FAKULTÄT
FÜR !NFORMATIK

Faculty of Informatics

Novel Methods for Writer Identification and Retrieval

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der technischen Wissenschaften

by

Stefan Fiel
Registration Number 0026641

to the Faculty of Informatics
at the TU Wien

Advisor: a.o.Univ.-Prof. Dr.techn. Robert Sablatnig

The dissertation has been reviewed by:

(a.o.Univ.-Prof. Dr.techn. Robert Sablatnig)

(Dr. Basilis Gatos)

Wien, 22.11.2015

(Stefan Fiel)

Erklärung zur Verfassung der Arbeit

Stefan Fiel
Wassergasse 16/14, 1030 Wien

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.

(Ort, Datum)

(Unterschrift Verfasser)

Danksagung

Ich möchte die Möglichkeit ergreifen an dieser Stelle einigen Leuten zu danken die mich in meinem Leben bisher unterstützt haben und ohne die auch diese Arbeit nicht möglich gewesen wäre. Ein herzliches Dankeschön gilt meinem Betreuer Robert Sablatnig, der mir immer mit Rat und Tat zur Seite gestanden ist und auch für ein positives Arbeitsumfeld gesorgt hat. Weiters möchte ich mich bei meinem Gutachter Basilis Gatos bedanken, der sich die Zeit genommen hat diese Arbeit durchzulesen und auch zu bewerten. Ein großes Dankeschön gilt natürlich auch meinen Arbeitskollegen des Computer Vision Labs der TU Wien. Ohne ihre Hilfestellungen in vielen Bereichen wäre diese Arbeit wohl nicht möglich gewesen. Zusätzlich zu allen fachlichen Qualitäten beweist jeder davon auch tagtäglich seine menschlichen Qualitäten und sorgt somit für ein erheiterndes, aber auch sehr produktives Arbeitsklima. Im Speziellen möchte ich mich hier auch bei Florian Kleber und Markus Diem bedanken, mit denen ich die Freude hatte die letzten 5 Jahre das Büro zu teilen. Neben guten fachlichen Diskussion, kam auch der Spaß nie zu kurz und natürlich die Einführung des Montagsbiers vor vielen Jahren sorgt dafür, dass das miteinander wirklich auf einzigartige Weise funktioniert. Weiters möchte ich noch Rainer Planinc erwähnen, die morgendlichen Diskussionen beim ersten Kaffee waren oft genug schon Anlass mich für den Tag zu motivieren.

Zusätzlich möchte ich mich bei meiner Familie bedanken. Meinen Eltern, Maria und Manfred, die mich all die Jahre tatkräftig unterstützt haben gilt es hier einmal ein großes "Dankeßu sagen. Dieser Dank gilt auch meinen Geschwistern, Reinhard, Dietmar und Regina, die mich auch von Anfang an immer begleitet haben und ohne deren Unterstützung vieles in meinem Leben nicht möglich gewesen wäre. Auch bei ihren Partnern, Katrin, Ulrike und Till, bei denen ich wirklich froh sind, dass sie über den Weg der Liebe in mein Leben getreten sind und auch immer für mich da sind, wenn ich sie brauche. Natürlich darf ich hier meine Nichten und Neffen, Franziska, Constanze, Valentin, Theresa, Magdalena und Juliane, nicht vergessen. Sie schaffen es immer wieder zu erinnern, dass es auch andere wichtige Sachen im Leben gibt für die ich schon lange den Blick verloren hätte.

Schlussendlich möchte ich mich noch bei meiner Partnerin Daniela für all die glücklichen und fröhlichen Momente/Stunden/Tage/Monate/Jahre die wir schon hatten und die wir noch haben werden bedanken. Ich bin froh, dass wir uns getroffen haben und dass wir uns gegenseitig durch unser Leben begleiten.

Abstract

Writer identification is the task of identifying the writer of a handwritten document. Therefore, a set of documents where the authors are known has to be available in advance. A feature is generated for a new document image containing handwriting and then this feature is compared to the features generated on the set of documents. The writer of the document with the highest similarity is then assigned as writer to the new document. Writer identification can be used e.g. for tasks in forensics and for historical document analysis. In contrast to this, writer retrieval is to receive a ranking of the pages in the set of documents sorted according to the similarity of handwriting. It allows for searching for documents which may have been written by the same author and thus can be used for clustering a not indexed set of documents according to the individual handwriting.

State-of-the-art methods calculate features for writer identification on the contours of the characters, so pre-processing steps are needed to extract this contour. In contrast to this in this thesis, three novel approaches for writer identification and writer retrieval are presented. The first is based on the bag of words approach, which is well known for object recognition. SIFT features are calculated on the handwriting and then an occurrence histogram is generated. This histogram is then used as feature vector for identification or used for sorting of the documents. The second method is based on the Fisher vector. Again, SIFT features are generated on the handwriting, but this time the gradient vectors of a Gaussian Mixture Model (GMM) are used to generate the feature vector for writer identification. Additionally, the SIFT features are modified such that a distinction between the upper and lower profile of the handwriting, which are distinctive features for writers, is possible. The last method is based on Convolutional Neural Network (CNN). To the best knowledge of the author, this is the first method which brings the field of deep learning to writer identification and retrieval. A CNN is trained on image patches and the classification layer is cut off and the second last layer is used as feature vector for this patch. The mean vector of all patches on one page is the feature vector for the handwriting and is used for identification and retrieval.

The methods presented are evaluated and compared to the state of the art on different scientific databases and additionally on a historic dataset using common evaluation metrics for writer identification. The evaluations show that the three methods proposed outperform the state of the art on many of the different tasks on these datasets. Advantages and possible weaknesses are discussed. The methods proposed achieve good results (>90%) on every dataset used for evaluation.

Kurzfassung

Als Schreiberidentifikation bezeichnet man die Aufgabe, einem Text, dessen Autor unbekannt ist, einen Schreiber zuzuordnen. Hierfür wird eine Datenbank von Dokumenten benötigt, für die die Schreiber bereits bekannt sind. Ein Merkmal, das die Handschrift beschreibt, wird auf einem neuen Dokument generiert und dieses wird dann mit den bereits vorberechneten Merkmalen in der Datenbank abgeglichen. Nach einem bestimmten Ähnlichkeitsmaß kann dann die ähnlichste Handschrift in der Datenbank gefunden werden und der Autor des ähnlichsten Dokuments wird dann dem neuen Dokument zugewiesen. Schreiberidentifkation wird z.B. in Bereichen der Forensik sowie auch bei der Analyse von historischen Dokumenten benötigt. Writer retrieval bezeichnet das Suchen von Dokumenten mit ähnlicher Handschrift. Hierfür müssen die Schreiber in der Datenbank nicht bekannt sein. Wieder werden die Merkmale auf der Handschrift generiert und die in der Datenbank befindlichen Dokumente nach Ähnlichkeit gereiht. Dadurch können nicht indexierte Datenbanken von Dokumenten durchforstet und nach ähnlichen Schriftbildern gruppiert werden.

Während State-of-the-art Methoden die Merkmale für eine Schreibererkennung meist auf den Konturen der Buchstaben erkennen, werden in dieser Arbeit zwei neue Methoden vorgestellt, die auf lokalen Merkmalen basieren. Die erste Methode beruht auf dem Bag of Words Modell, das in der Objekterkennung häufig eingesetzt wird. SIFT Features werden auf der Handschrift berechnet und ein Häufigkeitshistogramm wird generiert. Dieses Histogramm wird dann für die Berechnung der Ähnlichkeit verwendet. Die zweite Methode basiert auf dem Fisher Vektor. Wieder werden SIFT Features auf der Handschrift berechnet, aber bei dieser Methode werden die Gradientvektoren eines Gaussian Mixture Models (GMM) verwendet um die Handschrift zu beschreiben. Zusätzlich wird eine Modifikation der SIFT Features verwendet, die eine Unterscheidung zwischen dem oberen und unteren Profil der Handschrift, die charakteristisch für unterschiedliche Schreiber sind, zulässt. Die dritte Methode verwendet Convolutional Neural Networks (CNN). So weit bekannt ist, ist dies die erste Methode, die das Feld des Deep Learnings für Schreibererkennung verwendet. Ein CNN wird auf Ausschnitten des Bildes der Handschrift trainiert. Der Klassifikationslayer wird abgeschnitten und die Aktivierung des vorletzten Layers wird als Merkmalsvektoren für den jeweiligen Ausschnitt verwendet. Der Durchschnittsvektor über alle Ausschnitte wird dann als Merkmal für die Handschrift verwendet, mit dessen Hilfe dann die Identifikation bzw. die Reihung nach Ähnlichkeit erfolgt.

Die vorgestellten Methoden werden auf wissenschaftlichen Datenbanken evaluiert und mit dem State of the Art verglichen. Zusätzlich wird noch eine Datenbank aus historischen Dokumenten zur Evaluierung herangezogen. Die Resultate zeigen, dass die vorgestellten Methoden bessere Ergebnisse liefern als der State of the Art.

Contents

CHAPTER 1

Introduction

Writer identification is the task of assigning a writer to a document of which the author is previously unknown. For this task a database of documents with known authors has to be available. Features are generated on the handwriting of all documents in the database, and when identifying the author of a document the same features are generated and a comparison of the features is done. With a certain distance measurement the most similar document in the database can be found and the author of this particular document is then assigned to the new one. This allows for finding out the author of a specific document and can be used for example by the police for ransom threats. In contrast to this, the task of writer retrieval is to find documents in a database which have the most similar handwriting. The authors of the documents in the database do not have to be known. After the comparison of the features the documents in the database are ranked according to the distance and presented to the user as the most similar handwritings. Naturally, the most similar documents should originate from the same writer as the reference document if possible. Writer retrieval allows finding the most similar documents concerning the handwriting in a set of documents. It allows the users to look for documents which may have been written by the same writer. This can be used for example by historians for exploring large non-indexed archives of libraries.

Figure 1.1 illustrates the difference between identification and retrieval. In Figure 1.1 a) the identification is illustrated. A database with the handwriting of known writers is available and for a new document image the system should return the identification of its writer. Figure 1.1 b) illustrates the retrieval task. A set of documents, with known or unknown writers, is given and for a new document image a ranking according to the similarity of the handwriting of the document images in the database should be returned.

This thesis is giving an overview of the current state of the art of writer identification and retrieval and is describing three different methods, which have been developed in the scope of this thesis, in more detail. The proposed methods generate a feature vector for each page which is then used to calculate the similarity between two different document images. All three methods have in common that they originate from the field of object- respectively image recognition. This is also the novelty of these approaches, because state-of-the-art methods, such as the ones

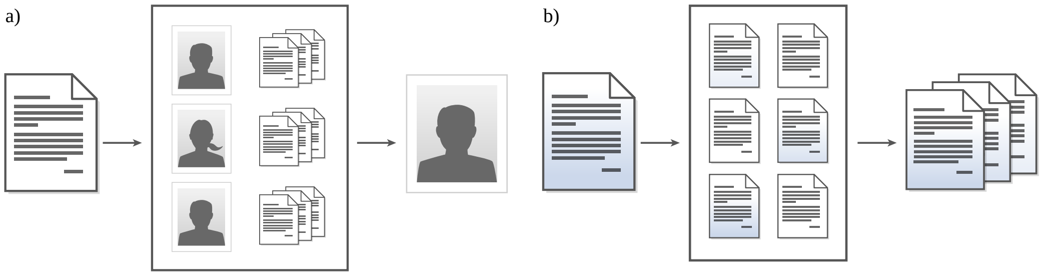


Figure 1.1: Illustration of the difference between writer identification and retrieval. Writer identification returns the identity of the documents' writer, whereas writer retrieval returns a ranking of the documents in the dataset according to the similarity of the handwriting.

described in this work, are mostly analyzing the contour of the characters for an identification of a writer. To analyze the character itself, preprocessing steps are needed to get a good segmentation of the characters including binarization, text line detection, and segmentation of characters. Since each of these steps can introduce new errors which can affect the accuracy of the identification or the ranking of the retrieval, one goal of this work is to avoid unnecessary preprocessing steps.

1.1 Motivation

The manual identification of the author of a handwriting sample is a very time consuming task which also requires expertise [27] [42]. The investigated handwriting has to be compared with numerous others and each comparison has to be made carefully and consciously. Nevertheless, the identification of writers is needed for forensics or historical documents. In forensics, for example, the author of threatening or ransom letters have to be identified. The police has databases of old cases with the handwriting and a detective has to compare the new sample with the old ones. Thus, only a few authors of such letters can be identified. A semi-automated tool would improve the overall efficiency and accuracy in the identification by providing more complete detailed evidence to support their expert opinion [77] [29]. Historians can also make use of an automated method, for example on medieval handwritings. At this time manuscripts played a key role as a medium of the transmission and exchange of ideas, and the reception and transformation of classical and contemporary erudition, knowledge, and science. The process of transmission has taken place in the frame of scholarly networks. The manuscripts circulated within these networks and were exchanged, copied, corrected, selected, and reworked. Historians are interested in the different hands who have contributed to a manuscript and also the traces of these writers through the network. Another possibility for the application of writer retrieval in a historic context is the Fall of the (Berlin-)Wall in 1989. The Stasi (Secret police in East Germany) tried to destroy parts of the secret records of the citizens which they had collected over the years, by tearing them up by hand [78]. Currently these file are being restored manually [73] and also with an automated computer system [64]. The result of this reconstruction are millions of single and not indexed pages. To restore the records again the similarity of the handwriting can be an important feature to cluster the documents beforehand, like in the system proposed by Diem et al. [18].

The difference between historical and modern documents is mainly the fact that historic document images do not have uniform background and noise may be present. The condition of the document has influence on the identification process because local features may be calculated on noisy parts of the image and influence the overall feature of the page. These pre-processing steps may introduce new errors, thus two of three methods developed within the scope of this thesis avoid these pre-processing steps. Additionally, they should not be dependent on grayscale or color images and should also have a good performance on binarized images, if preprocessing is necessary or only binarized images are available.

Working on historic documents is the main motivation of this thesis to propose a system for offline writer identification. An offline system only requires the image as input, which is the only information available when dealing with historic handwriting. In contrast to this, online handwriting collections contain a lot more information like the "pen-point movement, pen-point pressure, pen-point direction, pen-point velocity and acceleration" [85]. Since online handwriting has additional information about the writer available the results tend to be better. Thus, these systems are used for verification of the writer. In addition, knowing the identity of the writer in modern environments (e.g. smart meeting rooms) provides additional value e [83].

The challenges for writer identification and writer retrieval are that the handwriting of a person is not always exactly the same but varies according to some conditions: these might be the change of the material like the use of different pens or different paper types; the situation the writer is currently experiencing like if the text has been written in a hurry or if the writer is distracted by other persons; and also the condition of the author itself like fatigue or alcohol and other drugs [74]. Figure 1.2 shows some of these variabilities [79]: (a) affine transforms (b) neurobiomechanical variability (c) sequence variability (d) allographic variations. The sequence variability can only be detected by online handwriting tools.

Figure 1.3 a) shows an image taken from the CVL Database [44] where a writer used two pens in one document. It can be seen that, at least for humans, the writing differs clearly, even the slant of the writing has changed. Figure 1.3 b) shows a document image where the writer changed the writing style during writing. Apparently the writer was distracted by something or someone, or the text was written very fast. The slant increases, the shape of different characters is changing, and thus the handwriting looks different.

Methods also have to face the problem that a writer does not always write the same way even if there is no influence from outside. Figure 1.4 shows one part of a document image where the German word "Dann" is written 4 times within the same document. The "D" of this writer differs completely but also the remaining characters are written in different ways. Therefore, even within one document there are small variations of the handwriting, and writer identification and retrieval methods should be able to handle these variations. Another challenge for such methods is that the handwriting of humans changes over years [81]. This challenge is not covered by any available dataset since it would include a collection of the handwriting of the same persons for years but it is clear that these changes will make an identification task harder.

