means of a cross-validation, which is explained in Section 3.3.3.

Local Descriptor Parameters

Three parameters (*thresh*, r, o_{min}) are crucial for the computation of local descriptors. The threshold *thresh* rejects weak local maxima (see Section 3.1.1). Similarly, r detects interest points that have a poor localization as they are placed on edges. In contrast, o_{min} defines the minimum scale level.

In Figure 4.8, the evaluation of *thresh* is given. The grid is chosen to be logarithmic around 0.01. This is done to give a better insight on the system's performance around the maximum. The central line shows the $F_{0.5}$ score when varying *thresh*. In addition, the precision (upper line) and the recall (lower line) are given. The maximal F score (0.75) is obtained when *thresh* is set to 0.01 which is a low threshold compared to Lowe [Low04], who proposes 0.03. This can be attributed to the fact that Lowe matches local descriptors but they are classified in the proposed system. Thus, he needed reliably located descriptors. In contrast, the system introduced in this thesis benefits from more features as their number improves the voting. However, if the threshold is set too low (0.0025), noise adulterates the classification, which results in a lower F score, namely 0.71.

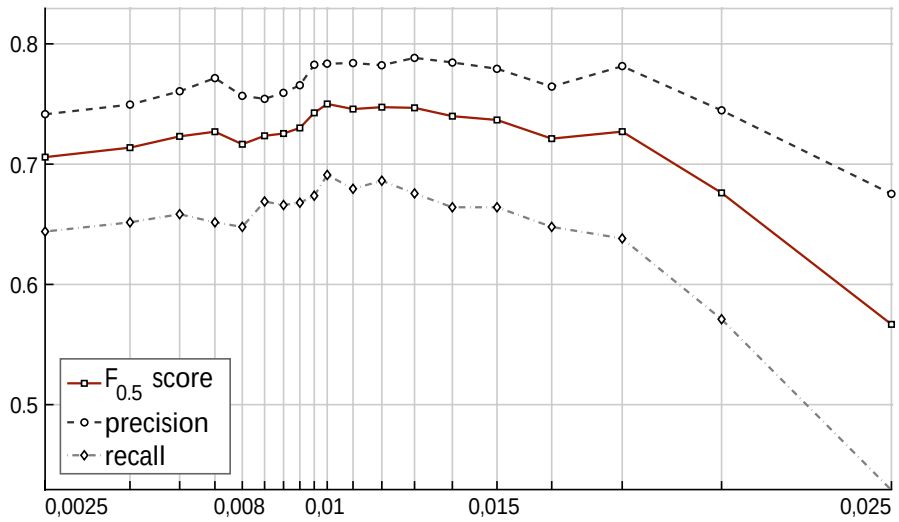

Figure 4.8: Evaluation of the local descriptors' threshold. A logarithmic grid around 0.01 is used to show the performance.

The curvature threshold r is evaluated between 10 and 90. It turns out, that varying the edge threshold parameter has hardly any influence on the system's performance ($\sigma = \pm 0.0028$). However, the maximal F score is achieved when $r = 35$ (Lowe proposes to set $r = 10$). It improves the performance by 0.01. Thus, the edge threshold can be neglected in further studies.

Lowe proposes to subsample the input image in order to detect interest points having the pixel's frequency. However, it turned out that subsampling decreases the performance. When o_{min} is set to -1 (scale-space with subsampling), the F score is 0.73. But if the image is not subsampled ($o_{min} = 0$), the performance increases to 0.76. If o_{min} is set to 1 (the second octave), the performance decreases dramatically to 0.22. This is, because parts of characters are not described by local descriptors if the first octave is not computed. Subsampling the image results in 28608 descriptors on the previously mentioned dataset which are reduced by 5267 when $o_{min} = 0$. That is why the images are not subsampled in the system proposed, which additionally improves the computational speed (fewer interest points) and the memory consumption (no subsampling).

Clustering Parameters

For the character center estimation (see Section 3.4.1), two thresholds s_t, d_t exist. The former defines the minimum scale of an interest point so that it is considered as describing a character. The latter specifies the minimum distance of two interest points to interpolate them.

The scale threshold s_t is evaluated in the range of 0.3 - 1. Again, a logarithmic grid around 0.6 is used in order to give a more detailed evaluation. If s_t is set to 0.3, fewer interest points are selected for the clustering initialization. On the opposite, $s_t = 1$ selects the maximum of the interest points' scale distribution. In Figure 4.9, the $F_{0.5}$ score, precision and recall are illustrated. It can be seen that too many initial character centers $s_t = 1$, which result in too many clusters, gain a low performance (0.56). This is, because parts of characters that are similar to parts of different characters are clustered having few interest points. On the other hand, the recall decreases if too few initial character centers are chosen ($s_t = 0.3 \Leftrightarrow \mathsf{recall} = 0.61$). This can be traced back to the fact that characters are missed if too few initial character centers are obtained for the k-means. The maximal performance, being 0.76, is achieved for s_t set to 0.6.

Apart from the scale s_t, the minimum distance threshold d_t is regarded. This threshold adapts to the particular image too. It is defined as the percentage of the minimum scale level chosen for character estimation. Thus, if $d_t = 1$, solely interest points, which are closer to each other than the minimal scale, are interpolated. The theoretical background is that if the areas of two interest points overlap for at least 39.1 %, they represent the same character[3]. The minimum distance threshold was evaluated between 0.3 and 2. The test results support the theoretical background since the maximum $F_{0.5}$ score is achieved for $d_t = 1$. This indicates that the minimum scale corresponds with the minimum distance $s_{min} = d_{min}$.

[3]In case of two interest points having the same radius ($r_1 = r_2$) and their centers lie on the circular path.