When performing writer identification, one of the first tasks of the state-of-the-art methods mentioned in Chapter 2 is a binarization of the image. However, on historical document im-

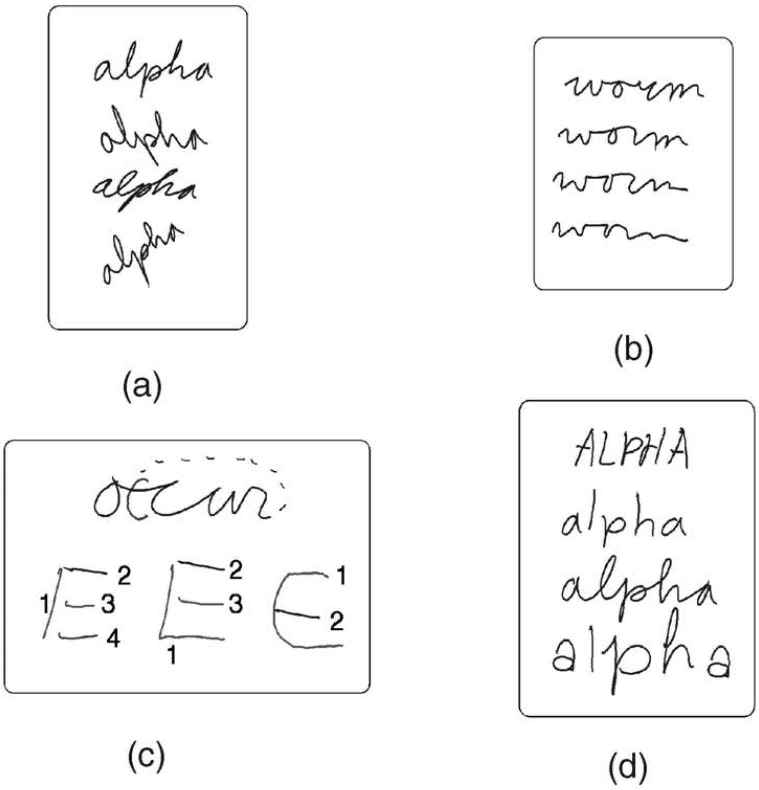
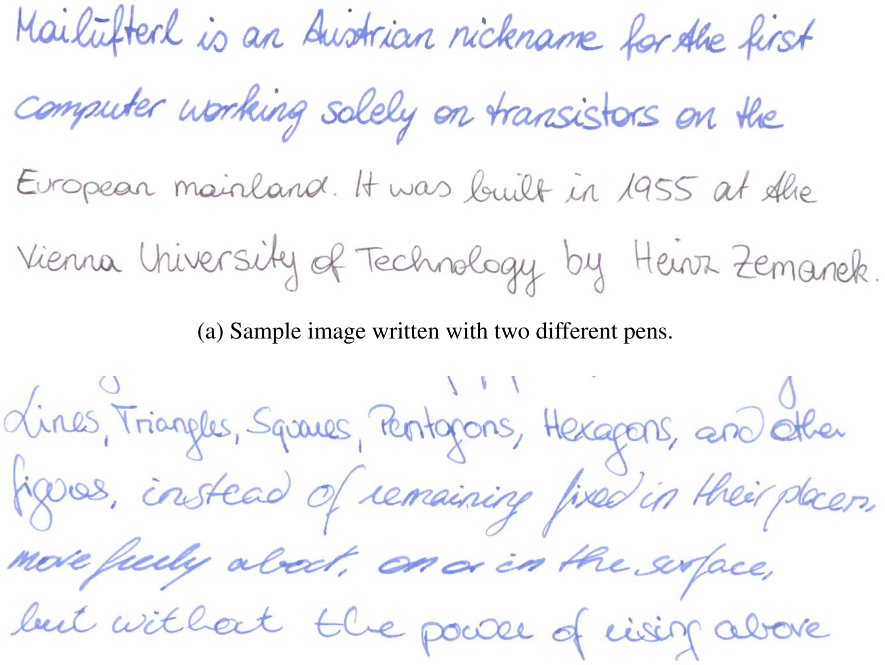


Figure 1.2: Different variabilities of handwriting. (a) affine transformations (b) neurobiomechanical variability (c) sequencing variability (d) allographic variations (taken from [79]).

ages this preprocessing step is still a challenging task, which can be seen in the "ICFHR2014 Competition on Handwritten Document Image Binarization (H-DIBCO 2014)" [66]. Two of the methods developed within the scope of this work avoid this binarization step and also others like text line segmentation or deskewing of the documents . For the third method preprocessing methods are applied, but these are not described in detail in this work since they are either very common or simple.

1.2 Problem Statement and Aim of the Work

The aim of this work is to develop a method for writer identification and writer retrieval which gives a reliable identification rate and also a good retrieval rate. The assessment of the results takes place by comparing the result of the proposed method against other state-of-the-art methods which are presented in this thesis. These methods have either published the results on common datasets or participated at the different writer identification contests ([55], [53], [54]). Also, the amount of text present on a document image is considered in this work. The methods developed should work on full text pages but should also give good results if only a few lines of text are present in the image. The task of identification and retrieval when there are only a few



Mailüfterl is an Austrian nickname for the first
computer working solely on transistors on the
European mainland. It was built in 1955 at the
Vienna University of Technology by Heinz Zemanek.

(a) Sample image written with two different pens.

Lines, Triangles, Squares, Pentagons, Hexagons, and other
figures, instead of remaining fixed in their places,
move freely about, on or in the surface,
but without the power of rising above

(b) Sample image in which the writers was apparently distracted and changed the writing within the text.

Figure 1.3: Sample images with variations of distraction and different pens.

lines of text is more important for real world applications, since e.g. in the Stasi files on many pages only a couple of lines of handwritten text are present and the rest of the page is written with a typewriter. The method should also work on forms which have been filled in. Localization or detection of text is not within the scope of this work. Thus, scientific databases are used for evaluation, since they can be preprocessed using well known methods from the literature based on the defined layouts and conditions. Also, other preprocessing steps, which may be needed to prepare data for writer identification or retrieval methods, are not addressed in this work. The proposed method is evaluated on multiple scientific databases and on a historic dataset, which due to copyright issues, is not publishable. All these datasets are presented in Chapter 2. The databases used are only in Latin and Greek scripts (except for the historic database), there are multiple databases available in Arabic [61], Chinese [51], and other languages. Since the methods proposed have the main focus on Latin scripts, they were not evaluated on these datasets.

Writer verification is the task of verifying if actually an author has written a specific document by calculating features on both documents and compare them to each other. Therefore, a dataset with known authors is available and one has to calculate the possibility that one of the writers in the database has written this document. This task sounds similar to the task of writer identification, but there is a small difference. According to Bensefia et al. [4], writer identification "provides a subset of relevant candidate documents, on which complementary analysis will be achieved by the experts". In contrast to this writer verification is a task which "must come to a conclusion about two samples of handwriting and determines whether they are written by the

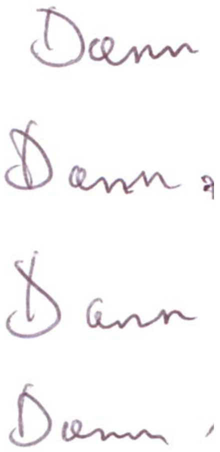


Figure 1.4: Part of a sample document image where the same word is written four times by the same writer and illustrates the inter-writer variability.

same writer or not". Figure 1.5 shows an overview of different tasks for handwriting processing tasks as seen by Atanasiu et al. [1]. According to them, the fundamental difference between these tasks is the output of the system. For a verification the output is a logical statement if the specific writer has written this document or not whereas for an identification task the output is the ID or the name of the writer and for retrieval the most similar documents according to the handwriting. Writer classification generates clusters with the different handwritings, e.g. female or male author. This work only deals with the task of writer identification and writer retrieval.

Since historic documents should also be processed with the proposed methods, the use of offline data is mandatory. Online data is not available for this kind of documents. The methods may be adopted to online data by incorporating the additional information into the feature vector, but no further analysis has been done.

Thus, the problem statement of this work can be summarized as follows:

- Is a reliable writer identification and retrieval method possible without being dependent on preprocessing steps?
- Can the grayvalue information be exploited for this task?
- Does this method also work if only a few lines of handwritten text are present?
- Can it be adopted easily for historic databases?

Furthermore, a scientific database is presented which provides data for writer identification, writer retrieval, and word spotting. In contrast to other databases it has an equal distribution of number of pages per writer and consists of more pages.

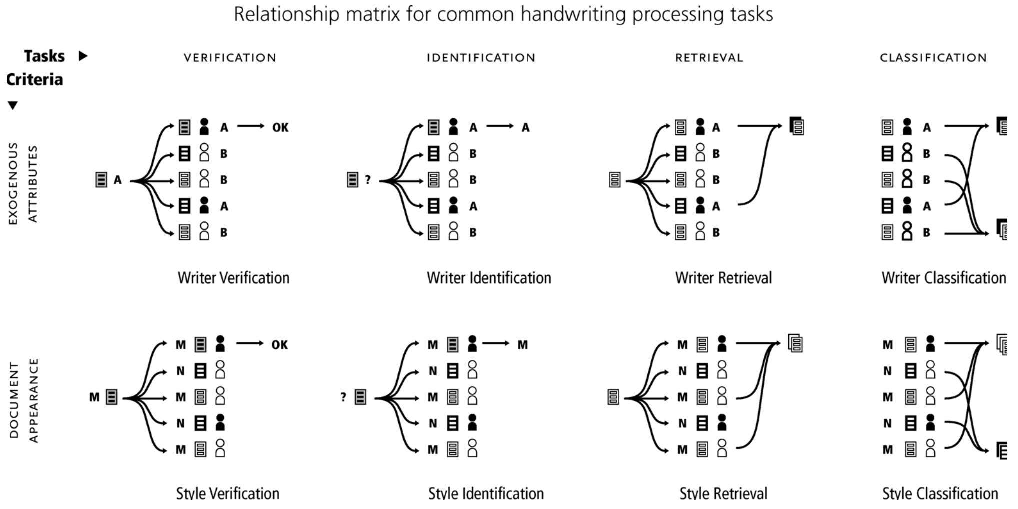


Figure 1.5: Common tasks when dealing with handwriting. The difference between most tasks is the number or type of the output. While verification only gives a true or false (or perhaps percentages), the identification gives the ID of the writer, writer retrieval returns the most similar documents to the reference document and the classification returns groups of different handwritings (Figure taken from [1]).

1.3 Methodological Approach

Nearly two-thirds of the state-of-the-art methods (nearly two thirds) which are presented in Chapter 2, calculate their features directory on the characters itself. Within the time of this work, some methods also focused on using the local information of the neighborhood for writer identification. Within this work methods from the field of object- and image recognition are analyzed and adopted to perform this task. Since errors which may be introduced during preprocessing steps are hard to detect and to correct, one focus of this work is to be independent from these steps. Another focus is that a new method should be working on modern (in the case of this work only Latin and Greek) scripts, but should also be applicable to historic handwriting with minimal effort.

The methodological approach includes an analysis of the current methods for writer identification, but also of some concepts used for object recognition and image classification. These methods are then adopted for the task of writer identification and retrieval and are evaluated on scientific databases.

1.4 Structure of Work

The structure of this work is as follows: In Chapter 2 scientific datasets for writer identification are introduced. All datasets have been used within the scope of this work, and most of the state-of-the-art methods, which are presented in Sections 2.2 and 2.3, also use these databases. In Section 2.4 a comparison of the methods presented is presented. Chapter 3 introduces the concepts which are used for the methods developed within the scope of this work, whereas Chapter 4 described these methods. The next chapter presents the evaluations of the three methods proposed on various datasets and also compares their performance. At the end a conclusion is drawn in Chapter 6.

CHAPTER 2

Related Work

This chapter gives an overview of the current state of the art of writer identification and writer retrieval. First, the most popular scientific datasets are presented with all the properties to allow an objective comparison of the different methods, as well as a historic dataset which is also used for evaluating the methods proposed in this thesis. The evaluation methods used are also presented in this section. Afterwards the current state-of the art of writer identification is presented. The methods have been divided into two different categories. First, methods which calculate the features directly on the character, thus requiring a binarization step. Second, texture based writer identification, which assume the handwriting on the paper as texture and calculate the features on the complete writing. Some methods also use a preprocessing step, like the removal of the spaces between two text lines. The performances of the different methods are then compared to each other. Last, a short summary is given.

2.1 Datasets

This section describes the most popular datasets for writer identification which are freely available. An additional dataset, which is only used in the scope of this work, is presented on which the proposed methods are evaluated to show their capabilities of dealing with historic documents. In Table 2.1 the key data of the dataset which are presented in this section are listed. In two datasets the number of pages from each writer is not equally distributed and thus the evaluation results are more difficult to interpret and to compare since the results can be dependent on specific writers. This dependence can either be positive if the performance is good for a writer who has more documents in the database or can decrease the influence if the accuracy for such a writer is low.

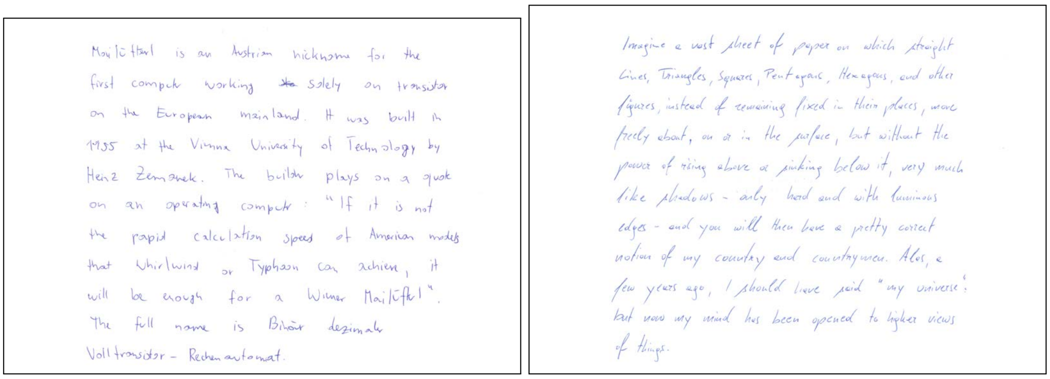
2.1.1 CVL-Database

The CVL-Database (CVL-DB), presented by Kleber et. al. in [44], was created within the scope of this work. It is a database for keyword spotting and for writer identification and retrieval. It

Dataset	# of Writers	# of images	equally distributed
CVl-Database	310	1604	yes
IAM Database	657	1539	no
ICDAR 2011	26	208	yes
ICDAR 2013	250	1000	yes
ICFHR 2012	100	400	yes
Glagolitic DB	7	361	no

Table 2.1: Key data of the different databases.

consists out of 1604 document images written by 310 different writers. 27 of these writers wrote the same 7 texts (6 English and 1 German) and 283 wrote 5 texts (4 English and 1 German). The 5 texts are a subset of the 7 texts. The writers are mainly students and employees of TU Wien, but also pupils of a public school. The texts which have been copied by the writers consist between 49 and 92 words. In total 101069 words have been written and tagged for using keyword spotting on this database. Furthermore, the authors of each document is stored, which allows an application of writer identification and writer retrieval methods. Additionally, the participants of the "ICDAR 2011 Writer Identification Contest" [55] and "ICFHR 2012 Competition on Writer Identification, Challenge 1: Latin/Greek Documents" [53] were asked to hand in their methods for an evaluation. Only 5 pages (from the Writer with Id 1) were sent to the participants. The evaluation results are presented together with the database in [44] for an objective comparison of current state of the art and newly developed methods. To obtain an equally distributed dataset, only the 5 pages which have been written by all writers are used. Two samples pages of the CVL-DB can be seen in Figure 2.1.



Mailüfterl is an Austrian nickname for the
first computer working ~~[illegible]~~ solely on transistor
on the European mainland. It was built in
1955 at the Vienna University of Technology by
Heinz Zemanek. The builder plays on a quote
on an operating computer: "If it is not
the rapid calculation speed of American models
that Whirlwind or Typhoon can achieve, it
will be enough for a Wiener Mailüfterl".
The full name is Binär dezimaler
Volltransistor - Rechenautomat.

Imagine a vast sheet of paper on which straight
Lines, Triangles, Squares, Pentagons, Hexagons, and other
figures, instead of remaining fixed in their places, move
freely about, on or in the surface, but without the
power of rising above or sinking below it, very much
like shadows - only hard and with luminous
edges - and you will then have a pretty correct
notion of my country and countrymen. Alas, a
few years ago, I should have said "my universe":
but now my mind has been opened to higher views
of things.

Figure 2.1: Two samples document images from the CVL-DB. Left Writer Id 1, Text 3 and right Writer Id 2, Text 1.

2.1.2 IAM Database

The IAM Handwriting Database (IAM-DB) is presented by Marti and Bunke in [59]. It is used to train and test handwritten text recognizers, keyword spotting and writer identification and

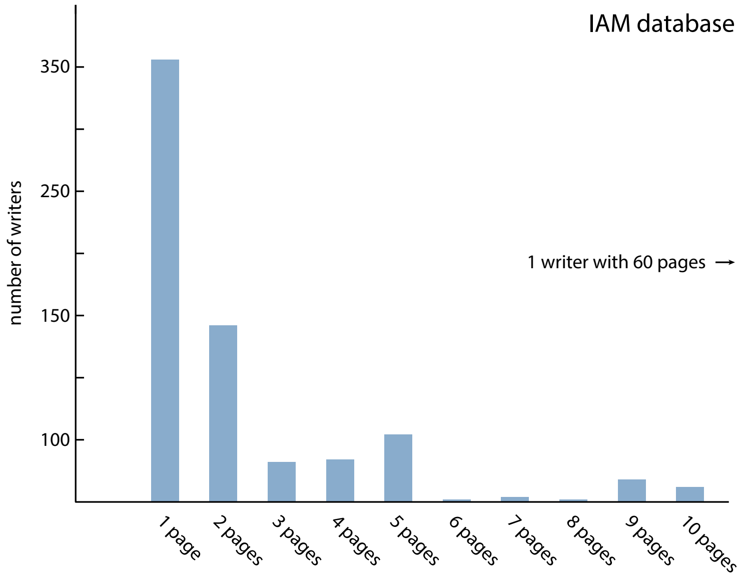
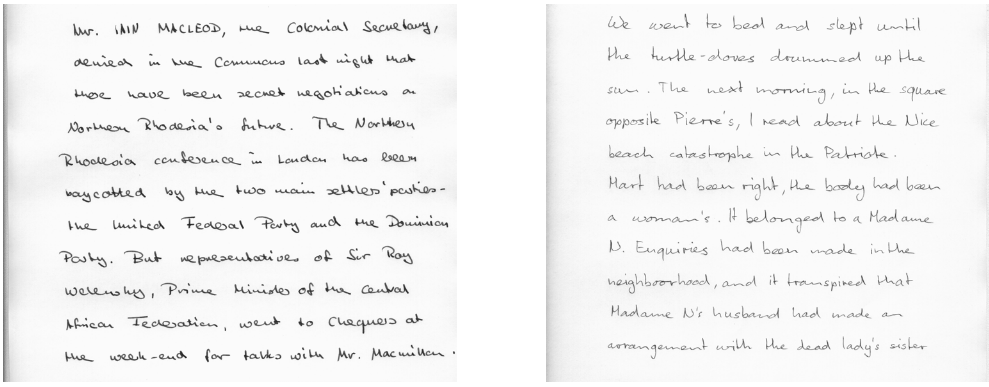


Figure 2.2: Distribution of the number of documents by writer in the IAM-DB.

verification. It consists out of 1539 document images, which have been written by 657 different writers. Different source texts have been used for copying and only some of them have been written by multiple writers. The distribution of the handwritten documents by each writer is not equal, one writer has contributed 60 pages and 350 writers have contributed only one text to the dataset. This distribution is illustrated in Figure 2.2. Since the writers with only one documents cannot directly be used for identification, different solutions are found in the literature how to use this database for evaluation. Some researches take only a subset of the dataset for their evaluation, some just cut the documents of the writers with one page in half, and some leave the database as it is and use these documents as "noise". Thus, the results on this database are not comparable. In this work the IAM-DB is used only for training purposes, but is also introduced since it is one of the most popular databases for writer identification and many of the state-of-the-art methods evaluated on this dataset. Two sample pages of the IAM-DB are shown in Figure 2.3.