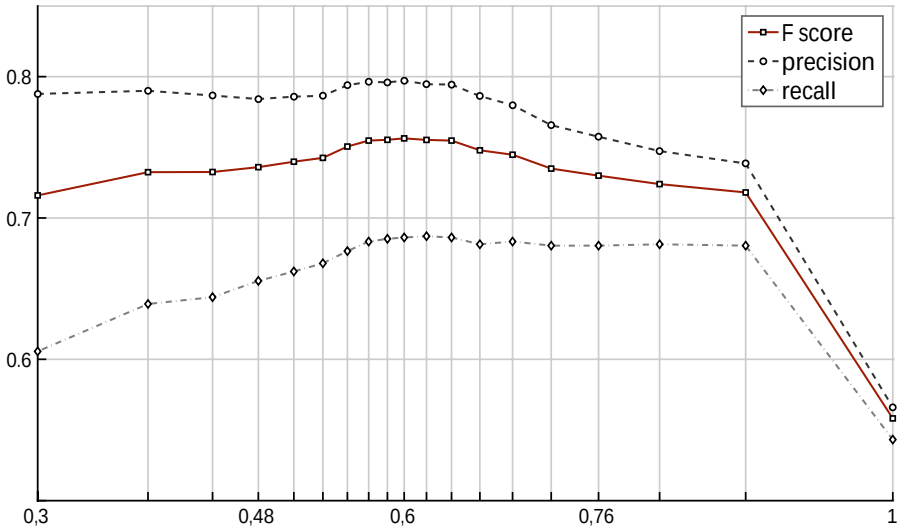

Figure 4.9: Evaluation of the minimum scale threshold s_t. The $F_{0.5}$ score, precision and recall are illustrated.

Feature Voting Parameters

For the voting scheme described in Section 3.5, three parameters ω_s, ω_d, m_b need to be considered. The first two are the scale weighting ω_s and the distance weighting ω_d, which weight local descriptors according to these properties. Both do not have tunable parameters, but can be turned on or off. Thus, the subsequent experiment evaluates their influence on the classification process. Finally, the last parameter m_b removes weak character clusters dependent on the class probability distribution.

If the scales are weighted $\omega_s = 1$, high weights are assigned to local descriptors which represent parts of characters. This method improves the classification performance from 0.716 by 0.039. The improvement can be attributed to the fact that a lower weight is assigned to descriptors which incorporate more than one character. Similarly, the distance weight ω_d, which favors descriptors being closer to the cluster center, improves the performance. In this case, the $F_{0.5}$ score is increased by 0.036. The distance weight improves the system because descriptors which are further away from the center have a higher probability of being background clutter or belonging to other characters.

In order to further improve the system's precision, weak character clusters are rejected. Therefore, a parameter m_b is introduced that controls the behavior of the cluster rejection. Figure 4.10 shows the system's performance when varying m_b. If it is low (e.g. 0.5), clusters are easily rejected, which results in a high precision but a low recall (correct clusters are rejected too). On the other hand, if clusters are not rejected at all ($m_b = 1$), the precision is decreased while the recall reaches its maximum. For this experiment, the F_1 score is considered, since it is not intended to reject correct clusters. The maximal performance is gained when $m_b = 0.875$ which is a good trade-off between precision and recall.

Summing up the parameter evaluation, it can be concluded that solely two parameters are crucial for the system, namely the descriptors' threshold *thresh* and the minimum scale threshold s_t. The first is crucial because it controls the amount of information extracted

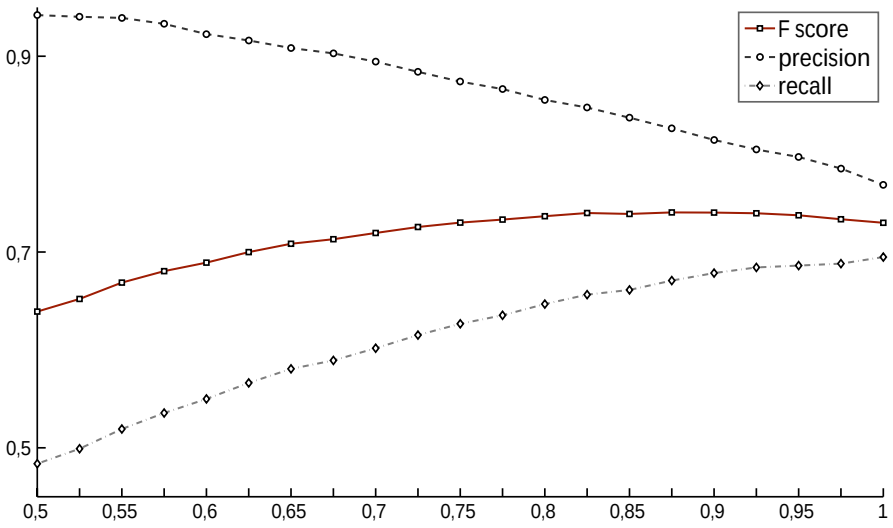

Figure 4.10: Removing weak clusters increases the precision (black dashed line) while the recall (gray dashed line) is decreased.

from a given image. Consequently, this changes the system's capability of handling uncertainness and the character localization process. The minimum scale threshold s_t directly influences the number of characters localized in a given image. These two parameters should be adapted if a different dataset is observed. All other parameters have little influence on the system's performance or they are not dependent to a particular dataset or script (e.g. r, d_t, ω_s).

4.3.2 Evaluation of the Investigated Dataset

The results of the system evaluation are presented in this section. Basically four tests are carried out on the whole annotated test set. First, an artificial clustering approach is implemented in order to evaluate the system's major steps (classification/localization) separately. In order to show the effect of degraded characters on the system's performance, the testpanels are additionally annotated according to this criterion. The performance of each individual testpanel and character class is extracted so that conclusions of the system's disadvantages can be drawn.

Clustering Evaluation

In order to demonstrate the effect of the character localization, an artificial clustering is implemented. This is based on the annotated groundtruth where cluster centers are defined as the center-of-mass of each blob. As constraint, solely interest points being within a character blob are considered. The evaluation with artificial clustering allows separately regarding the localization and classification step on the same dataset. Thus, the error introduced by clustering can be extracted.

Using optimized parameters as discussed in Section 4.3.1 results in an $F_{0.5}$-score of 0.772 (see Table 4.4 and Figure 4.11). If the artificial clustering is applied, a $F_{0.5}$-score of 0.805 is achieved. This directly draws the conclusion that the F-score is decreased by

0.033 because of the character localization. The test setup additionally shows that the character clustering has hardly any influence on the system's precision (difference: 0.005). In contrast, the proposed k-means decreases the recall rate by 0.075. This results from clustering errors which increase the False Negatives rate as characters are not localized correctly.

	#	**recall**	**precision**	$F_{0.5}$-score
with clustering	1055	0.673	0.832	0.772
no clustering	1055	0.748	0.837	0.804

Table 4.4: Number of characters, system's recall, precision and F-score when the system proposed and groundtruth clustering is applied.

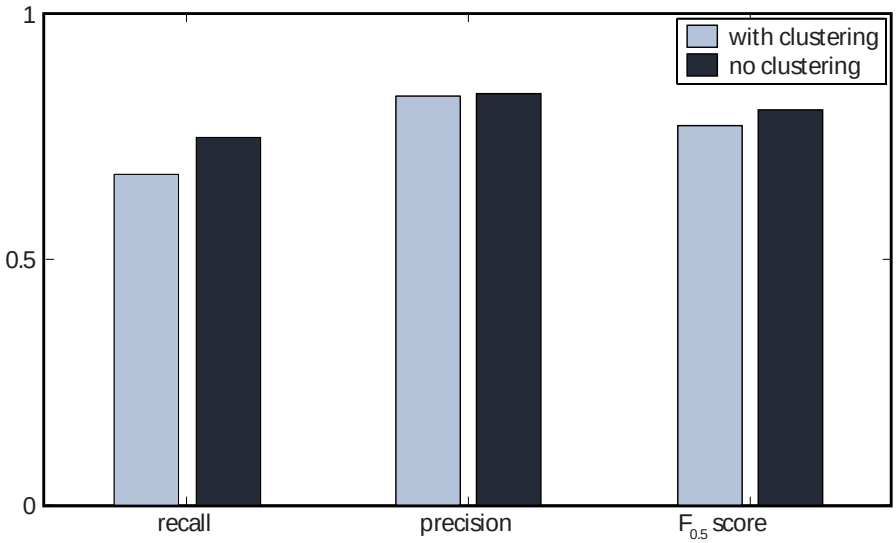

Figure 4.11: System's recall, precision and F-score when the proposed system and groundtruth clustering is applied. Note that the precision is the same (difference: 0.005).

Character Quality Evaluation

The dataset used for the discussed evaluation comprises normal and degraded characters. This is done to guarantee a statistically representative dataset of the investigated manuscripts. In the subsequent discussion, results are presented that show the system's performance on good and degraded characters, which were manually annotated beforehand. It is intended to show the system's behavior when solely good characters are considered and to draw conclusions about the character localization when degraded characters are considered.

Table 4.5 and Figure 4.12 show the system's recall, precision and F-score on the investigated dataset. The investigated dataset contains 142 degraded characters which are 13.5 % of all characters evaluated. If normal characters are regarded, a $F_{0.5}$-score of 0.79 is achieved. In contrast, degraded characters have a lower performance (namely: 0.38). This arises mainly from the fact that the recall is low due to 64 False Negatives which

draws the conclusion that 45.1 % of degraded characters are missed. When comparing these numbers to previous tests discussed in Section 4.2, where degraded characters were extracted, a performance loss can be observed. On the one hand, it can be attributed to the fact that no recall was obtained in this test since False Negatives do not exist if characters are extracted. On the other hand, the interest point's threshold was chosen to be lower (0.009) which results in more interest points that improve the precision.

	#	**recall**	**precision**	$F_{0.5}$-score
normal	913	0.732	0.862	0.792
degraded	142	0.296	0.539	0.382
setB	198	-	0.712	0.712

Table 4.5: System's recall, precision and F-score when normal and degraded characters are considered. The last row shows the character evaluation from Section 4.2 with degraded characters.

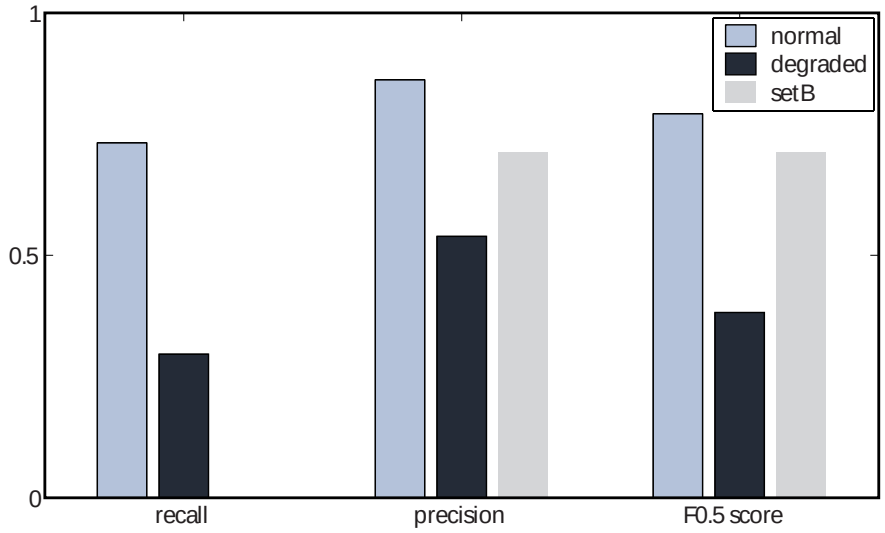

Figure 4.12: System's recall, precision and F-score when normal and degraded characters are considered.

Test Panel Evaluation

In the experiments discussed previously, all test panels where considered at the same time in order to give statistically significant results. However, the test panels' quality differs according to the manuscript folios they were extracted from. In order to show these differences, the precision, recall and F-score of each test panel are regarded.

Table 4.6 shows the system's performance on the individual test panels. The mean F-score averaged over the test panels is 0.75. However, it can be seen in Table 4.6 that two test panels are outliers, namely: test panel #1 and test panel #10. Both test panels are illustrated in Figure 4.13. As can be seen, test panel #, 1 which has a F-score of 0.9, solely contains two faded-out characters. That is why the system's performance is better

#	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
***F*-score**	0.9	0.6	0.7	0.8	0.6	0.8	0.8	0.8	0.8	0.5	0.6	0.8	0.8	0.9	0.8
recall	0.9	0.6	0.6	0.8	0.5	0.7	0.8	0.7	0.6	0.3	0.5	0.7	0.7	0.7	0.7
precision	1.0	0.7	0.8	0.9	0.7	0.9	0.9	0.9	0.8	0.7	0.6	0.9	0.9	0.9	0.8

Table 4.6: The system's F-score, recall and precision on the respective test panels.

on this panel compared to the other test panels. On the other side, test panel # 10 was extracted from a so-called palimpsest, which means that characters were partially erased and a new script was written over the original text. This results in degraded characters. More precisely, the clustering fails on this test panel since the stains of the second script produce false interest points which results in false clusters. That is why the recall being 0.3 is lower compared to the other test panels.