2.1.3 ICDAR 2011 Writer Identification Contest

In 2011 the first writer identification contest within the International Conference on Document Analysis and Recognition (ICDAR) has been carried out. It has been organized by Louloudis et al. [55]. For the contest 26 writers copied eight pages, resulting in 208 document images. The texts are equally distributed in four languages (English, German, French and Greek). Additionally, a second dataset has been created by cropping out the first two lines of each document and thus making the writer identification task harder, since less text is present on these images. Participants have to submit their method knowing only a small set of the sample pages (which were not part of the evaluation dataset) and their methods have been evaluated according to cer-

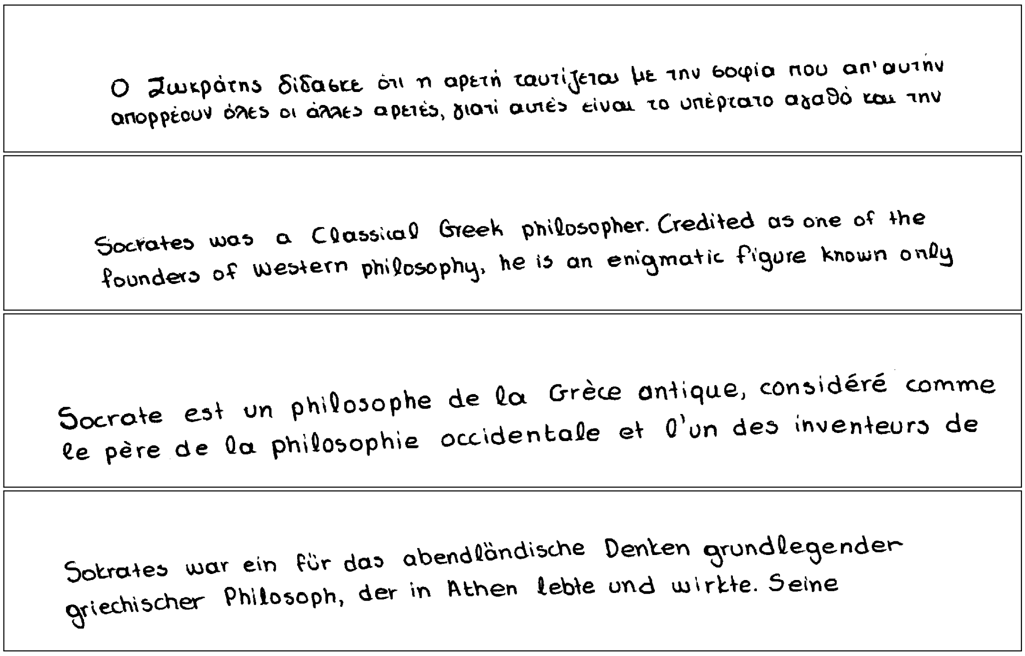


Mr. IAIN MACLEOD, the Colonial Secretary,
denied in the Commons last night that
there have been secret negotiations on
Northern Rhodesia's future. The Northern
Rhodesia conference in London has been
boycotted by the two main settlers' parties -
the United Federal Party and the Dominion
Party. But representatives of Sir Roy
Welensky, Prime Minister of the Central
African Federation, went to Chequers at
the week-end for talks with Mr. Macmillan.

We went to bed and slept until
the turtle-doves drummed up the
sun. The next morning, in the square
opposite Pierre's, I read about the Nice
beach catastrophe in the Patriote.
Hart had been right, the body had been
a woman's. It belonged to a Madame
N. Enquiries had been made in the
neighboorhood, and it transpired that
Madame N's husband had made an
arrangement with the dead lady's sister

Figure 2.3: Two samples document images from the IAM-DB. Left Writer Id 0 and right Writer Id 671.

tain criteria, which is described later. Eight different methods have been submitted by seven different institutions and their results are presented in [55]. Since the Greek language uses a different alphabet, the performance of all methods on these images is significantly worse than on the documents written in Latin alphabet. Figure 2.4 shows four samples of the cropped version of ICDAR 2011 dataset (Writer Id:1, Texts: 1-4).



Ο Σωκράτης δίδασκε ότι η αρετή ταυτίζεται με την σοφία που απ' αυτήν
απορρέουν όλες οι άλλες αρετές, γιατί αυτές είναι το υπέρτατο αγαθό και την

Socrates was a Classical Greek philosopher. Credited as one of the
founders of western philosophy, he is an enigmatic figure known only

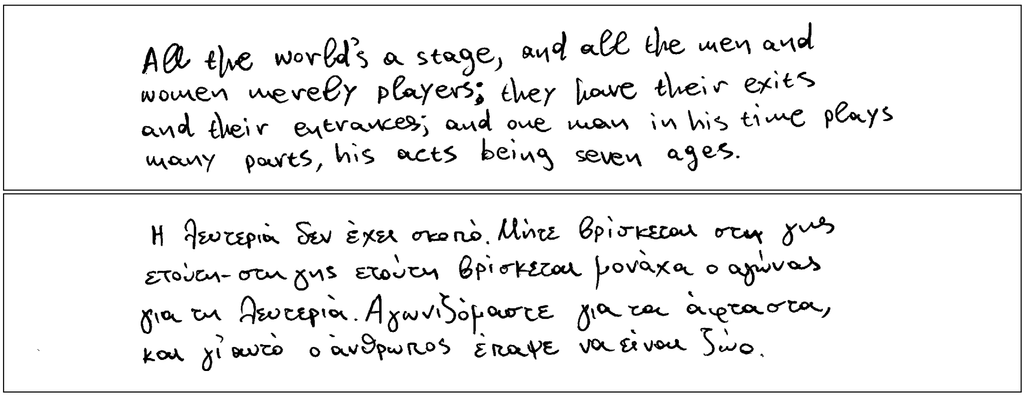
Socrate est un philosophe de la Grèce antique, considéré comme
le père de la philosophie occidentale et l'un des inventeurs de

Sokrates war ein für das abendländische Denken grundlegender
griechischer Philosoph, der in Athen lebte und wirkte. Seine

Figure 2.4: Four sample images from the cropped ICDAR 2011 Dataset. All samples are from Writer 1.

2.1.4 ICDAR 2013 - Competition on Writer Identification

In 2013 another competition on writer identification has been carried out. Again, it was organized by Louloudis et al. [54] and a new dataset was created for this contest. The benchmarking dataset consists of 1000 document images which have been written by 250 different writers. Two texts are in English and two texts are in Greek. 6 institutions handed in 12 different writer identification methods for this contest. Figure 2.5 shows two sample images of the ICDAR 2013 dataset.



All the world's a stage, and all the men and
women merely players; they have their exits
and their entrances; and one man in his time plays
many parts, his acts being seven ages.

Η Λευτεριά δεν έχει σκοπό. Μήτε βρίσκεται στη γης
ετούτη- στη γης ετούτη βρίσκεται μονάχα ο αγώνας
για τη Λευτεριά. Αγωνιζόμαστε για τα άφταστα,
και γι' αυτό ο άνθρωπος έπαψε να είναι ζώο.

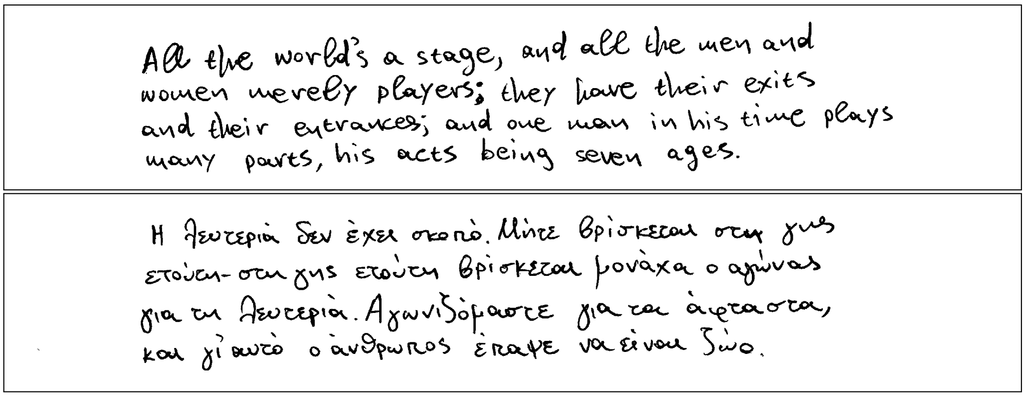
Figure 2.5: Two samples document images from the ICDAR 2013 competition. Text 1 and 2 from writer 19.

2.1.5 ICFHR 2012 Competition on Writer Identification

At the International Conference on Frontiers in Handwriting Recogntion (ICFHR) 2012 a competition on writer identification [53] took place and for this competition a new database was created with 400 images. 100 writers copied four texts in two languages (English and Greek). Each document image contains roughly 4 lines of text. 4 Institutes participated at this competition by submitting 7 different methods. Two sample images can be seen in Figure 2.6.

2.1.6 Glagolitic Database

The Glagolitic database is not publicly available due to copyright issues. It consists of 361 document images which were written by 7 writers. Table 2.2 shows the distribution of writers in the database. Writer 1 has the most documents in the dataset, whereas writer 6 has only 3 documents. The documents are written in Glagolitic, which is the oldest Slavic script [62], and originate from the 10th to the 14th centuries. The writings belong to five different manuscripts, three books have been imaged at Mt. Sinai and the remaining writings were photographed in libraries in Austria and Italy. The assignment of the writers to a specific documents has been done manually by philologists, whereby two manuscripts were written by several hands, while one manuscript (Codex Clozianus), which today is stored in two parts in different libraries, was written by one single scribe. Sample images of the database can be seen in Figure 2.7.



All the world's a stage, and all the men and women merely players; they have their exits and their entrances; and one man in his time plays many parts, his acts being seven ages.

Η λευτεριά δεν έχει σκοπό. Μήτε βρίσκεται στη γης ετούτη- στη γης ετούτη βρίσκεται μονάχα ο αγώνας για τη λευτεριά. Αγωνιζόμαστε για τα άφταστα, και γι' αυτό ο άνθρωπος έπαψε να είναι ζώο.

Figure 2.6: Two samples document images from the ICFHR 2012 competition. Text 1 and 3 from writer 36.

Writer Id	1	2	3	4	5	6	7
Psalterium Demetrii Sinaitici	207	22	40				
Euchologium Sinaiticum					7		
Missale Sinaiticum				7	51		
Codex Marianus						3	
Codex Clozianus							24
Total	207	22	40	7	58	3	24

Table 2.2: Distribution of the writers in the glagolitic database.

Since these documents contain a noisy background, the text blocks have to be found in these images which is done automatically by a text line detection method similar to the one proposed by Yosef et al. [2]. First, Local Projection Profiles (LPP) are applied to the image considered. Afterwards, the LPP image is filtered with a Gaussian column kernel and zero crossings of its first derivative are found. The final step is a non-extremum suppression to remove false positives. The zero crossings found are located at local minima and maxima, whereby the minima encode the text lines. Text lines which are smaller than a predefined threshold are filtered out and the bounding box containing the residual text lines is used for the cropping of the text region. Figure 2.8 shows two examples of the cropping procedure. The main text block of both manuscripts is successfully extracted. This approach is limited to rectangular text blocks. Thus, these regions can also contain noise from the background. Furthermore, for the removal of decorative elements (like initials) a more sophisticated layout analysis technique would be needed, but this is not within the scope of this work.

2.1.7 Evaluation Method

To evaluate writer identification methods usually, a nearest neighbor classification is used in literature [55] [53] [54] [36] [37] [20] [49] [30]. The identification rate is calculated with the leave-one-out strategy, so one document image is taken as reference image and the similarity to

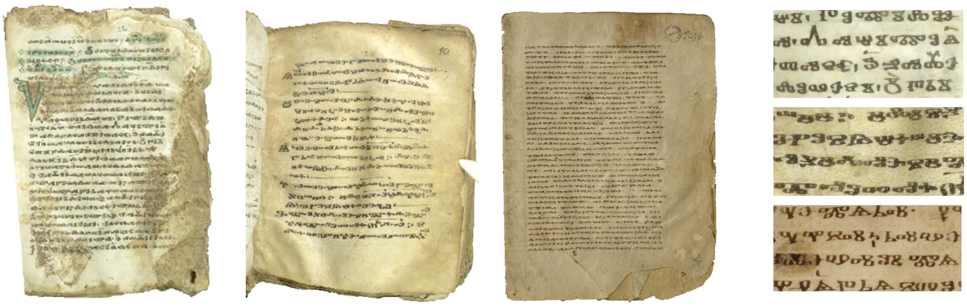


Figure 2.7: Examples taken from the dataset. (From left to right) Euchologium Sinaiticum folio 22 recto (Writer 3), Psalterium Demetrii Sinaitici folio 10 recto (Writer 2), Codex Clozianus folio 8 verso (Writer 7). Portions taken from the same images, from top to bottom: Writer 3, Writer 2, Writer 7.

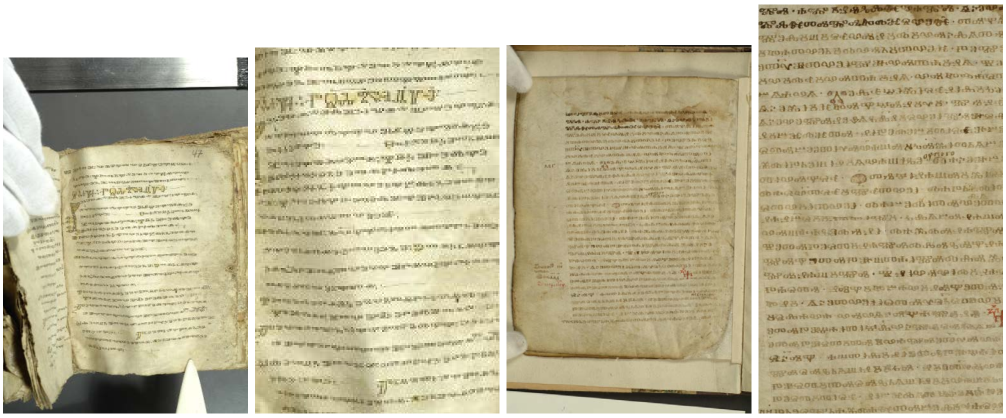


Figure 2.8: Input and result images of the cropping procedure. (From left to right) Psalterium Demetrii Sinaitici folio 47 recto (Writer 1). Corresponding result image. Codex Marianus folio 2 verso (Writer 6). Corresponding result image.

all other document images is calculated and sorted according to this value. Then it is checked if the first document originate from the same writer and if this is the case it is counted as a hit otherwise as a miss. Usually also not only the Top-1 document is examined but also some varying ranges. In the *ICDAR 2011 Writer Identification contest* two criteria have been defined, namely the *hard* and the *soft* criterion. According to [55] a hit in the soft Top-N criterion is counted “when at least one document image of the same writer is included in the N most similar document images” [55]. In contrast to this the *hard* criterion is defined stricter, because a correct hit is only achieved “when all N most similar document images are written by the same writer” [55]. In the *ICDAR 2011 Writer Identification contest* the values of N are set to 1, 2, 5, and 10 for the *soft* criterion and to 2, 5, 7 for the *hard* criterion. These values are varying from dataset to dataset, especially for the *hard* criterion, according to the property of the dataset. For example the *ICDAR 2013 - Competition on Writer Identification* used 2 and 3 as N for the hard criterion, since each writer has only 4 documents in the dataset and thus 3 is the maximal value for the hard criterion. Otherwise the hard criterion cannot be fulfilled.

The Top-1 criterion can be seen as the identification of the writer, since only one writer is returned by the method and it is either correct or wrong. The other criteria are not truly meaningful for human beings, since for the *hard* criterion all the documents and for the *soft* criterion only one document has to be from the same writer. If a dataset, like the CVL-DB, has 5 pages for each writer and three out of the four returned documents (one is taken as reference document) originate for the same author, it is not counted as correct hit for the *hard* criterion even though 75% of the returned documents are correct. The *soft* criterion does not distinguish if all documents or only one document is correct. Thus, these criteria do not reflect the retrieval of handwritings of one author in a dataset. For this reason for the CVL-DB a new criterion was introduced, namely the retrieval criterion [44]. It is calculated by examining "the first N document images and calculate how many of them are from the same writer as the reference document. Using all document images in the database once, we calculate the percentage of how many document images have been found correctly" [44]. Again, this criterion is calculated on varying number of N.

One lack of these standard evaluation methods is that only a nearest neighbor classification is used. This has mainly one reason: if machine learning methods are applied to this task a subset of the database has to be defined for learning. Since one document image is not representative for one writer it has to consist of multiple pages per writer and thus making the evaluation dataset smaller. Additionally, writer identification methods are more focusing on the features itself and not on machine learning. Furthermore, for real world applications it can be expected that the database of known writers is changing frequently and thus online learning methods are recommended.

2.2 Character Based Writer Identification Methods

Marti et al. propose in [60] twelve features which mainly correspond to visible characteristics of the handwriting. The first three features used are based on the three different writing zones which are illustrated in Figure 2.9. The height of the different zones are taken as features, whereas the *middle zone* can be considered as *x-height* and the upper and lower zone represent the height of the ascenders respectively descenders. These bounding lines are determined by generating a horizontal projection of the text line. An ideal histogram is matched against this profile and thus the 4 lines can be fitted. Furthermore, to avoid the variability of the writing's height, the ratios of the three different zones form the next three features. Next, the median of the length of white pixel runs which is calculated on the row with the most white-black transitions is taken as feature. Again, to respect different sizes of the handwriting, the ratio between this median and the middle zone is also used for identification. Additionally, the slant of the writing is also taken into account. The slant is determined by generating an angle histogram, which is built by approximating the contour pixels to straight lines. The mean value μ and the standard deviation σ are used as features. The last two features are formed by using fractal geometry. Using these 12 features they compare the performance using a *k-NN* and a feed forward neural network and also compare the performance of the different feature sets (relative to absolute features). On a subset of the IAM-DB with 100 pages an identification rate of 87.8% is achieved with the k-NN classifier (using only the relative features) and 90.7% using the neural network with all features. These results are achieved only when using one text lines for identification. When combining the text lines of one page using a voting strategy, then all writers have been identified.