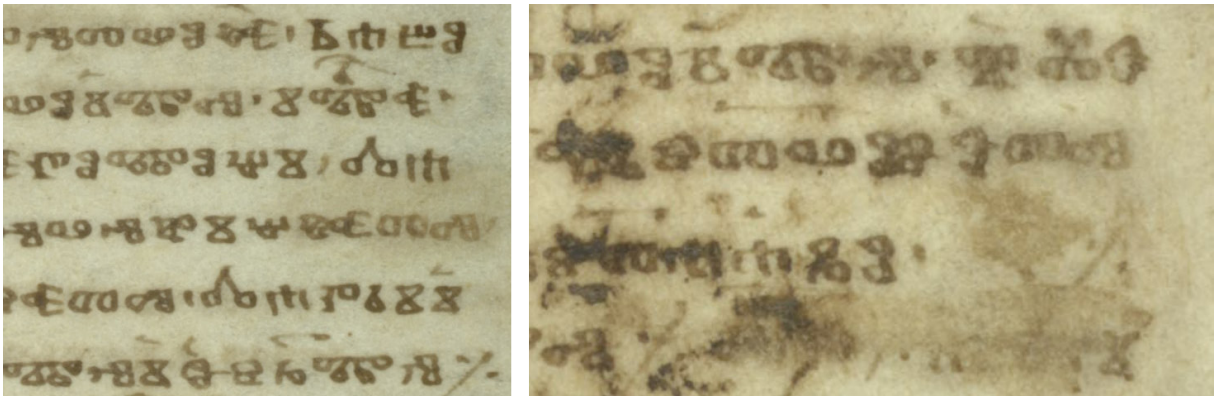

Figure 4.13: The two outliers, test panels # 1 (left) and test panel # 10 (right).

Character Class Evaluation

In order to show the classification performance of each character class separately, the class statistics over all test panels are extracted. Figure 4.14 shows the $F_{0.5}$-score of each character class. Since the characters have different a-priori probabilities, a different number of characters are observed per character class. The width of each bar in Figure 4.14 indicates the normalized number of characters. This allows comparing the F-score of a given character class with its a-priori probability in the observed test set.

Figure 4.14 shows that 𝔬 has most instances (namely: 114) in the given test set. In contrast, 𝔐 and 𝔒$_{\mathrm{a}}$ are contained only once in the whole test set. Hence, their performance cannot be regarded as statistically relevant. The lowest performance being 0.355 is gained by 𝔟, which is in most cases confused with 𝔬 (27.3 %) and 𝔥 (18.2 %). This can be traced back to the fact that 𝔟 solely consist of a circle and one vertical stroke, which are mistaken with the circles of 𝔬 and 𝔥. The highest performance of statistically relevant character classes ($n > 50$) is achieved by 𝔚 having a $F_{0.5}$-score of 0.911. This can be traced back to its complex and individual shape (4 connected circles).

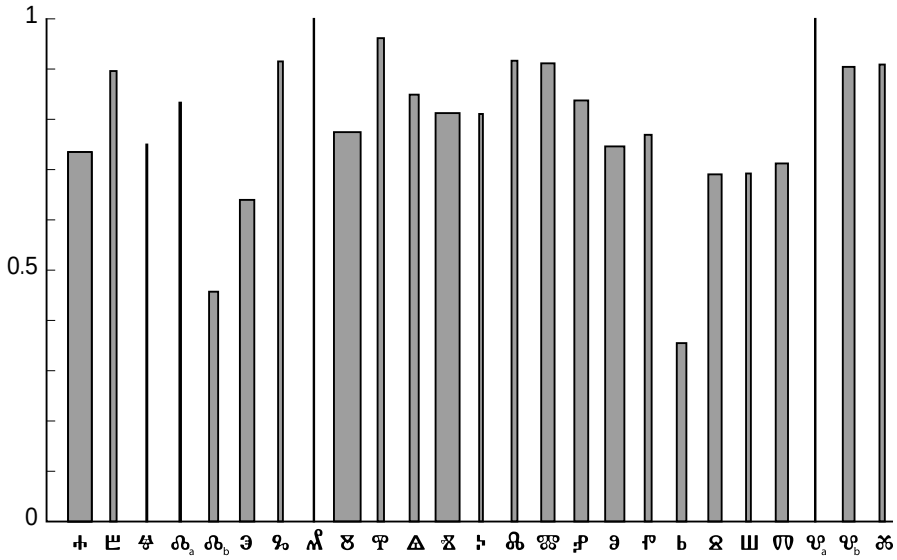

Figure 4.14: Weighted $F_{0.5}$-score of the character classes. The width of each bar indicates the percentage of characters that belong to the class.

Summary

In this chapter, the system's performance was discussed. The experiments on synthetic data were carried out in order to show the system's behavior if undistorted data is considered. But it was additionally shown that the method proposed can easily be adapted to other writing systems. This experiment proofed the implementation's correctness. Adding noise to synthetic data allowed for evaluating the system's robustness with respect to noise.

The second experiment aimed at analyzing system's behavior when degraded characters need to be recognized. Therefore, degraded characters were compared with normal characters extracted from the *Cod. Sin. Slav.* 5N. Beside this comparison, the performance trend was analyzed if the number of classes is increased.

The final evaluation on annotated ground truth data allowed for conclusions on the system's behavior in real world applications. There, the system's major steps were again computed separately in order to derive a detailed performance evaluation.

Chapter 5

Conclusion

This thesis presents a new methodology for character recognition of ancient manuscripts. The approach, which is inspired by recent object recognition systems, exploits local descriptors directly extracted from grayscale images. Multiple SVMs with RBF kernels are used to classify the local descriptors. The character localization is based on clustering interest points previously extracted for the computation of local descriptors. A scale selection that adapts to the observed manuscript image allows for the cluster center initialization.

The system proposed was evaluated on synthetically generated data as well as real world data extracted from the *Cod. Sin. Slav.* 5N. Experiments showed the system's capability to be trained on Latin font as well as the Glagolica, even though both writing systems have little in common. Experiments on synthetic data demonstrated the system's behavior when noise, such as white Gaussian noise, or partially visible characters are present. In addition, a dataset was created that consists of highly degraded Glagolitic characters. Experiments on this dataset proofed the system's capability to recognize degraded characters and the difference to well preserved characters. Additional tests with annotated ground truth allowed for extracting errors introduced by clustering and those of the classification.