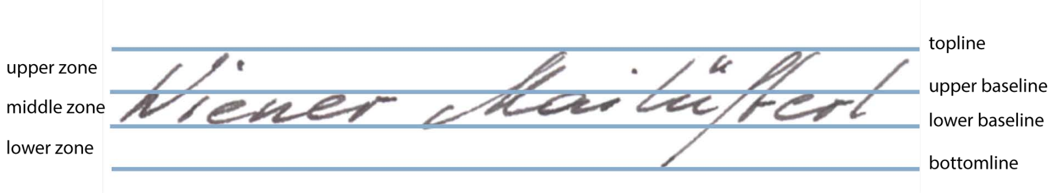


Figure 2.9: The three different writing zones and bounding lines according to Marti et al. [60].

Bulacu et al. [9] propose to use edge-based direction features for writer identification. They also use the directions of the contour by quantizing the orientation which is determined by analyzing the neighborhood of the edge pixels. Figure 2.10 shows the distribution of the edge-directions of two different writers in a polar diagram. The predominant direction corresponds to the slant of the writer. The differentiation of the edge-directions is used as feature vector since it leads to a significant performance improvement. They also introduce a new feature called the edge-hinge distribution. Instead of using only the direction of one edge, the angle between two edges is taken into account. Figure 2.11 shows the creation of this feature. (a) shows the generation of the directions, which are then used to generate a 2-D-histogram. Figure 2.11 (b) shows histograms of two different writers. On the Firemaker data set with 250 writers, two pages each, a performance of 75% is achieved. With a multiscale approach van der Maaten and Postma [86] raised this result by 11%.

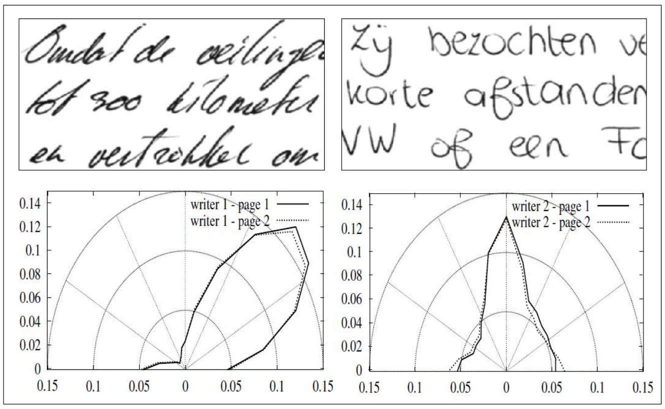
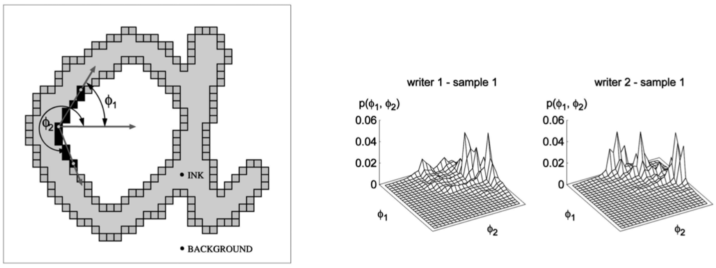


Figure 2.10: Two handwriting samples from two different writers and their according edge-direction distribution. The main predominant direction corresponds to the slant of the writing. (Taken from [9])

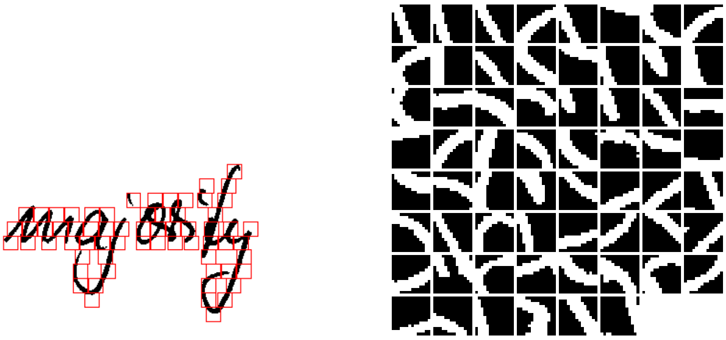
Siddiqi and Vicent [80] propose to use an approach for highlighting the frequent details in a handwriting by exploiting the redundancy of individual handwritings. To achieve this, they



(a) (b)

Figure 2.11: (a) Schematic description for the edge-hinge feature. (b) Two 2-D-histograms of the distribution of the edge-hinges. (Taken from [8])

extract small sub-images on the text with a fixed size. Figure 2.12 (a) shows a handwritten word with the corresponding sub-images which are extracted. For training, those sub-images are generated over the complete trainings set and then they are clustered. Every cluster which contains more than five images is stored as feature. Such a codebook can be seen in Figure 2.12 (b), these features are then used for writer identification. For each new page the sub-images are generated and clustered again. The representative feature of the particular writer is then compared with the trained ones and with the probabilities of these comparisons the similarity measurement is calculated. The method is evaluated on a subset of 50 writers of the IAM-DB. One page is used for training and one page is used for evaluations. The identification rate is 94%.

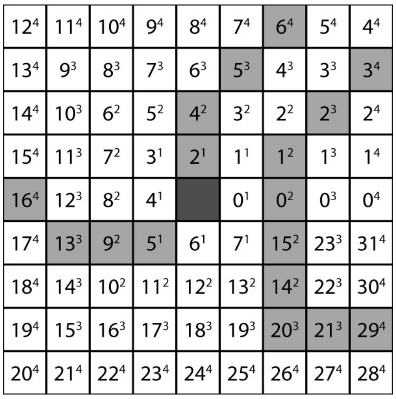


(a) Sub-images extracted from the writing.

(b) Features used for writer identification.

Figure 2.12: Proposed method from Siddiqi and Vicent [80] (Pictures also taken from [80])

In 2009 Li and Ding [49] introduce the Grid Microstructure Features (GMSF) for writer identification. It is described in this work in more detail, because the improved method won the *ICDAR 2011 Writer Identification Contest*. They use the edge image of the handwriting to generate their features on every edge pixel by using a sliding window along the contour. On every center pixel on such a sliding window a grid is laid. Figure 2.13 shows such a grid on an edge pixel with a layer size L of 4. The other pixels in the window are then numbered counterclockwise and with the layer level as superscript, resulting in an i^m representation for each pixel. For feature generation, only the edge pixels in the sliding windows are used. Then pixel pairs are found according to three different criteria: [49]



12^4	11^4	10^4	9^4	8^4	7^4	6^4	5^4	4^4
13^4	9^3	8^3	7^3	6^3	5^3	4^3	3^3	3^4
14^4	10^3	6^2	5^2	4^2	3^2	2^2	2^3	2^4
15^4	11^3	7^2	3^1	2^1	1^1	1^2	1^3	1^4
16^4	12^3	8^2	4^1	■	0^1	0^2	0^3	0^4
17^4	13^3	9^2	5^1	6^1	7^1	15^2	23^3	31^4
18^4	14^3	10^2	11^2	12^2	13^2	14^2	22^3	30^4
19^4	15^3	16^3	17^3	18^3	19^3	20^3	21^3	29^4
20^4	21^4	22^4	23^4	24^4	25^4	26^4	27^4	28^4

Figure 2.13: The grid which is laid on all edge pixels and which is used to generate the GMSF-features.

$$C_1 : \begin{cases} i \leq m = l \leq L, \ i \leq j \\ i_m \ and \ j_l \ are \ the \ edge \ pixels \\ k_m \ is \ not \ the \ edge \ pixel, \ i \leq k \leq j \end{cases}$$

$$C_2 : \begin{cases} i \leq m = l - 1 \leq L, \ i \leq j \\ i_m \ and \ j_l \ are \ the \ connected \ edge \ pixels \\ i_m \ is \ the \ nearest \ to \ j_m \end{cases}$$

$$C_3 : \begin{cases} i \leq m = l - 2 \leq L, \ i \leq j \\ i_m \ and \ j_l \ are \ the \ connected \ edge \ pixels \\ i_m \ is \ the \ nearest \ to \ j_m \end{cases}$$

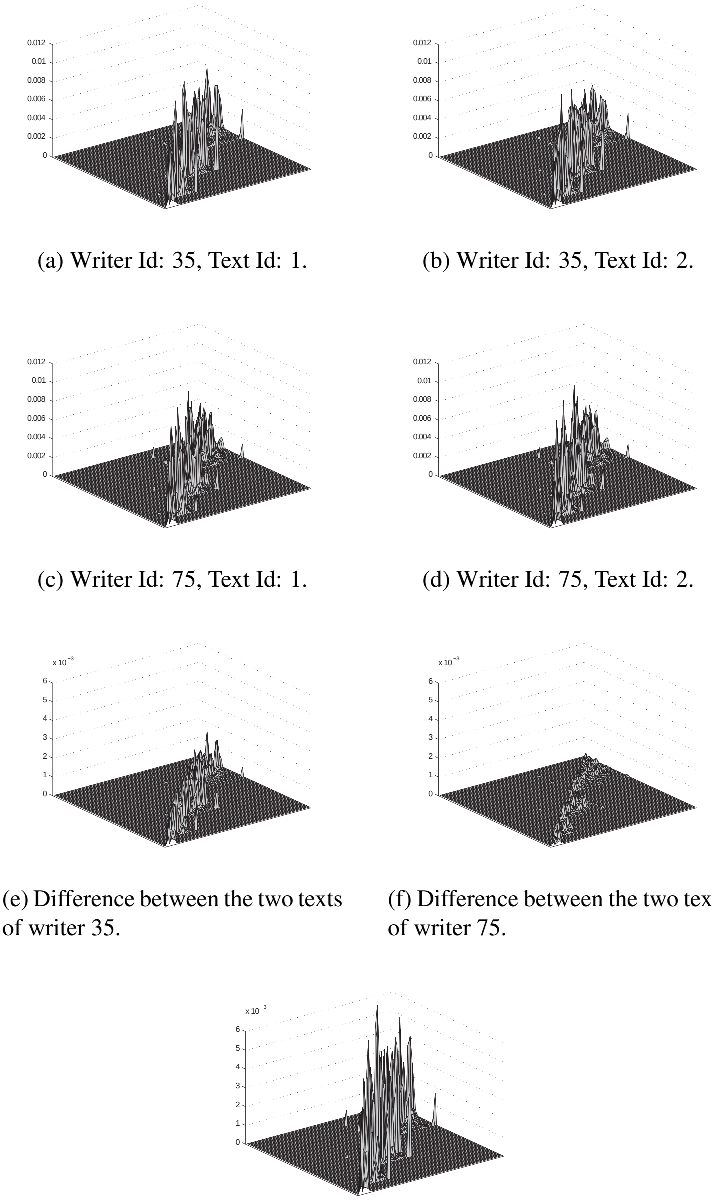
For the sample image in Figure 2.13, criterion C_1 finds amongst others following pixel pairs: $< 2^1, 5^1 >, < 1^2, 15^2 >, < 4^2, 9^2 >, < 5^3, 13^3 >, < 6^4, 16^4 >, < 3^4, 29^4 >, < 2^3, 21^3 > \ldots$. Criterion C_2 returns amongst others following pairs: $< 2^1, 4^2 >, < 9^2, 5^1 >, < 14^2, 20^3 >, < 2^3, 3^4 >, < 5^3, 6^4 >, < 13^3, 16^4 >, < 21^3, 29^4 >$,…and criterion C_3 finds following pairs: $< 2^1, 5^3 >, < 5^1, 13^3 >, < 1^2, 3^4 >, < 4^2, 6^4 >, < 9^2, 16^4 >$ and $< 20^3, 29^4 >$. The occurrences of all pixel pairs are stored in one 2-D-Histogram, which is divided by the sum of all pixel pairs to get the 2-D probability density distribution, which is the GMSF. Two sample GMSF of two different writers are shown in Figure 2.14 (a)-(d) of two different texts on the ICDAR 2011 dataset, also the difference between the two texts of the writers (Figure 2.14 (e) and (f)) and the difference of the GMSF of text 1 of both writers are shown (Figure 2.14 (g)). It can be seen in Figure 2.14 (e) and (f) that the intra writer distance is very low and all the probability values are below 0.0018 whereas the difference of the GMSF of both writers differ clearly. To measure the similarity between two GMSF Li and Ding propose to use an improved weighted χ^2-Distance [49]. To weight the difference of each component the standard deviation of all reference handwritings is used. This method is evaluated on a Chinese dataset with 240 writers and achieves an identification rate of 95.0%.

In [58] this method was improved by making it insensitive against pen-width variations. Figure 2.15 illustrated the problem of the pen-width sensitivity. Two strokes are in this image, one consisting of the gray and red pixels and the other consisting of the gray and blue pixels. Obviously, both strokes are the same except for the width of the pen and also two different GMSF are generated. To avoid this sensitivity handwriting from 25 individuals are analyzed and the pen-width synthetically changed by using morphological operations. They distinguish between three different types of pens: thin (0.3mm, 3-4 pixels), median (0.5mm, 5-6 pixels), and wide (1.0mm, 7-8 pixels). They calculate the average GMSF for each writer by calculating the mean GMSF of the three different widths. The mean GMSF of all sample writers is then incorporated into the distance measurement. With this approach they won the *ICDAR 2011 Writer Identification Contest* by achieving an identification rate of 99.5% on the full dataset and 90.9% on the cropped dataset.

Jain and Doermann [35] propose K-Adjacent Segments (KAS) for offline writer identification. KAS are introduced by Ferrari et al. [23] and are local contour features which describe the relationship of K neighboring edges. They extract the KAS on the lines of the contour of the writing using a line fitting algorithm. They evaluated the K equals 2, 3, or 4, with 3 being the best. A bag of feature model is used to compare the handwriting of different document images. Figure 2.16 shows the 20 most popular 3-Adjacent Segments of their IAM-DB trainings set. Each segment of a new document image is compared to the ones in the codebook and an occurrence histogram is built up which is then used for identifying the writer. They achieve an identification rate of 93.3% on 300 writers from the IAM-DB.

A combination of run-length features and the already described edge-hinge features is porposed by Djeddi et al. [20]. This method won the "ICFHR 2012 Competition on Writer Identification". For the run-length feature they scan the document image in four directions (horizontal, vertical, left-diagonal, and right-diagonal) and generate gray level run-length matrices and the histogram of run-lengths which is normalized and interpreted as probability density function. Black and white run-lengths are used. These features "give information on the average width of the letters, the density of writing, the structure of the letters, the average size of the letters, the ink width, the characters position, the regions enclosed inside the letters and also the empty spaces between letters and words, the regularity and irregularity of handwriting and finally the slope in handwriting" [20]. In the ICFHR 2012 competition they achieved an identification rate of 94.5% using the Manhattan distance for a nearest neighbor classification.

An alphabet of contour gradient descriptors for writer identification is proposed by Jain and Doermann [36]. Again, this approach is described in more detail since it is the winner of the *ICDAR 2013 - Competition on Writer Identification*. Three different segmentation strategies to extract the characters of the handwriting are evaluated. First, they simple use Connected Components (CC) for segmentation, which is nearly optimal if the characters are not touching. Second, they use vertical cut which allow for splitting large connected components into repeatable



(a) Writer Id: 35, Text Id: 1.

(b) Writer Id: 35, Text Id: 2.

(c) Writer Id: 75, Text Id: 1.

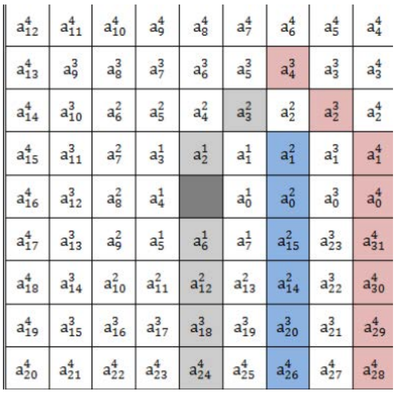
(d) Writer Id: 75, Text Id: 2.

(e) Difference between the two texts of writer 35.

(f) Difference between the two texts of writer 75.

(g) Difference between writer 35 and 75 (Text Id: 1).

Figure 2.14: (a)-(d) Four sample 2-D probability density distribution (GMSF) generated from two different texts of two different writers. (e-f) Difference between the two GMSF of Writer 35, respectively Writer 75. (g) Difference between the two GMSF of Text 1 of both writers.



a^4_{12}	a^4_{11}	a^4_{10}	a^4_9	a^4_8	a^4_7	a^4_6	a^4_5	a^4_4
a^4_{13}	a^3_9	a^3_8	a^3_7	a^3_6	a^3_5	a^3_4	a^3_3	a^4_3
a^4_{14}	a^3_{10}	a^2_6	a^2_5	a^2_4	a^2_3	a^2_2	a^3_2	a^4_2
a^4_{15}	a^3_{11}	a^2_7	a^1_3	a^1_2	a^1_1	a^2_1	a^3_1	a^4_1
a^4_{16}	a^3_{12}	a^2_8	a^1_4		a^1_0	a^2_0	a^3_0	a^4_0
a^4_{17}	a^3_{13}	a^2_9	a^1_5	a^1_6	a^1_7	a^2_{15}	a^3_{23}	a^4_{31}
a^4_{18}	a^3_{14}	a^2_{10}	a^2_{11}	a^2_{12}	a^2_{13}	a^2_{14}	a^3_{22}	a^4_{30}
a^4_{19}	a^3_{15}	a^3_{16}	a^3_{17}	a^3_{18}	a^3_{19}	a^3_{20}	a^3_{21}	a^4_{29}
a^4_{20}	a^4_{21}	a^4_{22}	a^4_{23}	a^4_{24}	a^4_{25}	a^4_{26}	a^4_{27}	a^4_{28}

Figure 2.15: Illustration of the pen-width sensitivity of the GMSF. The stroke with the gray and red pixels generate a different feature as the gray-blue one, even though it is only wider. (Figure taken from [58])

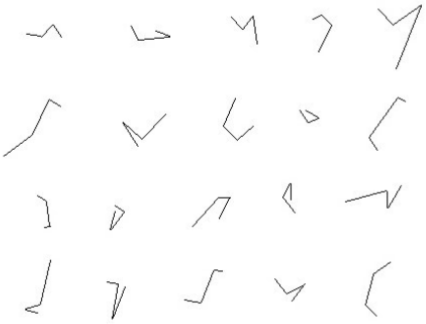


Figure 2.16: Some samples of the codebook showing the 20 most popular 3-Adjacent-Segments (Figure taken from [35]).

pieces. Contour and line pixels are assigned an energy value according to a specific function and the methods tries to split the word at places with the minimum energy, which means that loops and multiple lines are less likely to be divided. The last method they evaluated for segmenting the characters are seam cuts, which use a heuristic path planning. The advantage of seam cuts is, that it can make cuts around curved strokes or slanted characters. Again the energy of every pixel is generated and a path is found with the minimal energy. Figure 2.17 shows two iterations of the vertical cut method and an example of the seam cut method. The red areas in the vertical cut image indicate areas where a cut is not allowed to prevent the method from continuously selecting small components.

A new binary feature, which is calculated on the cut characters is introduced, namely the contour gradient feature. It is able to “capture the shape and the curvature of a character-like segment” [36]. The contour of the character is extracted and for each point on the contour the gradient is calculated by combining the slope of two contour segments of size P at this point. The size of P is set to the median size of the writing height. Figure 2.18 shows the calculation of the contour gradient and the generation of the feature. First, the contour of a binarized character

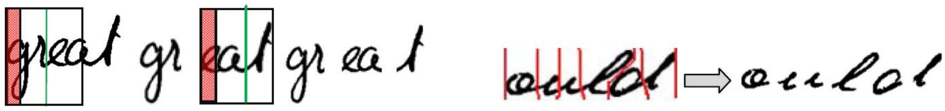


Figure 2.17: Left: Two iterations of the vertical cut methods. Right: Example of the seam cut method (both images taken from [36]).

is calculated and then for each point the gradient is calculating with the slope of the line segments of the point. These gradients are then put into a SIFT-like descriptor by placing a grid over the character and the resulting feature is then normalized by the sum of the total gradient energy. By clustering the character-like segments they generate a global codebook with exemplar letters for each writer. Samples of such a codebook can be seen in Figure 2.19. For writer identification they search for each entry in the codebook of one sample the closest entry in the other codebook. The sum, normalized by the number of cluster centers, is used as distance. This distance can be seen as “the minimum distance required to transform alphabet A into alphabet B” [36].

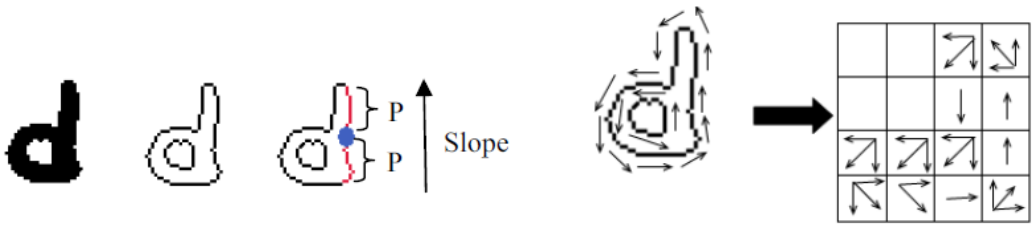


Figure 2.18: Left: Generation of the contour gradient on a character. The contour of the binarized character is extracted and at each point (blue) the gradient is determined by the slopes of the contour segments (red) Right: Building the gradient contour feature out of the contours. (Taken from [36]).

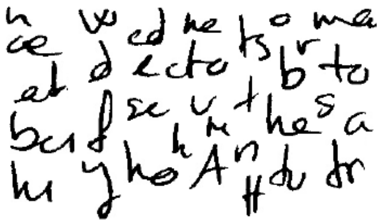


Figure 2.19: Samples of the codebook generated (Taken from [36]).

As already stated, they won the *ICDAR 2013 - Competition on Writer Identification* by achieving an identification rate of 95.1%. For this result they use the seam cut method, which performs slightly better than the vertical cut method. They also evaluated their approach on the IAM-DB and receive an accuracy of 96.5% using vertical cuts, 95.4% using seam cuts and 91.8% using CC as segmentation.

2.3 Texture Based Writer Identification Methods

A writer identification method using wavelet domain local binary patterns is proposed by Du et al. [21]. The first step of their approach is the normalization of the handwriting by using the method proposed by Peake and Tan [69]. This preprocessing step removes the spacing between the text lines and characters. Figure 2.20 illustrates the workflow of the feature generation. A wavelet decomposition is applied on the normalized image of the handwriting. For each of the wavelet subbands the Local Binary Patterns (LBP) are calculated. The LBP consider the difference between the gray value of a pixel and its neighbor sets and thus giving a representation of the corresponding texture. The feature vector of the handwriting is generated by concatenation of the results of each subband. The χ^2 distance is then used to calculate the distance between two features. They evaluate their approach on a dataset of 100 document images written by 50 writers in Chinese. Using a nearest neighbor classification an identification rate of ~68% is achieved.

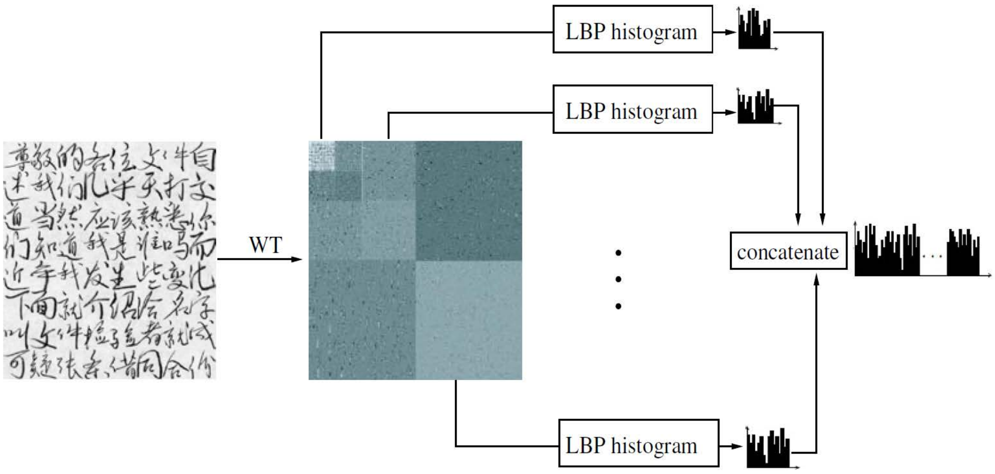


Figure 2.20: Workflow of the feature generation of Du et al. [21] (Figure taken from [21]).

Hiremath et al. [30] also propose an approach which uses the wavelet decomposition. They apply the 2D Discrete Wavelet Transform and the image of the handwriting and calculate the co-occurrence histograms for each pair of subbands. For each histogram the normalized cumulative histogram is formed and the features are generated on this histogram. They use the slope of the regression line, mean, and mean deviation as features. They evaluate their approach on a dataset of 30 writers in which each writer wrote 50 samples. 25 of them are used as training and the rest for testing. When using the complete dataset, an identification rate of 87.94% is achieved using a nearest neighbor classification.

Also the use of local features can be seen as texture based writer identification. Christlein et al. [11] propose to use Gaussian Mixture Model (GMM) supervectors for writer identification, which is also used for speaker verification [10]. They employ RootSIFT features, which is a variant of the Scale Invariant Feature Transform (SIFT) with an additional normalization using the square root (Hellinger) kernel. An independent trainings set is used to form an Universal

Background Model (UBM), by estimating a GMM on the features. For each document image examined, this UBM is adapted. The new parameters of the GMM are then stacked into one vector, the so called supervector, which is then normalized. Using the cosine distance, a nearest neighbor classifier can be applied and the method is evaluated on several databases. On the CVL-DB an identification rate of 99.2% and on the ICDAR 2013 dataset a rate of 97.1% are achieved.

Dhandra and Vijayglaxmi [15] use Gray Level Co-occurrences Matrices (GLCM), which consider the spatial relationship of pixels, for writer identification. They chose 5 distances and four directions to create twenty GLCM statistics and then use the energy, contrast, homogeneity, and correlation as features. They evaluated their approach on datasets with 100 writers with 3 different scripts (Roman, Kannada and Devanagari). The identification rate on Roman is 82.75%, for Kannada also 82.75% and for Devangari 85.25%. When combining the three scripts into one dataset, the identification rate is still 82.19%.

Jain and Doerman [37] propose the combination of three different approaches for writer identification. Apart from their own two methods, [35] and [36], which have been described in Section 2.2 they use SURF [3] with the application of the Fisher Vector. They use a weighted linear combination as distance measurement, where the weights have been found empirically. The evaluation is carried out on different datasets achieving an identification rate of 94.7% on the IAM-DB and 98.3% on the CVL-DB.

2.4 Comparison of Writer Identification Methods

Table 2.3 gives an overview of the results of the different methods on different databases. It can be seen in this table, that often only a subset of the dataset is used for evaluation and thus even results on the same dataset cannot be compared to each other. The performance of the different approaches range from 68 to 99.2%. The more text is available on the document image, like for example on the CVL-DB, the better the results. This can be seen especially on the results of the *ICDAR 2011 Writer Identification Contest*, where a cropped dataset is available in which only 2 lines are present.

Multilingual datasets tend to have worse results than the ones which contain only one language, especially the results of the *ICDAR 2013 Competition of Writer Identification* are showing this influence where the winner (Jain and Doerman [36]) has an identification rate of 95.6% on the Greek respectively 94.6% on the English pages of the dataset but only 19.6% in the Top-2 criterion. For the Top-2 criterion, one page of the other language needs to be found. Of the methods presented in the work, only Djeddi et al. [20] handle the difference between the two alphabets very good by achieving an identification rate of 63.2% in the Top-2 criterion.

2.5 Summary

In this chapter a short overview of datasets and of the state of the art on writer identification has been given. Since nearly all methods are working with a nearest neighbor classification, they can also be used for writer retrieval. Also some datasets, which are either common in the community and freely available, or used in this work, were presented. The IAM-DB is

Author	Database	Writers	Pages	Language	Top 1
Marti et al. [60]	IAM-DB	20	100	English	90.7%
Bulacu et al. [9]	Firemaker	250	500	Dutch	75%
van der Maaten et al. [86]	Firemaker	150	300	Dutch	86%
Siddiqi et al. [80]	IAM-DB	50	100	English	94%
Li et al. [49]	HIT-MW	240	480	Chinese	95%
Xu et al. [58]	ICDAR 2011	26	208	English, Greek, German, French	99.5%
	ICDAR 2011 cropped	26	208	English, Greek, German, French	79.8%
	CVL-DB	310	1604	English (4 texts), German (1 text)	97.7%
Jain et al. [35]	IAM-DB	300	600	English	93.3%
	MADCAT	302	3020	Arabic	~78% %
	CVL-DB	310	1604	English (4 texts), German (1 text)	97.9%
	ICDAR 2013	250	1000	English, Greek	85.5%
Djeddi et al. [20]	IFN/ENIT	275	1374	Arabic	93.5%
	ICFHR 2012	100	400	English, Greek	94.5%
	ICDAR 2013	250	1000	English, Greek	93.4%
	CVL-DB	310	1604	English (4 texts), German (1 text)	97.6%
Jain et al. [36]	ICDAR 2013	250	1000	English, Greek	95.1%
	IAM-DB	301	602	English	96.5%
	ICFHR 2012	100	200	English	98%
	ICFHR 2012	100	200	Greek	97.5%
	MADCAT	316	632	Arabic	87.5%
Du et al. [21]	-	50	100	Chinese	68%
Hiremath et al. [30]	-	30	750	English	87.94%
	-	30	750	Kannada	91.45%
Christlein et al. [11]	CVL-DB	310	1604	English (4 texts), German (1 text)	99.2%
	ICDAR 2013	250	1000	English, Greek	97.1%
Jain et al. [37]	IAM-DB	657	1314	English	94.7%
	CVL-DB	310	1604	English (4 texts), German (1 text)	98.3%
Dhandra et al. [15]	-	100	400	English	82.75%
	-	100	400	Kannada	82.75%
	-	100	400	Devanagari	85.25%
	-	100	1200	English, Kannada, Devanagari	82.19%

Table 2.3: Overview of different state-of-the art writer identification methods and their performance. Since the methods are applied on different datasets respectively subsets of the databases some results cannot be compared which each other.

the most often used dataset, but it lacks an equal distribution of writers. Thus, the CVL-DB has been proposed, which has 310 writers and 1604 pages. The evaluation sets of the ICDAR 2011, ICDAR 2013, and ICFHR 2012 writer identification competition were also described. Additionally, the Glagolitic DB was presented, which is used in this work to show that the methods proposed also work on historic documents.

The state-of-the-art methods have been divided into two groups: the ones that calculate their features directly on the characters and the ones that assume the handwriting as texture. The overview about the results show that currently the writer identification methods are performing well (>90%) on all datasets but lack when less text is available (like on the ICDAR 2011 cropped dataset). Additionally, if the dataset consists of multiple languages (or even alphabets) the performance is also dropping.

CHAPTER 3

Concepts Used

This chapter gives an overview of the concepts which are used for the methods proposed for writer identification and retrieval. First, the BOW approach is presented, including the BOW for natural language processing and the features used. Then the Fisher Vector is introduced, which is used for the second writer identification method. The last concept which is presented is the Convolutional Neural Network (CNN). At the end a short summary is given.

3.1 Bag of Words

The BOW approach, sometimes also called bag of features, was introduced in the field of natural language processing and text classification [40], which are described first in this section. Afterwards SIFT are presented, which is then used with the visual BOW approach.

3.1.1 BOW in Natural Language Processing

The BOW model in natural language processing is used to determine the content of a text. This is done by analyzing the words which occur in the text and then compare the occurrences of words with pretrained texts to calculate the similarity. This model assumes that the text is an unordered collection of words, regardless of the grammar or word order. Articles, filler words, pronouns, determiners, and also some verbs are filtered out in a preprocessing step since they do not contain any information about the context of a text. To analyze a new text, dictionaries have to be defined first which represent the classes. This is done by taking texts which deal with a specific topic and count the occurrences of words. A straight forward idea would be to just take a look at the words with the highest occurrence and assign the text to the class in the database which has also the highest occurrence of these words. The drawback of this idea is that some text classes do not differ very much and thus the texts are easily misclassified. To overcome this problem the BOW approach builds up an occurrence histogram of all words which are stored in the dictionary and classify the text using the histogram similarity or machine learning methods on these histograms. To show an example of text classification, the following text, which contains the first three sentences of the abstract of Fiel and Sablatnig [25], should be classified:

> *In this paper a method for writer identification and writer retrieval is presented. Writer identification is the task of identifying the writer of a document out of a database of known writers. In contrast to identification, writer retrieval is the task of finding documents in a database according to the similarity of handwritings. The approach presented in this paper uses local features for this task. First a vocabulary is calculated by clustering features using a Gaussian Mixture Model and applying the Fisher kernel.*

Then we also have to define the dictionaries for various types of texts. These dictionaries can be defined by analyzing texts with a known topic but for simplicity reasons they are defined manually here. So the dictionaries D of three class are defined as follows:

$D_{writer\ identification}$ = *{”writer”, “identification”, “handwriting”, “author” }*
$D_{layout\ analysis}$ = *{ “document”, “layout”, “analysis”, “structure”}*
$D_{document\ analysis}$ = *{”document”, “analysis”, “structure”, “writer” }*

Again, for simplicity reasons, we assume that each word also represents the plural and also the adjective form of the word. Some words can also occur in two dictionaries. The next step is to eliminate the articles, filler words, pronouns, determiners, and some verbs like “are”, “use”, and “present”. Thus, the text looks like this:

> *paper method writer identification writer retrieval. Writer identification task identifying writer document database writers. identification, writer retrieval finding documents database similarity handwritings. approach paper local features task. vocabulary clustering features Gaussian Mixture Model Fisher kernel.*

Now the occurrences of the words in the dictionary are counted. Words which are not in the dictionary are not counted. Another possibility would be to identify the most similar word in the dictionary, but for this showcase this is not done because a similarity measurement for words would have to be defined. Figure 3.1 shows the normalized histogram of the occurrences of the dictionary words in the text. It can be seen that words of the *writer identification* dictionary appear most often, followed by the *document analysis class*. This is mainly because both contain the word “writer” and thus it has to be marked as not so important for the specific class. When analyzing a new text, the occurrence histogram is also generated and with a histogram similarity measurement the distance between those can be calculated and can be taken for e.g. a nearest neighbor classification. Especially machine learning methods have shown [40] to give a good classification of the text, since such methods learn the importance of single words in the dictionary.

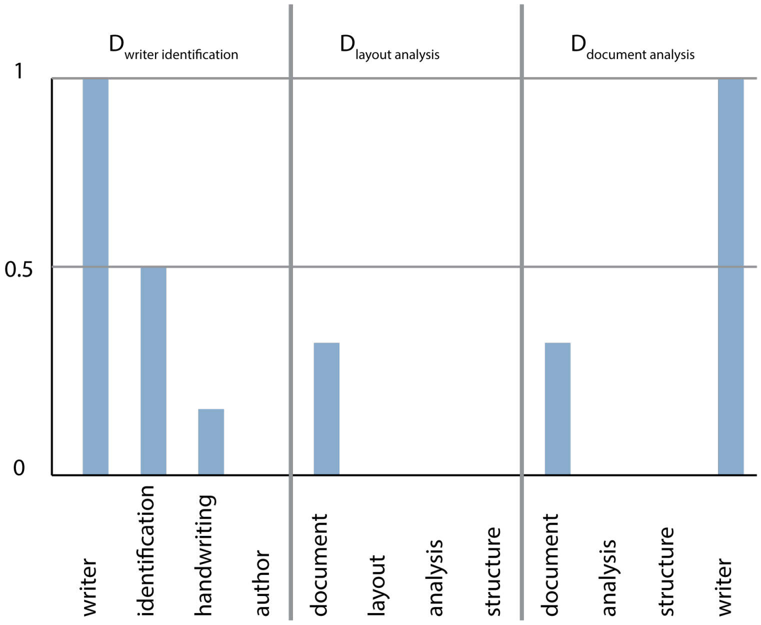


Figure 3.1: The normalized histogram of the occurrences of the words in the dictionaries in the sample text.