The presented character recognition system does not need any pre-processing of document images. In contrast to existing systems, a new architecture was designed that focuses on degraded manuscript images. Since ancient manuscripts – in contrast to modern ones – exhibit stains, faded-out ink and rippled pages, new challenges are faced when trying to recognize characters of ancient documents. The degradations can be attributed to bad storage conditions, on-purpose destruction and the ravages of time.

Although the data dramatically changes between modern and ancient manuscripts, the methodology proposed does not change except for minor optimizations. As a consequence to the previously mentioned degradations, a binarization is not applicable for ancient manuscripts. This fact is stated by other authors and was further discussed in Section 2.1. A simple example why binarization fails when ancient documents are regarded is subsequently given. If an image is binarized, every pixel gets assigned one out of two class labels: foreground, background. But considering ancient manuscripts, gray-values of characters are the same as those of stains. When regarding faded-out ink, degraded characters have the same gray- value as background in a different region. That is why

a binarization – local or global – cannot separate foreground from background correctly. Hence, methods are proposed that incorporate context knowledge in order to improve the binarization. But nevertheless, features that are extracted from binary images suffer from misclassifications that occur within the binarization step. Thus, false predictions within the binarization propagate through all subsequent processing steps.

As a consequence of this reasoning, a new character recognition architecture was developed. It is designed similarly to existing object recognition systems. In contrast to character recognition systems, object recognition is not based on binarization since decades. Thus, an object recognition system allows for recognizing characters even if the ink is faded-out or background clutter degrades characters.

Disadvantages of the Proposed System

It was shown in Chapter 4 that the proposed system has disadvantages when certain aspects are considered which will be discussed subsequently. If modern fonts such as Latin need to be recognized, characters with little topological structure exist such as i, j, l. Considering these characters and assuming they do not have Serifs, local structure information is not capable for recognition. This can be attributed to the fact that solely corners with changing orientations are passed to the classifier. Since the only difference between an i and a j is the descender which is not recognized by local descriptors, a correct classification of these characters cannot be guaranteed. Handwritten characters, in contrast to printed fonts, have the advantage that the topological structure – even for similar characters – is changed according to the sequence of strokes written.

In addition to this, the character localization, which is currently based on the interest points extracted, is still weak if characters are at the image border, or highly degraded characters are considered. It was shown in Section 4.3 that recognizing degraded characters performs better if the clustering does not need to be performed.

In contrast to state-of-the-art OCR engines, the system proposed does not exploit dictionaries to improve the recognition rate. This is, on the one hand, because up to now, there does not exist a Glagolitic dictionary that would be applicable for OCR. On the other hand, the thesis concentrates on Computer Vision, not Information Retrieval.

Advantages of the Proposed System

It was stated in Section 4.3 that the system proposed achieves an overall $F_{0.5}$ score of 0.772 on degraded manuscript images when 25 character classes are trained. The precision is even higher, being 0.832. These experiments were performed on randomly selected manuscript pages that contain faded-out characters, background clutter and locally skewed text lines. In contrast to state-of-the-art OCR systems, no prior knowledge about the page layout, the page scale or orientation needs to be incorporated for the method introduced. Thus, the recognition rates mentioned are achieved without preprocessing. This allows for a flexible recognition system that can easily be adapted to other datasets, writers and writing systems. Considering the dataset investigated and the system's performance, it can be stated that it is capable for recognizing degraded manuscript pages which it was designed for.

It was shown in Section 4.1 that the system is suitable for to correctly classifying partially preserved characters, which is important when degraded manuscripts are con-

sidered. This can be attributed to system's design which directly classifies local structure information. Thus, the global topology of characters does not need to be considered in order to correctly predict the character class. An additional advantage of classifying local information is its robustness with respect to intra-class variations arising from different writers and writing materials.

The system proposed does not solely predict character classes but assigns class probabilities to each character recognized. This restrict alternatives for characters that are not recognizable by human experts anymore. Thus, a faster transcription is achieved if philologists apply the system introduced. As an example, characters having faded-out ink need manual (local) contrast enhancement so as to allow for a human recognition. These characters are easily recognized by the system since the first derivation is exploited, which renders the system invariant to linear illumination (contrast) changes.

Future Work

Since this thesis covers rather a case study on a new architecture for character recognition systems than a complete OCR, the methodology can be improved in order to challenge state-of-the-art OCR engines. A major drawback is the system's previously mentioned disability to recognize topologically similar characters. This could be improved if a global merging of local descriptors within character clusters – similar to the Bag-of-Features concept – would be exploited [SRE+05, MS06]. Another advantage of this approach would be a computational speed-up since not every local descriptor but solely one feature per character had to be classified. However, experiments proving that this methodology is still capable for recognizing partially visible characters would have to be carried out.

Another basic approach for improvements concerns the character localization. In the approach proposed, characters are localized according to interest points detected. This allows for localizing degraded characters as there is no need for binarization, but fails if background clutter impairs the interest point localization. Thus, a combined approach using texture information and the interest points' locations could be aspired. In addition, the features' probability histograms could be incorporated to the clustering which would emphasize clusters having similar class signatures.

A recent study by Zhang et al. [ZMLS07] focusing on local feature based object recognition proposes to exploit different interest point detectors and local descriptors. This approach could improve the character recognition system to the effect that topologically similar characters would be distinguished based on additional information extracted.

Appendix

Glag.	LaTeX	Class	ClassIdx	Glag.	LaTeX	Class	ClassIdx
ⰰ	a	a	1	ⰿ	n	n	17
ⰱ	b	b	2	ⰾ	o	o	18
ⱆ$_a$	v	v$_a$	130	ⰿ	p	p	19
ⱆ$_b$	v	v$_b$	131	ⰻ	r	r	21
ⰳ	g	g	4	ⰼ	s	s	22
ⰴ$_a$	d	d$_a$	150	ⰵ	t	t	23
ⰴ$_b$	d	d$_b$	151	ⰹⰹ	u	ou	24
ⰵ	e	e	6	ⰶ	f	f_A	26
ⰶ	Zz	zh	7	ⰽ	h	kh	27
ⰷ	9	dz	8	ⰿ	q	omega	29
ⰸ	z	z	9	ⰾⰻ	Ch	sht	30
ⰹ	i	i_A	10	ⰽ	c	c	31
		i_B1	11	ⰾⰽ	Cc	ch	32
ⰺ	y	i_B2	12	ⰾⰾ	Ss	sh	33
ⰻ	j	gj	13	ⰴ$_a$	4	jor	34
ⰼ	k	k	14	ⰴ$_b$	7	jer	38
ⰽ	l	l	15	ⰰ	w	jat	39
ⰾ	m	m	16	ⰹ	2	ju	40

Table 1: Glagolitic alphabet with corresponding class labels and class indices.

List of acronyms

CC	Connected Component
CV	Computer Vision
DoG	Difference-of-Gaussian
FAST	Features from Accelerated Segment Test
GLOH	Gradient Location-Orientation Histogram
HMM	Hidden Markov Model
ID3	Iterative Dichotomiser 3
JPEG	Joint Photographic Experts Group
k-NN	k-Nearest Neighbor
LoG	Laplacian-of-Gaussians
MSER	Maximally Stable Extremal Regions
NN	Neural Networks
NP-hard	**n**on-deterministic **p**olynomial-time hard
OCR	Optical Character Recognition
PCA	Principal Component Analysis
PDA	Personal Digital Assistant
RBF	Radial Basis Function
SIFT	Scale Invariant Feature Transform
SURF	Speeded Up Robust Features
SUSAN	Smallest Univalue Segment Assimilating Nucleus
SVM	Support Vector Machine

Bibliography

[AAFF05] Shahpour Alirezaee, Hassan Aghaeinia, Karim Faez, and Alireza Shayesteh Fard. An Efficient Feature Extraction Method for the Middle-Age Character Recognition. In *Proceedings of the International Conference on Intelligent Computing*, pages 998–1006, 2005.

[ABR64] A. Aizerman, E. M. Braverman, and L. I. Rozoner. Theoretical Foundations of the Potential Function Method in Pattern Recognition Learning. *Automation and Remote Control*, 25:821–837, 1964.

[Aea07] Julie D. Allen and et al. *The Unicode Standard, Version 5.0.0*. Boston, MA, Addison-Wesley, 2007.

[ARFMB05] Denis Arrivault, Noël Richard, Christine Fernandez-Maloigne, and Philippe Bouyer. Collaboration Between Statistical and Structural Approaches for Old Handwritten Characters Recognition. In *Graph-based Representations in Pattern Recognition*, pages 291–300, 2005.

[AV07] David Arthur and Sergei Vassilvitskii. k-means++: The Advantages of Careful Seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[AYV01] N. Arica and F. T. Yarman-Vural. An Overview of Character Recognition Focused on Off-line Handwriting. *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews*, 31(2):216–233, 2001.

[BGV92] Bernhard E. Boser, Isabelle Guyon, and Vladimir Vapnik. A Training Algorithm for Optimal Margin Classifiers. In *Proceedings of the 20th Annual Conference on Learning Theory*, pages 144–152, 1992.

[BMP02] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape Matching and Object Recognition Using Shape Contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.

[BP98] J.C. Bezdek and N.R. Pal. Some New Indexes of Cluster Validity. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 28(3):301–315, Jun 1998.

[BSB09] Syed Saqib Bukhari, Faisal Shafait, and Thomas M. Breuel. Foreground-Background Regions Guided Binarization of Camera-Captured Document Images. In *Proceedings of Third International Workshop on Camera-Based Document Analysis and Recognition*, jul 2009.

[BTG06] Herbert Bay, Tinne Tuytelaars, and Luc J. Van Gool. SURF: Speeded Up Robust Features. In *Proceedings of the European Conference on Computer Vision*, pages 404–417, 2006.

[Can86] J. Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, November 1986.

[CDS+06] Peter Carbonetto, Gyuri Dorkó, Cordelia Schmid, Hendrik Kück, and Nando de Freitas. A Semi-supervised Learning Approach to Object Recognition with Spatial Integration of Local Features and Segmentation Cues. In *Toward Category-Level Object Recognition*, pages 277–300, 2006.

[CL96] Richard G. Casey and Eric Lecolinet. A Survey of Methods and Strategies in Character Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):690–706, 1996.

[DHS00] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2000.

[DKS09] Markus Diem, Florian Kleber, and Robert Sablatnig. Analysis of Document Snippets as a Basis for Reconstruction. In Kurt Debattista, Cinzia Perlingieri, Denis Pitzalis, and Sandro Spina, editors, *Proceedings of the 10th International Symposium on Virtual Reality, Archaeology, and Cultural Heritage*, pages 101 – 108, 2009.

[DLS07] Markus Diem, Martin Lettner, and Robert Sablatnig. Registration of Multi-Spectral Manuscript Images. In *Proceedings of the 8th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST07*, pages 133–140, Brighton, UK, 2007.

[DS03] Gyuri Dorkó and Cordelia Schmid. Selection of Scale-Invariant Parts for Object Class Recognition. In *Proceedings of the International Conference on Computer Vision*, pages 634–640, 2003.

[DS09] Markus Diem and Robert Sablatnig. Recognition of Degraded Handwritten Characters Using Local Features. In *Proceedings of the 10th International Conference on Document Analysis and Recognition*, pages 221–225, Barcelona, Spain, 2009.

[DS10] Markus Diem and Robert Sablatnig. Recognizing Characters of Ancient Manuscripts. In *Proceedings of IS&T SPIE Conference on Computer Image Analysis in the Study of Art*, 2010. accepted.

[EIP97] Shimon Edelman, Nathan Intrator, and Tomaso Poggio. Complex Cells and Object Recognition. Unpublished: `http://kybele.psych.cornell.edu/~edelman/Archive/nips97.pdf`, 1997.

[FA91] William T. Freeman and Edward H. Adelson. The Design and Use of Steerable Filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991.

[FB09] Volkmar Frinken and Horst Bunke. Self-training Strategies for Handwriting Word Recognition. In *Proceedings of the 9th IEEE International Conference on Data Mining*, pages 291–300, 2009.

[FFJS08] Vittorio Ferrari, L. Fevrier, Frédéric Jurie, and Cordelia Schmid. Groups of Adjacent Contour Segments for Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(1):36–51, 2008.

[FHRKV94] L Florack, B M ter Haar Romeny, J J Koenderink, and M A Viergever. General Intensity Transformations and Differential Invariants. In *Journal of Mathematical Imaging and Vision*, volume 4, pages 171–187, 1994.

[FPF+09] Volkmar Frinken, Tim Peter, Andreas Fischer, Horst Bunke, Trinh Minh Tri Do, and Thierry Artières. Improved Handwriting Recognition by Combining Two Forms of Hidden Markov Models and a Recurrent Neural Network. In *Proceedings of the 13th International Conference on Computer Analysis of Images and Patterns*, pages 189–196, 2009.