3.1.2 Scale Invariant Feature Transform

The SIFT was introduced by Lowe [56] and improved in [57]. It was introduced in the field of object recognition, but has also shown good results for texture analysis [63]. The features are invariant to rotation, scale, and translation. To generate the set of image features four stages of computation are used [57]:

- Scale-space extrema detection
- Keypoint localization
- Orientation assignment
- Keypoint descriptor

In the first step possible candidates for keypoints are identified over all scales and image locations, whereas in the second the keypoints are selected according to a stability criterion. Afterwards the main orientation of the keypoint is determined to achieve the rotation invariance. The last step is the transformation of the keypoint and its surrounding area into a fixed feature representation.

The scale space allows for extraction of structures in the image at different scales. With increasing scale, the fine details in the image are successively suppressed. This is achieved by increasing the σ fpr the Gauss filter used. According to Lindeberg “the Gaussian kernel and its derivatives are singled out as the only possible smoothing kernels” [50] and also that “the output from the scale space representation can be used for a variety of early visual tasks; operations like feature detection, feature classification and shape computations can be expressed directly in terms of (possibly non-linear) combinations of Gaussian derivatives at multiple scales" [50]. For SIFT features, Difference Of Gaussians (DOG) are used, since the provide a close approximation to the scale-normalized Laplacian of Gaussian. This scale space is constructed on successive down sampled input image. These down sampling steps are called octave. Figure 3.2 shows the construction of the scale space at different octaves when generating the SIFT features. First, the images are consecutively convolved with Gaussians for each octave. Adjacent images are then subtracted to gain the DOG image. For each octave the image is down sampling by a factor of 2. The detection of the local extrema in the scale space is done by comparing each sample point to its neighbors. In total these are 26 comparisons, since the sample point is compared with its 8 neighbors in the current image and also with the 9 neighbors in the scale above and below. Because of the pre-smoothing of the image before extrema detection, the highest spatial frequencies are discarded. Thus Lowe [57] propose to double the size of the input image using linear interpolation before building the first level of the pyramid.

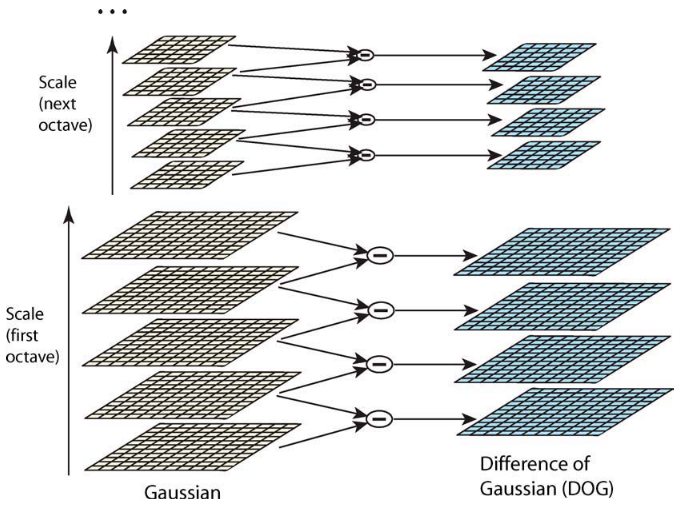


Figure 3.2: Construction of the Scale Space at different octaves. For each octave the image is convolved using a Gaussian Filter and consecutive images are subtracted to produce a DOG image. For each scale the Gaussian image is down sampled by the factor of 2 (Figure taken from [57]).

In the next step candidate pixels for keypoints have to be examined if they have poor contrast or are poorly localized along an edge. These candidate pixels are skipped for further processing. Brown and Lowe [7] developed a method to determine the interpolated location of the maximum by fitting a 3D quadratic function to the local sample points. The value of this function can be used to reject unstable keypoints. To eliminate edge responses, which have been taken as candidate pixels since the DOG has a strong response on edges, the Hessian matrix is computed at the location and scale of the keypoint. The ratio of the eigenvalues are taken as a stability criterion.

Now the main orientation of the keypoint has to be calculated. This is needed to achieve the rotation invariance of the SIFT feature. The main orientation is calculated on the Gaussian image which has to closest scale to the particular keypoint. The gradient magnitude and orientation of the surrounding pixels of the keypoint are calculated and an orientation histogram is formed. It consists of 36 bins, covering the 360 degree of orientation. Every orientation of the surrounding pixels is weighted by the magnitude of the gradient and also by a Gaussian window. The Gaussian window is used to increase the robustness of the descriptor for small variations and affine distortions. The highest peak in the orientation histogram indicates the main orientation of the keypoint. If there exists a second peak, which is within 80% of the highest, then two keypoints are created. The first using the orientation of the highest peak and the second using the orientation of the second peak.

The SIFT features are described using a 128-dimensional feature vector. It describes the local neighborhood of the keypoint. A coordinate system of a local region is rotated in the orientation of the particular keypoint. Figure 3.3 shows the creation of the descriptor. The gradient magnitudes and orientation and each pixel in the coordinate system is calculated. These are weighted by a Gaussian window, to ensure that pixels at the border have less influence to the descriptor than pixels in the center. The weighted magnitudes and orientation are then accumulated into several (usually 16) orientation histogram which describe the contents over a 4×4 subregion (in Figure 3.3 only a 2×2 array is shown). For each orientation histogram a vector is formed containing its values. The coordinate system for the SIFT feature is 16×16 which is divided into a 4×4 array using 8 orientation bins. Thus, resulting in a 128 descriptor vector for each keypoint.

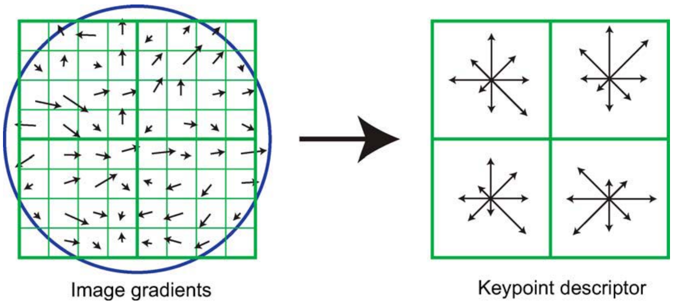


Figure 3.3: Generation of the feature descriptor. The orientation and magnitude in the neighborhood of the keypoints are calculated and weighted by a Gaussian window. Then the coordinate system is divided into an Array and orientation histograms are accumulated. The values of the orientation histogram are then interpreted as vector and all tiles of the array are concatenated as feature vector. (Figure taken from Lowe [57])

.

3.1.3 Bag of Visual Words

The BOW approach which was presented in Section 3.1.1 was brought to the field of computer vision Zhu et al. [87] who partition images into smaller segments, so called keyblocks, which represents the image. Low-level features, such as color, shape, and texture, are used to describe those keyblocks. These descriptions of the keyblocks are then stored in a codebook. The codebook consists of multiple different keyblocks, which represents different objects like “water” or “forest”. A new image is partitioned or segmented into various small elements, Zhu et al. [87] use elements with a size of up to 16×16 pixels, and for each element the most similar keyblock in the codebook is found, this can be seen in Figure 3.4. The image is then encoded using the number of keyblocks used in the codebook at the particular position. Using the keyblocks stored in the codebook, the image can be reconstructed with the indexes stored in the encoded image.

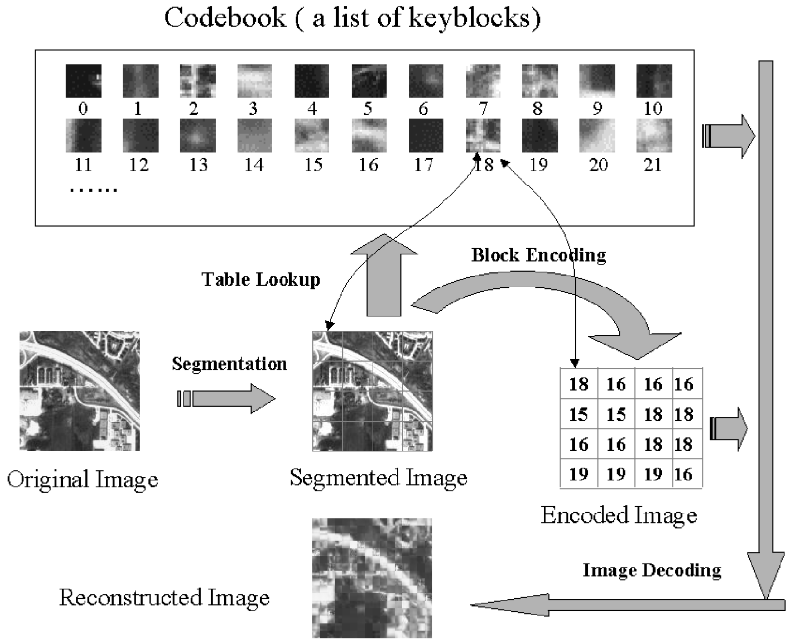


Figure 3.4: Workflow for image encoding and decoding using keyblocks in the image (Figure taken from [87]).

This approach was improved by Csurka et al. [13] for generic visual categorization which should overcome the problem of “identifying the object content of natural images while generalizing across variations inherent to the object class” [13]. Objects should be recognized independently of variations in view, lighting conditions, occlusions, and imaging conditions. Instead of keyblocks they use SIFT features for describing the image. The visual vocabulary $voc = \{v_1, v_2, \ldots, v_k\}$ is generated by using *k-means* clustering of the features in the trainings set. The amount k of the cluster centers is set empirically since a correct clustering is not the goal but a good categorization [13]. When categorizing a new image, first the SIFT features have to be calculated on the image, resulting in a vector:

$$I = \{d_1, d_2, ..., d_N\}$$

where I is the image and d are the descriptors of the SIFT feature. The number N is the total number of the features calculated on the image which is dependent on the content of the image.

For every feature the closest cluster center in the codebook is found and an occurrence histogram is generated. This is done using following equation:

$$v(d_i) = \operatorname*{argmin}_{v \in voc} dist(v, d_i)$$

where $v(d_i)$ is the visual word in the vocabulary where the $i - th$ descriptor d_i is assigned to, according to the distance function $dist$. Usually the Euclidean distance is used to calculate the similarity between two different features respectively a feature and an entry in the visual vocabulary. These histograms are then used for the classification of the image or object. The workflow of the BOW approach on images can be seen in Figure 3.5.

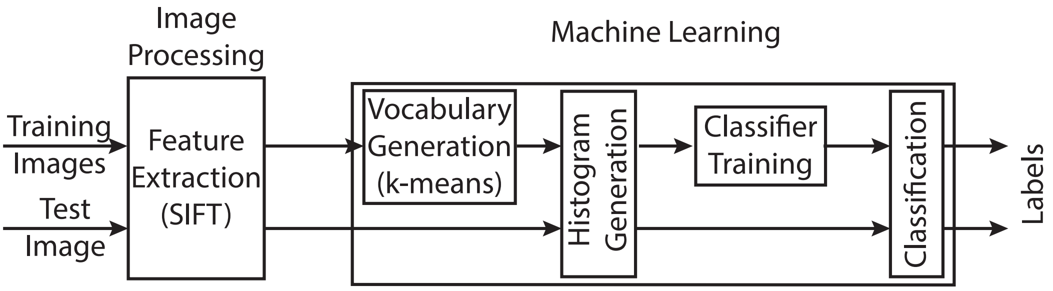


Figure 3.5: Workflow of the Bag of visual Words approach. First the SIFT features are calculated. For the trainings set the vocabulary is generated using *k-means* and a classifier is training using the occurrence histograms of the training images. For a test image also the SIFT features are calculated and the occurrence histogram of the visual words is classified.

Limitations of this approach are that only one object in the image can be recognized and that the background of the image has influence on the classification. Figure 3.6 shows an example of a misclassified image. The image is labeled as “face” whereas the system returns the label “tree”. The features on the background have a too high influence on the classification and thus the result refers to the background.

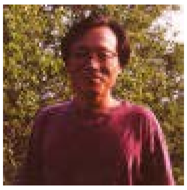


Figure 3.6: Example of a misclassified image with the BOW approach (taken from [13]). The system classifies this image as “tree”, whereas the correct label “face” is only ranked at the 5^{th} position.

Another drawback of this method for object recognition is, that the spatial relationship between the different features is not taken into account. There exist various methods which overcome this problem by building up a spatial pyramid [46] [39] or by encoding the spatial relationship by pairing identical visual words [43]. Work has also been done in analyzing the size of the codebook or different sampling strategies [65], where they state that for object recognition on some databases interest point based samplers, like the Harris-Laplace or Laplacian of Gaussian, cannot compete with uniform random sampling strategies. The reason for this is, that random sampling returns more image patches than interest point based samplers and thus the histogram of the BOW is not accurate enough for these datasets. Furthermore, different cluster methods were applied to the BOW approach, like GMM [22] [71] or mean-shift [41].

3.2 Fisher Vector

Perronnin and Dance [70] improved the BOW approach by applying the Fisher kernel as an extension of the visual vocabulary. The Fisher kernel, introduced in [33], is "a powerful framework which combines the strength of generative and discriminative approaches to pattern classification" [70]. When using Kernel methods the label of a new example is obtained by a weighted sum of the training labels [33]. For this purpose a kernel function k has to be defined, which measures the similarity between two examples. A kernel function is given by the relation [5]

$$k(x, x') = \phi(x)^T \phi(x'). \tag{3.1}$$

with x and x' being two different input vectors. Given a trainingset X_t and its corresponding binary labels S_t then a new sample X can be classified using the following rule [33]

$$\hat{S} = sign(\sum_t S_t \lambda_t k(X_t, X)) \tag{3.2}$$

where λ_t is the overall importance of the example X_t. According to [33] the free parameters are the coefficients of λ and also the kernel function k. An optimization function has to be found to determine appropriate values for the coefficients of λ.

The key idea of the Fisher kernel is to derive the kernel function from a generative probability model by using the gradient space of this model. Let p be a Probability Density Function (PDF) with the parameters λ. The sample x can then be characterized with the gradient vector, the so called Fisher score [70]

$$g(\lambda, x) = \nabla_\lambda log\, p(X|\lambda). \tag{3.3}$$

The Fisher information matrix F_λ can then be formulated as [70]

$$F_\lambda = E_X[g(\lambda, x) g(\lambda, x)^T] \tag{3.4}$$

and a thus the Fisher kernel for this gradient can be defined as [33]

$$k(x, x') = g(\lambda, x)^T F_\lambda^{-1} g(\lambda, x'). \tag{3.5}$$

The visual vocabularies are represented by means of a GMM. Low-level features $X = x_t,\ t = 1 \ldots T$ which are extracted from an image are clustered with a GMM with the parameter set $\lambda = \{w_i, \mu_i, \Sigma_i, i = 1 \ldots N\}$. w_i denotes the weight, μ_i the mean vector and Σ_i the covariance matrix of the $i - th$ Gaussian. In total there are N Gaussians used, which are trained using the Maximum Likelihood criterion. Each of these Gaussians represents a word in the visual vocabulary, where w_i can be seen as the relative frequency of occurrence of the word i, μ_i the mean of the word, and Σ_i the variation [70]. The likelihood of the GMM can be defined under an independence assumption as [70]:

$$L(X|\lambda) = \sum_{t=1}^{T} logp(x_t|\lambda) \tag{3.6}$$

For each feature x_t the likelihood that it was generated by the GMM can be indicated as [70]:

$$p(x_t|\lambda) = \sum_{i=1}^{N} w_i p_i(x_t|\lambda) \tag{3.7}$$

where the sum of all weights $\sum\limits_{i=1}^{N} = 1$ and the components of p_i are given by:

$$p_i(x|\lambda) = \frac{exp\{-\frac{1}{2}(x-\mu_i)^T\Sigma_i^-1(x-\mu_i)\}}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \tag{3.8}$$

where D is the dimensionality of the feature vector. The probability that one of the features x_t is generated by the $i - th$ Gaussians can be defined by [70]:

$$\gamma_t(i) = p(i|x_t, \lambda) = \frac{w_i p_i(x_t|\lambda)}{\sum\limits_{j=1}^{N} w_j p_j(x_t|\lambda)} \tag{3.9}$$

Using the Equations 3.6, 3.3, 3.9 and with the derivations from [70] gives following formulas for the gradient vector:

$$G_{\mu,i}^X = \frac{1}{T\sqrt{w_i}} \sum_{t=1}^{T} \gamma_t(i)(\frac{x_t - \mu_i}{\sigma_i}) \tag{3.10}$$

$$G_{\sigma,i}^X = \frac{1}{T\sqrt{2w_i}} \sum_{t=1}^{T} \gamma_t(i)(\frac{(x_t - \mu_i)^2}{\sigma_i^2} - 1) \tag{3.11}$$

The feature vector, which is used for classification, is generated by concatenating the vectors from Equation 3.10 and 3.11 for every $i-th$ Gaussian. Thus, the feature vector has a dimension of $2DN$, where D is the dimension of the low level feature, in this case the 128 dimensions of the SIFT descriptor, and N is the number of Gaussians used for the GMM.

In [72] Perronin et al. suggested improvements for the image classification with the Fisher Vector. The first improvement is a L2 normalization of the feature vector which removes the dependence on the proportion of image-specific information, which is important especially for small objects. The second improvement is a power normalization of each dimension by $f(z) = sin(z)|z|^\alpha$. The reason for that is, that the more Gaussians are used, the probability for most of these Gaussians that one features is generated by a particular distribution is close to zero. Thus, the gradient vectors are close to null and the Fisher vector is sparser. Figure 3.7 shows the effect of the power normalization, the distribution of the values in the feature vector are shown when using a different number of Gaussians. Figure 3.7(a) uses 16 Gaussians, while (b) and (c) uses 64 respectively 256. The more Gaussians are used, the more zero values are present in the feature vector. When using the power normalization the values the Fisher vector are less sparse, this is shown in Figure 3.7(d). The last improvement Perronin et al. are suggesting is the use of a spatial pyramid and so they receive 8 Fisher vectors for each image: one for the whole image, three for the top, middle and bottom regions and four for each of the four quadrants [72]. This improvement is done for the image classification task so that an error, like the one presented for BOW in Figure 3.6 is less likely.

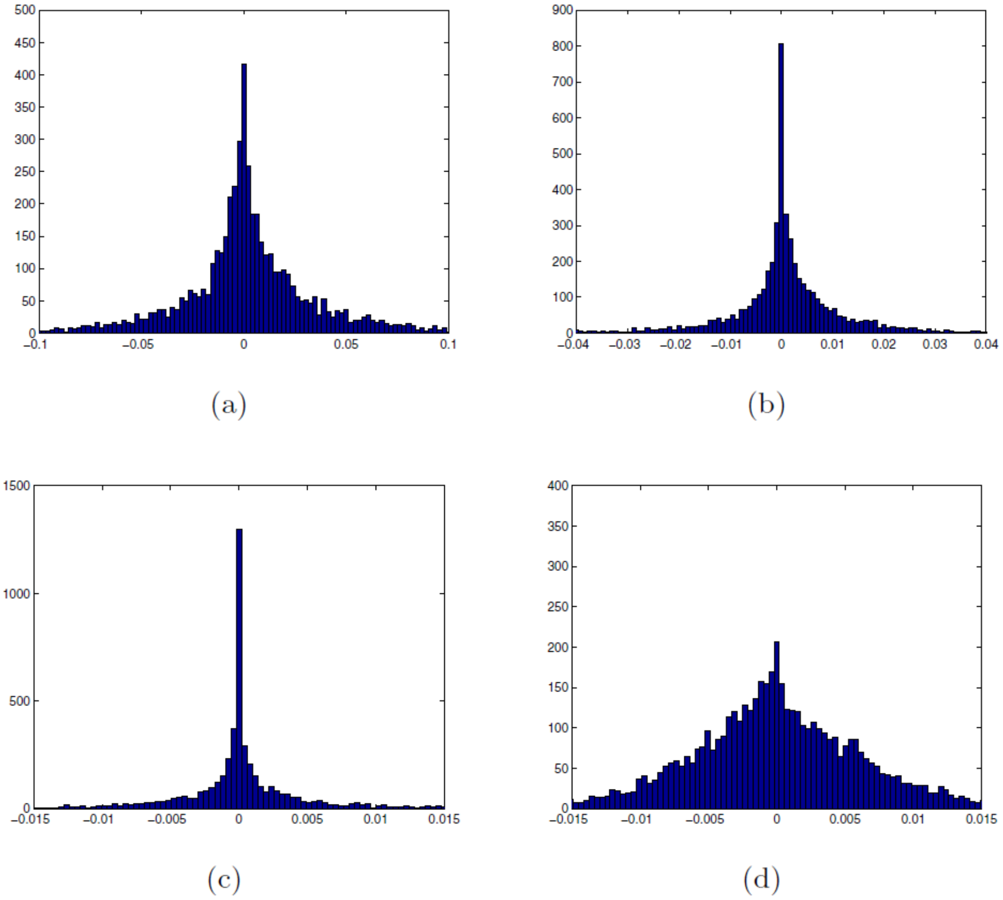


Figure 3.7: Distribution of the values in the Fisher vector when using different number of Gaussians. For (a) 16 Gaussians are used, for (b) 64 and for (c) 256 Gaussians. (d) shows the effect of the power normalization (with $\alpha = 0.5$) on the distribution when using 256 Gaussians. The feature vector is less sparse since not most of the data has zero values (Figure taken from [72]

3.3 Convolutional Neural Networks

CNNs are a type feed-forward artificial neural networks which were introduced in 1980 by Fukushima [28]. He proposes a network which has an ability of unsupervised learning and consists of multiple layers. The neural network should have the same pattern recognition like a human being and "acquires an ability to recognize stimulus patterns based on the geometrical similarity (Gestalt) of their shapes without affected by their position nor by small distortion of their shapes" [28]. After training, called self-organization by Fukushima, the network acquires a similar structure as proposed by Hubel and Wiesel [31] [32]. This hierarchy model of the visual nervous system has the following structure [28]: LGB (Lateral Geniculate Body) → simple cells → complex cells → lower order hypercomplex cells → higher order hypercomplex cells. A cell in the higher layer of the hierarchy tends to respond to more complicated features whereas a large receptive field increases the invariance against shift in position.

Figure 3.8 shows the diagram of the network of Fukushima [28] and the interconnections between the layers. The network consists of two different layers: the *S-layer*, which contains simple cells or lower order hypercomplex cells (*S-cells*) and the *C-layer*, which consists of complex cells or higher order hypercomplex cells (*C-cells*). The two layers are named U_{Sl} for the *S-layer* in the l-th module respectively U_{Cl}. For simplicity reasons only one cell is shown in each layer but all the cells of one layer have input synapses of the same spatial distribution. The only cells which can be modified during self-organization are the *S-cells*. The neural network is successfully applied to simple classification tasks with only 4 classes (first task being 4 numbers and the second being 4 characters), but when using all 10 digits the network was too small for the correct recognition and due to the hardware restriction at that time the network was not enlarged.

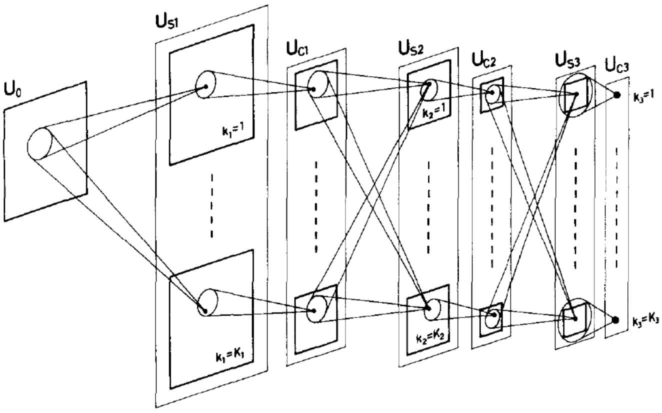


Figure 3.8: Schematic digram of the neural network used by Fukushima [28].

Lecun et al. [48] propose a system, which is not using handcrafted features but instead features which are learned by the system itself. Usually a pattern recognition system is built up on two modules: the feature extraction module and the machine learning module. In the first step handcrafted features are created and the input is described using a (often low-dimensional) feature vector or in later approaches like BOW, as described in Section 3.1.3, a feature vector is generated on top of multiple local features. The feature vectors should be invariant against transformations and should furthermore have the possibility that it can be compared easily. Thus, the feature extraction task often contains prior knowledge of the input data and is rather specific to the task [48]. They mention explicitly that document recognition systems have a lot of preprocessing steps for selecting and segmenting characters on which the features are calculated and that the overall performance of a system is dependent on these steps. The feature vector is then classified using machine learning methods, which have been trained on a database. The authors improve their work which was presented in [47], where they evaluated different designs of CNNs for the recognition of zip codes. According to [48] "CNNs combine three architectural ideas to ensure some degree of shift, scale, and distortion invariance:

1. local receptive fields;

2. shared weight (or weight replication); and

3. spatial or temporal subsampling".

When using a learning from data strategy, a loss function is defined which measures the discrepancy between the correct output of the network and the produced one. The average error of the system, which is calculated using the loss function, should be minimized on the trainings set. In [48] a gradient-based learning is used, which means that the estimated impact of small variations of the parameter is measured using the gradient of the loss function. According to [48] a popular minimization procedure is the stochastic gradient algorithm, also called the online update. It consists in updating the parameter vector using a noisy, or approximated version of the average gradient and is defined as follows

$$W_k = W_{k-1} - \epsilon \frac{\partial E^{p_k}(W)}{\partial W} \tag{3.12}$$

where W_k is the $k - th$ adjustable parameter of the system, ϵ is a scalar constant, and E^{p_k} is the loss function of the $p - th$ input pattern.

Lecun et al. [48] present an architecture of the CNNs for recognizing characters, which is shown in Figure 3.9 and called *LeNet-5*. The input layer gets the images of single characters which are size normalized and centered. Local Receptive field neurons "can extract elementary visual features such as oriented edges, endpoints, corners (or similar features in other signals such as speech spectrograms)" [48]. The subsequent layer combines these features in order to detect higher order features. Units in a layer are organized in planes within all units share the same weights and thus distortion and shift invariance is achieved. In Figure 3.9 the first convolutional layer is organized in 6 planes, each of which is a feature map. The input to this layer is 5×5 area of the input image. The receptive fields of neighboring units overlap. Since the feature maps of the other planes use different weights and biases, they extract different types of local features. The subsampling layer is used to reduce the precision with which the position of distinctive features are encoded in a feature map [48]. This is done by a 2×2 area on which the average input is calculated and multiplied by a trainable coefficient. The last layers of the network are a classical Neural Network (NN).

The LeNet-5 was evaluated on the Modified NIST (National Institute of Standards and Technology) (MNIST) set, which also is presented in [48]. It contains handwritten digits of an image

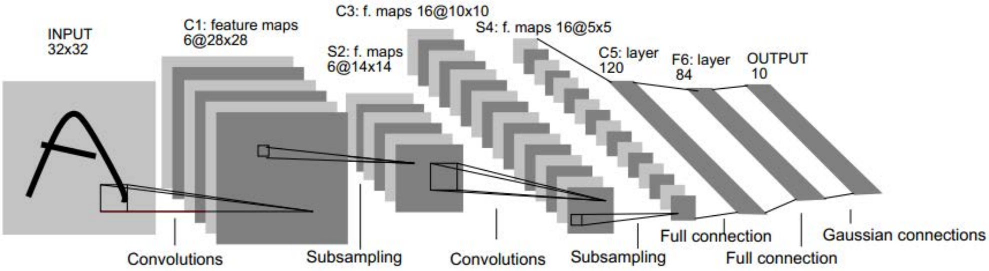


Figure 3.9: Architecture of *LeNet-5* (taken from [48]).

size of 32 × 32 and has 60 000 trainings samples and 10 000 test images. After the publication of the dataset it became a famous dataset for machine learning and various authors evaluated their algorithm on this dataset [45] [75] [12] [1].

With the annually Imagenet Large Scale Visual Recognition Challenge (ILSVRC), which takes place since 2010, the CNNs have become more popular since these methods outperform traditional image classification methods like BOW or the Fisher vector. In 2014 the top performing participants of the ILSVRC all use CNNs in their methodology [76].The dataset of the ILSVRC contains approx. 450000 of manually annotated images for training, approx. 20000 for validation, and approx. 40000 images for testing. Participants are allowed to train their method on the trainings set and then they have to automatically annotate the test images, which are then taken for evaluation. There are two different annotations, the first one being a binary annotation (e.g. “there are cars in this image” but “there are no tigers”) and the second being an annotation at object level, where each object is surrounded by a tight bounding box and a class label. 1000 classes are used for this dataset. For the later dataset a novel crowdsourcing approach was designed for collecting large-scale annotations [14]. The ILSVRC consists of three different tasks: the image classification task, single-object localization task, and object detection task. For the image classification task each submitted method produces a list of object categories present in the image and the quality of this labeling is measured on the label that best matches the ground truth label. For the single-object localization (which was introduced 2011) the algorithms produce a list of object categories present in the image, but this time also with a bounding box indicating the position in the image. The methods are evaluated according to the ground truth label and also its position. This is done to learn the appearance of the target object itself and not its image context. For the last task, the object detection, each method has to provide bounding boxes indicating the position of all target objects, but this time also negative images are present, which do not contain any of the target objects. The quality of labeling is evaluated with precision and recall.

Figure 3.10 shows some sample images from the ILSVRC dataset. Instead of just recognizing the class dogs the methods have to differentiate between 120 breeds of dogs. Also various types of birds and cats are in the dataset. It can also be seen, that the main class of the image is sometimes not covering the main part of the image, e.g. the Egyptian cat, or on some images

[1] Some results are also presented at the homepage of the MNIST dataset, http://yann.lecun.com/exdb/mnist/ (accessed July 2015)

not the whole animal is present e.g. giant schnauzer. Also the number of objects from one class in the image is not limited, e.g. three quails. Thus, the ILSVRC is also challenging for humans since some classes are only familiar to experts in the field. One of the authors tried to calculate the human accuracy on the ILSVRC dataset[2]. For this calculation he only used a subset of the dataset, because the manual labeling process was too time consuming. He achieved an error rate of 5.1%. The GoogLeNet [82], which was the winner of the 2014 challenge, has an error rate of 6.8%.

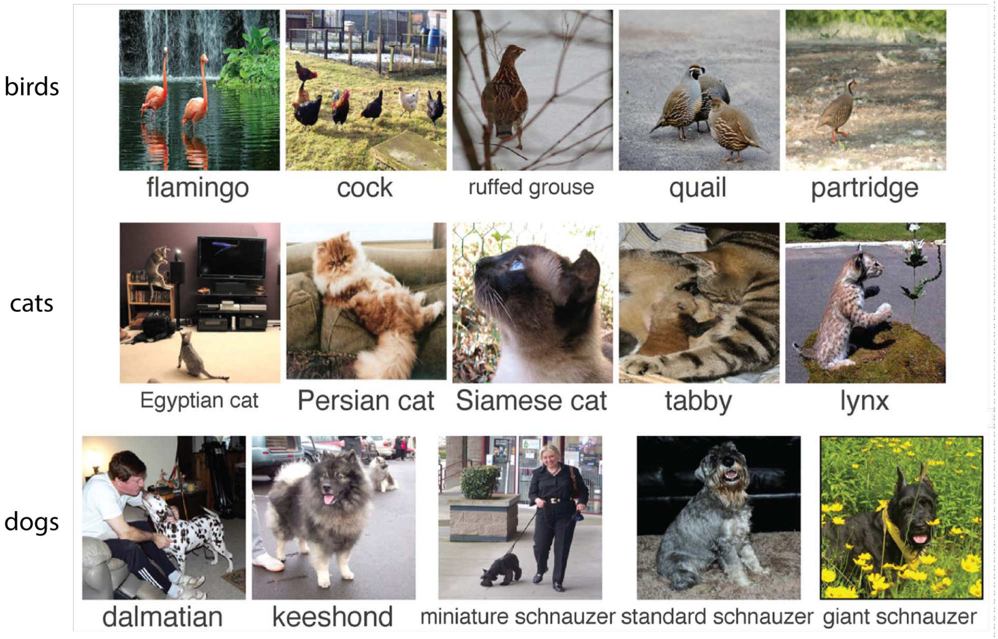


Figure 3.10: Samples images from the ILSVRC. The dataset consists out of 120 different breeds of dogs, and various species of dogs and cats. (Figure taken and adopted from [76]).

Apart from the work done on the MNIST database for digit recognition, CNNs are also used for character recognition [34]. Wang et al. [84] take use of the CNNs for text recognition by proposing an end to end recognition system and they are also used for detecting and recognizing text in natural scenes [88] [67].

To apply CNNs to an image classification task a trainings set of images is used. The more images are available, the better the CNN learns the structure from these images. To ensure that the network does not overfit to the training data, a possibility is to use cropped version of the images [12] [82]. Every epoch a new crop, which only cuts away a few pixels at the border, is carried out and this image is then given to the network. The exact location of the cropping is chosen randomly. The user has to follow the loss value on the trainings set and the accuracy of the evaluation dataset to check for the correct epoch, when the CNN is no longer improving the classification rate but is still reducing the loss value of the trainings set. From this point

[2] http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/ (accessed July 2015)

on the CNN is just overfitting to the trainings set, thus the loss value is decreasing. For a new classification also subcrops of the images are used. Usually 10 subcrops of the image are cut out and each of these images is handled to the CNN for classification. The mean values of the classification layers is then taken as the result of the classification of the original image.

3.4 Summary

In this chapter concepts, which are used for the development of writer identification and writer retrieval methods, have been presented. The first concept, namely the BOW, is based on an approach used for natural language processing. SIFT features are extracted on the image and with the help of *k-means* clustering a dictionary of (visual) words has been defined. The occurrences of these words in the image is then used to generate a histogram which can then be used for classification. The second concept is the Fisher vector, which is an extension of the BOW. Instead of a strict partition of the feature space a GMM is used for clustering. The gradient of the probability function of the Gaussians are then used to form the feature vector for each image. This feature vector is then used for classification. The last concept which has been presented is the CNN, which is, in contrast to the previous methods, not only responsible for the generation of feature vectors, but also for the feature extraction and the machine learning part. Training images are used as input to the CNN which learns automatically the distinctive feature of the different classes and these features are then fed into a neural network, which is responsible for the classification. It has been shown in this chapter, that the CNN show a high performance on image classification tasks, like the ILSVRC [76].

CHAPTER 4

Methodology

In this chapter three methods for writer identification and retrieval, which were developed within the scope of this thesis, are presented. The methods are based on the concepts, which have been presented in Chapter 3. The first method is based on the BOW, followed by a method which uses the Fisher vector. The last method presented uses a CNN for writer identification and retrieval. For the first two methods a parameter evaluation is included in the respective section to find the optimal parameters for this task. A short summary is given in the last part of this chapter.

4.1 Writer Identification Using BOW

The methodology for performing writer identification using the BOW approach is presented in [24]. The workflow is shown in Figure 4.1. First the SIFT features are calculated on the document image. For every feature the nearest cluster center is searched and a occurrence histogram is built up. These histograms are then used to calculate the similarity of the handwriting.