[FPZ03] Robert Fergus, Pietro Perona, and Andrew Zisserman. Object Class Recognition by Unsupervised Scale-Invariant Learning. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 264–271, 2003.

[FWL+09] Andreas Fischer, Markus Wüthrich, Marcus Liwicki, Volkmar Frinken, Horst Bunke, Gabriel Viehhauser, and Michael Stolz. Automatic Transcription of Handwritten Medieval Documents. *International Conference on Virtual Systems and MultiMedia*, 0:137–142, 2009.

[GMU96] Luc J. Van Gool, Theo Moons, and Dorin Ungureanu. Affine/ Photometric Invariants for Planar Intensity Patterns. In *Proceedings of the 4th European Conference on Computer Vision*, volume 1, pages 642–651, London, UK, 1996. Springer-Verlag.

[GNP09] Basilios Gatos, Konstantinos Ntirogiannis, and Ioannis Pratikakis. Document Image Binarization Contest (DIBCO 2009). In *Proceedings of International Conference on Document Analysis and Recognition*, pages 1375–1382, 2009.

[Han33] Paul W. Handel. Statistical Machine, US Patent 1,915,993 1933.

[HBV01] Michel Herbin, N. Bonnet, and Philippe Vautrot. Estimation of the Number of Clusters and Influence Zones. *Pattern Recognition Letters*, 22(14):1557–1568, 2001.

[HS88] C. Harris and M. Stephens. A Combined Corner and Edge Detector. In *4th ALVEY Vision Conference*, pages 147–151, 1988.

[Hul98] Jonathan J. Hull. Document Image Skew Detection: Survey and Annotated Bibiliography. In Jonathan J. Hull and Suzanne L. Taylor, editors, *Document Analysis System II, World Scientific*, pages 40–64, 1998.

[JH97] Andrew Edie Johnson and Martial Hebert. Recognizing Objects by Matching Oriented Points. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 684–689, 1997.

[Jol02] I. T. Jolliffe. *Principal Component Analysis*. Springer, 2nd edition, October 2002.

[KB01] Timor Kadir and Michael Brady. Saliency, Scale and Image Description. *International Journal of Computer Vision*, 45(2):83–105, 2001.

[KC09] Jung Gap Kuk and Nam Ik Cho. Feature Based Binarization of Document Images Degraded by Uneven Light Condition. In *Proceedings of International Conference on Document Analysis and Recognition*, pages 748–752, 2009.

[KS04] Yan Ke and Rahul Sukthankar. PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 506–513, 2004.

[KS08] Florian Kleber and Robert Sablatnig. High Resolution Imaging for Cultural Heritage Applications. In A. Kuijper, B. Heise, and L. Muresan, editors, *Proceedings of the 32nd Workshop of the Austrian Association for Pattern Recognition*, volume 232, pages 137–178, 2008.

[KSGM08] Florian Kleber, Robert Sablatnig, Melanie Gau, and Heinz Miklas. Ancient Document Analysis Based on Text Line Extraction. In *Proceedings of the 19th International Conference on Pattern Recognition (ICPR 2008)*, Tampa, Florida, USA, 2008.

[KvD87] J J Koenderink and A J van Doom. Representation of Local Geometry in the Visual System. *Biological Cybernetics*, 55(6):367–375, 1987.

[LBH09] Christoph H. Lampert, Matthew B. Blaschko, and Thomas Hofmann. Efficient Subwindow Search: A Branch and Bound Framework for Object Localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2129–2142, 2009.

[Lin94] Tony Lindeberg. Scale-Space Theory: A Basic Tool for Analysing Structures at Different Scales. *Journal of Applied Statistics*, 21(2):224–270, 1994.

[Llo82] S. Lloyd. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, Mar 1982.

[Low99] David G. Lowe. Object Recognition from Local Scale-Invariant Features. In *Proceedings of the International Conference on Computer Vision*, pages 1150–1157, 1999.

[Low04] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[LRM04] Victor Lavrenko, Toni M. Rath, and R. Manmatha. Holistic Word Recognition for Handwritten Historical Documents. In *Proceedings of the International Conference on Document Image Analysis for Libraries*, pages 278–287, 2004.

[LSP03] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. A Sparse Texture Representation Using Affine-Invariant Regions. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 319–326, 2003.

[LSZT07] Laurence Likforman-Sulem, Abderrazak Zahour, and Bruno Taconet. Text Line Segmentation of Historical Documents: a Survey. *International Journal on Document Analysis and Recognition*, 9(2):123–138, 2007.

[LT07] Shijian Lu and Chew Lim Tan. Thresholding of Badly Illuminated Document Images Through Photometric Correction. In *ACM Symposium on Document Engineering*, pages 3–8, 2007.

[Mac67] J. B. Macqueen. Some Methods for Classification and Analysis of Multivariate Observations. In *Procedings of the Fifth Berkeley Symposium on Math, Statistics, and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[Mai03] F Mairinger. *Strahlenuntersuchung an Kunstwerken*. E. A. Seemann Verlag, 2003.

[MCUP04] Jiri Matas, Ondrej Chum, Martin Urban, and Tomás Pajdla. Robust Wide-Baseline Stereo from Maximally Stable Extremal Regions. *Image and Vision Computing*, 22(10):761–767, 2004.

[MEE+09] Vincent Malleron, Véronique Eglin, Hubert Emptoz, Stéphanie Dord-Crouslé, and Philippe Régnier. Text Lines and Snippets Extraction for 19th Century Handwriting Documents Layout Analysis. In *Proceedings of International Conference on Document Analysis and Recognition*, pages 1001–1005, 2009.

[MGK+08] Heinz Miklas, Melanie Gau, Florian Kleber, Markus Diem, Martin Lettner, Maria Vill, Robert Sablatnig, Manfred Schreiner, Michael Melcher, and Ernst-Georg Hammerschmid. St. Catherine' s Monastery on Mount Sinai and the Balkan-Slavic Manuscript-Tradition. In Heinz Miklas and Anissava Miltenova, editors, *Slovo: Towards a Digital Library of South Slavic Manuscripts. Proceedings of the International Conference*, pages 13–36, Sofia, Bulgaria, 2008. "Boyan Penev" Publishing Center.

[Mik00] Heinz Miklas, editor. *Glagolitica - Zum Ursprung der slavischen Schriftkultur*. Verlag der Österreichischen Akademie der Wissenschaften, 2000.

[Mik02] Krystian Mikolajczyk. *Detection of Local Features Invariant to Affine Transformations*. PhD thesis, Institut National Polytechnique de Grenoble, France, 2002.

[MLS06] Krystian Mikolajczyk, Bastian Leibe, and Bernt Schiele. Multiple Object Class Detection with a Generative Model. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 26–36, 2006.

[Mor81] Hans P. Moravec. Rover Visual Obstacle Avoidance. In *Proceedings of the International Joint Conferences on Artificial Intelligence*, pages 785–790, 1981.

[MS01] Krystian Mikolajczyk and Cordelia Schmid. Indexing Based on Scale Invariant Interest Points. In *Proceedings of the International Conference on Computer Vision*, pages 525–531, 2001.

[MS05] Krystian Mikolajczyk and Cordelia Schmid. A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[MS06] Marcin Marszalek and Cordelia Schmid. Spatial Weighting for Bag-of-Features. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 2118–2125, 2006.

[MTS+05] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and Luc J. Van Gool. A Comparison of Affine Region Detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005.

[NGP+07] Kostas Ntzios, Basilios Gatos, Ioannis Pratikakis, Thomas Konidaris, and Stavros J. Perantonis. An Old Greek Handwritten OCR System Based on an Efficient Segmentation-free Approach. *International Journal on Document Analysis and Recognition*, 9(2-4):179–192, 2007.

[NGP09] Konstantinos Ntirogiannis, Basilios Gatos, and Ioannis Pratikakis. A Modified Adaptive Logical Level Binarization Technique for Historical Document Images. In *Proceedings of International Conference on Document Analysis and Recognition*, pages 1171–1175, 2009.

[Nib90] Wayne Niblack. *An Introduction to Digital Image Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1990.

[Ots79] N. Otsu. A Threshold Selection Method from Grey-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, January 1979.

[PHA09] Stefan Pletschacher, Jianying Hu, and Apostolos Antonacopoulos. A New Framework for Recognition of Heavily Degraded Characters in Historical Typewritten Documents Based on Semi-Supervised Clustering. In *Proceedings of International Conference on Document Analysis and Recognition*, pages 506–510, 2009.

[PS00] Réjean Plamondon and Sargur N. Srihari. On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):63–84, 2000.

[PSS86] P.W. Palumbo, P. Swaminathan, and S.N. Srihari. Document Image Binarization: Evaluation of Algorithms. *Proceedings of IS&T SPIE Conference on Computer Image Analysis in the Study of Art*, 697:278–285, 1986.

[QMO+05] Pedro Quelhas, Florent Monay, Jean-Marc Odobez, Daniel Gatica-Perez, Tinne Tuytelaars, and Luc J. Van Gool. Modeling Scenes with Local Descriptors and Latent Aspects. In *Proceedings of the International Conference on Computer Vision*, pages 883–890, 2005.

[Ram06] Deva Ramanan. Learning to Parse Images of Articulated Bodies. In *Proceedings of the Conference on Neural Information Processing Systems*, pages 1129–1136, 2006.

[RD06] Edward Rosten and Tom Drummond. Machine Learning for High-Speed Corner Detection. In *European Conference on Computer Vision*, pages 430–443, 2006.

[RK09] Oriol Ramos and Dimonsthenis Karatzas, editors. *10th International Conference on Document Analysis and Recognition, ICDAR 2009, Barcelona, Spain, 26-29 July 2009*. IEEE Computer Society, 2009.

[RM07] Toni M. Rath and R. Manmatha. Word Spotting for Historical Documents. *International Journal on Document Analysis and Recognition*, 9(2-4):139–152, 2007.

[RS06] Deva Ramanan and Cristian Sminchisescu. Training Deformable Models for Localization. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 206–213, 2006.

[SB97] Stephen M. Smith and J. Michael Brady. SUSAN - A New Approach to Low Level Image Processing. *International Journal of Computer Vision*, 23(1):45–78, 1997.

[SM97] Cordelia Schmid and Roger Mohr. Local Grayvalue Invariants for Image Retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, 1997.

[SP00] Jaakko J. Sauvola and Matti Pietikäinen. Adaptive Document Image Binarization. *Pattern Recognition*, 33(2):225–236, 2000.

[SRE+05] Josef Sivic, Bryan C. Russell, Alexei A. Efros, Andrew Zisserman, and William T. Freeman. Discovering Objects and their Localization in Images. In *Proceedings of the International Conference on Computer Vision*, pages 370–377, 2005.

[Tan09] Hiroshi Tanaka. Threshold Correction of Document Image Binarization for Ruled-line Extraction. In *Proceedings of International Conference on Document Analysis and Recognition*, pages 541–545, 2009.

[Tau35] Gustav Tauschek. Reading Machine, US Patent 2,026,329 1935.

[vBSB09] Joost van Beusekom, Faisal Shafait, and Thomas M. Breuel. Resolution Independent Skew and Orientation Detection for Document Images. In *Proceedings of IS&T SPIE Conference on Computer Image Analysis in the Study of Art*, pages 1–10, 2009.

[VC74] Vladimir Vapnik and Alexey Chervonenkis. *Theory of Pattern Recognition*. Nauka, Moscow, 1974.

[VGSP08] G. Vamvakas, B. Gatos, N. Stamatopoulos, and S.J. Perantonis. A Complete Optical Character Recognition Methodology for Historical Documents. *IAPR International Workshop on Document Analysis Systems*, 1:525–532, 2008.

[Vin02] Alessandro Vinciarelli. A survey on off-line Cursive Word Recognition. *Pattern Recognition*, 35(7):1433–1446, 2002.

[XCL+07] Y. Xi, Y. Chen, Q. Liao, L. Winghong, F. Shunming, and D. Jiangwen. A Novel Binarization System for Degraded Document Images. In *Proceedings of International Conference on Document Analysis and Recognition*, pages 287–291, 2007.

[Yos05] Itay Bar Yosef. Input Sensitive Thresholding for Ancient Hebrew Manuscript. *Pattern Recognition Letters*, 26(8):1168–1173, 2005.

[ZMLS07] Jianguo Zhang, Marcin Marszalek, Svetlana Lazebnik, and Cordelia Schmid. Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. *International Journal of Computer Vision*, 73(2):213–238, 2007.