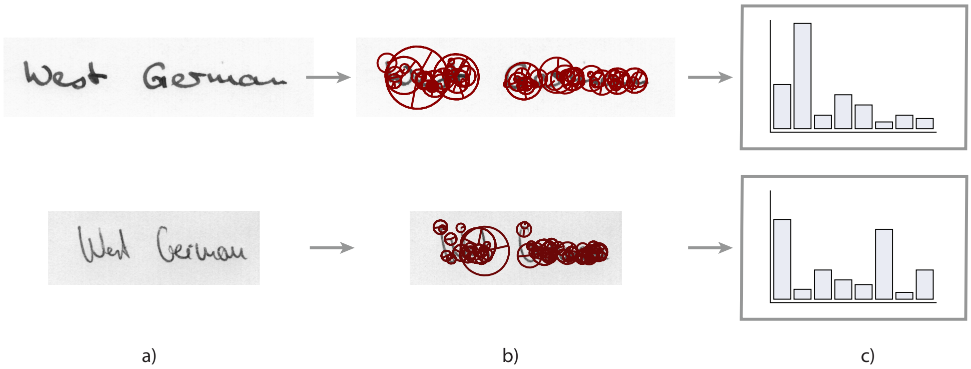


Figure 4.1: Workflow for writer identification using the BOW approach. a) input image b) SIFT features are calculated on the input iamge c) generation of occurrence histograms.

SIFT features are also detected in small scales. Since it is possible that these features do not have any or even a worse influence to the performance of writer identification, because they do not carry much information about the writer itself but rather information about the pen used, it is evaluated later if filtering out small features does have an influence to the performance. Figure 4.2 shows such a sample image and some of the corresponding SIFT features where the small features have been highlighted. It can be seen, that the highlighted features do not describe the writing itself. Within their neighborhood the features rather contain information about their pen, like roundness and sharpness. Feature with a higher scale than 100 pixels are removed, since they cover whole characters or even words.

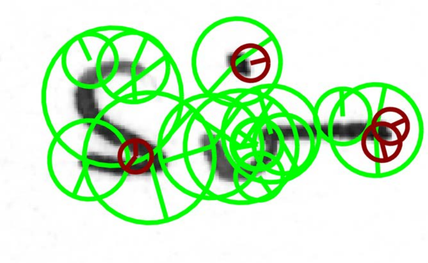
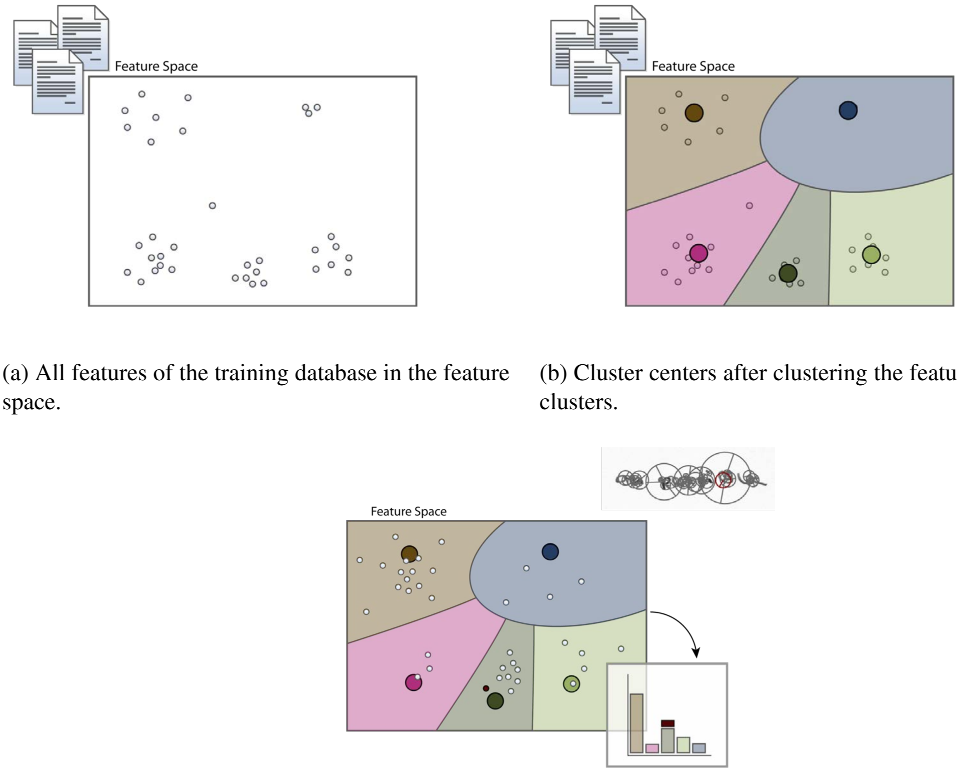


Figure 4.2: One word of the IAM-DB dataset which some of its corresponding SIFT features. Features with a small scales are marked in red and are skipped for further calculation since they do not describe the characteristics of the handwriting.

Figure 4.3 shows the generation of the visual vocabulary for writer identification. All features of a training database are inserted into the feature space, which is illustrated in Figure 4.3 (a) for some features. These features are then clustered according to a predefined number of clusters. Figure 4.3 (b) shows the cluster centers when the number is set to 5. *k-means* is used for clustering, which minimizes the within-cluster sum of squared differences. The standard algorithm for this clustering method was first proposed by Lloyd in [52]. The assumption is that the distribution of features in the feature space describes the characteristics of a specific writer. For this work, the *k-means* method is carried out 10 times and the best result is taken as visual vocabulary. This is done to avoid that the *k-means* method is stuck in a local minima. To model this distribution the BOW approach can be used by assigning each feature to the nearest cluster center. For SIFT features the Euclidean distance has been proposed by Lowe in [57]. The occurrences of the assignments to one cluster center is used to form an occurrence histogram, which is shown in Figure 4.3 (c). When identifying a writer of an unknown page, the SIFT features are calculated and for each feature the nearest cluster center is searched. Each cluster center forms a bin in a histogram and number of occurrences that a center is the closest to one feature is the value in the histogram. Since the number of features in an image is variable, the histogram is normalized with the number of features. The number of cluster centers used is determined empirically.

Parameter Evaluation

The parameter evaluation is used to find a parameter set which optimizes the performance for the given task. Therefore, the free parameters of the given methods are varied and the accuracy



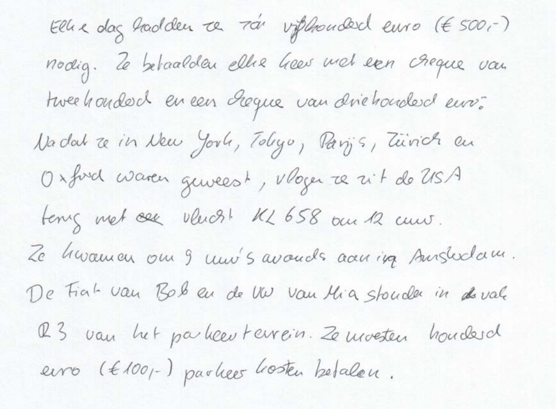
(a) All features of the training database in the feature space.

(b) Cluster centers after clustering the features into 5 clusters.

(c) Creation of the occurrence histogram. The feature in red is counted as occurrence for the third center.

Figure 4.3: Generation of the visual vocabulary. First the features of the training database are inserted into the feature space and then clustered according to a predefined number of clusters. When identifying a new image then the SIFT features are calculated on that image and for each feature the nearest cluster center is searched and the occurrences are counted.

of the method on a dataset is investigated. The parameter set with the best performance on this dataset is then chosen for the extensive evaluation of the method on different datasets. Since it is possible, that the best performing parameter set is overfitted to the evaluation set a second parameter set with a similar performance but different values is also used for the extensive evaluation set. The free parameters for the method proposed are the minimal size of the SIFT features and the number of cluster centers. To evaluate the parameters, the TrigraphSlant database is used to generate the vocabulary and the tests were made on the ICDAR 2011 datasets. The TrigraphSlant database by Brink et al. [6] consists of 188 document images. For the creation of the database 47 writers have written four different texts in Dutch. The first two texts are written with the natural handwriting and the remaining two are written with the maximal slant to the left respectively to the right. For training all document images are taken into account, since for the creation of the vocabulary a forged handwriting does not have any impact. This database was chosen as training dataset to keep the evaluation databases presented in Chapter 2 untouched and because of its size, since it is small enough that the whole database can be used for clustering and there is no need for generating a subset.



Elke dag hadden ze zó vijfhonderd euro (€ 500,-)
nodig. Ze betaalden elke keer met een cheque van
tweehonderd en een cheque van driehonderd euro.
Nadat ze in New York, Tokyo, Parijs, Zürich en
Oxford waren geweest, vlogen ze uit de USA
terug met vlucht KL 658 om 12 uur.
Ze kwamen om 9 uur 's avonds aan in Amsterdam.
De Fiat van Bob en de VW van Mia stonden in de vak
R3 van het parkeerterrein. Ze moesten honderd
euro (€100,-) parkeer kosten betalen.

Figure 4.4: A sample page of the TrigraphSlant database.

For the filtering of the SIFT features the minimum size of the features was set to 0, 10, 20, 30, 40, 50, 60, and 70. For the number of cluster centers the values 50, 100, 200, 250, 300, 350, 400, 500, 600, 700, and 800 were chosen. Figure 4.5 shows the mean accuracy for the hard Top-1, Top-2, Top-3, and Top-4 criterion using the different parameters. The detailed plots for each parameter set are shown in Figure A.1 in Appendix A. It can be seen, that the highest mean accuracy value is achieved when using a Filter size of 10 for the SIFT features and using 250 cluster centers for the *k-means* clustering. A filter size of 30 is too high for the method proposed and thus the accuracy is dropping rapidly with increasing filter size. It can also be seen that for these databases the filtering of the features according to the size is not necessary, since 250 cluster centers have a similar high accuracy. When choosing too few cluster centers the accuracy is also lower since the partition of the feature space is not accurate enough to get a precise description of the writer using the BOW approach. Only when the filter size is chosen very high, a few clusters are sufficient for the best performance on these sets since there are only a couple of features left per image. If the size of the visual vocabulary is set too high, the performance is also slightly dropping since the feature space is partitioned in too many parts and thus similar SIFT features are assigned to different cluster centers. Interesting is the performance drop when using 200 cluster centers for the filter size of 10 and 20. The only explanation for this is, that for this trainings set *k-means* returns a partitioning of the feature space that is not usable, in terms of accuracy, on this trainings set for writer identification.

4.2 Writer Identification Using the Fisher Vector

The approach of writer identification using the Fisher vector is presented in [25]. Again, like for the BOW approach first the SIFT features are calculated. Modified SIFT features, as proposed by Diem and Sablatnig [17], are also used. The modification makes the feature orientation

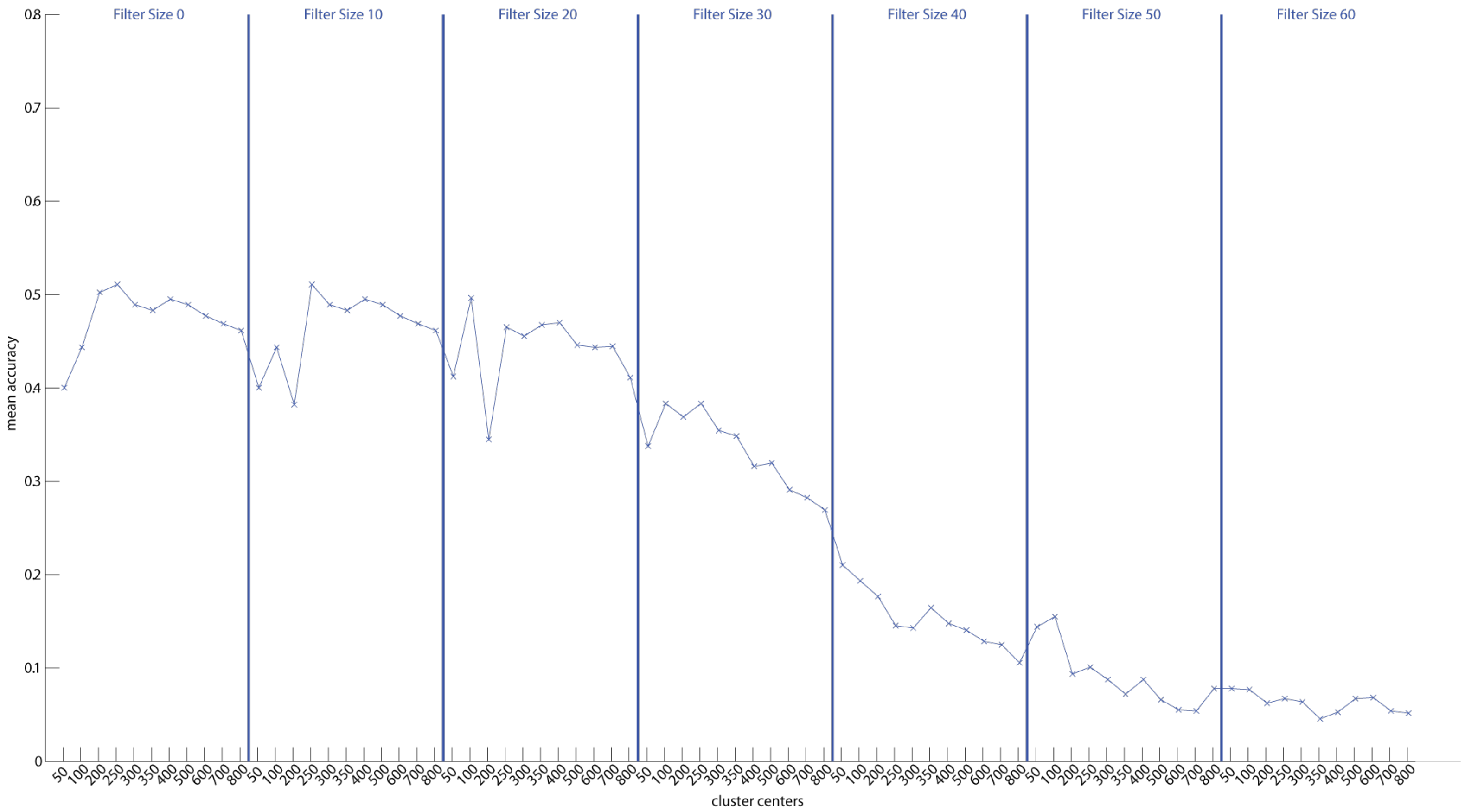


Figure 4.5: Mean accuracy of the hard Top-1, Top-2, Top-3, and Top-4 criterion on the ICDAR 2011 cropped dataset with different parameters.

variant to 180 degrees. This is done because the upper and lower profile of the writing is a discriminative feature. When the SIFT feature are rotation invariant, two features on the upper and the lower profile cannot be distinguished. Figure 4.6 shows the same word written by two different persons. The red marked features in the left image generate nearly the same descriptor as the red marked features in the right image. Also note that the left handwriting does not contain any features like the red ones in the upper profile. Thus, the discrimination between features generated on the upper respectively lower profile is desirable.

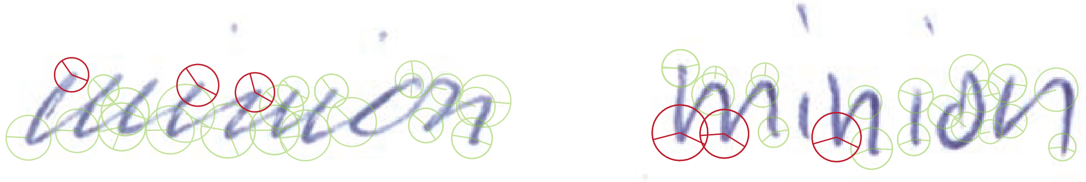
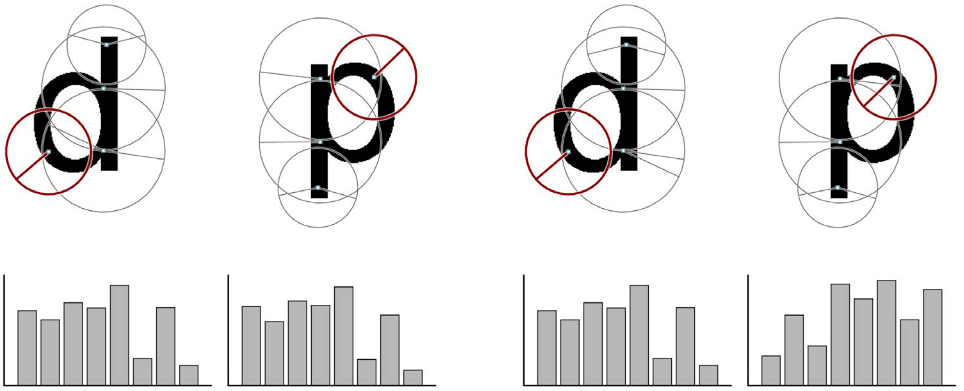


Figure 4.6: The word "minion" written by two different writers. The selected features on the left generate a similar descriptor as the selected features on the right. With the 180 degree rotation variance these features can be distinguished and thus the upper and lower profile of the writer is described.

Diem and Sablatnig [17] achieved this discrimination by flipping the main orientation of the keypoint if its angle is greater than 180 degrees. Figure 4.7 shows the effect of mirroring the keypoint orientation. Figure 4.7 (a) shows the characters *d* and *p* and some of their corresponding SIFT features and for two of them the descriptors. Both red marked features generate the identical descriptor and thus it is no longer distinguishable if the keypoint is generated on the upper or lower profile of the character. When using a rotation variant SIFT feature, like in Figure 4.7 (b), the descriptor of the *p* changed and a distinction between both descriptors is possible. The change in the descriptor is only a permutation, because only the main orientation of the feature is changed and due to the binning step of the SIFT descriptor the gradients are assigned to a different bin.

The modified features are inserted into the feature space, like for the BOW approach described in Section 4.1. This time instead of a k-means clustering a GMM is fitted to the data. The advantage is that the partitioning of the feature space is no longer with strict borders, instead the PDF of the Gaussians are used to determine the distance of the feature and the cluster center. This is illustrated in Figure 4.8, the dashed lines are the partitioning of the feature space with k-means, where the colors indicate the probability functions of the GMM.

According to Perronin et al. [72] the feature vector is then generated using the Equations 3.10 and 3.11. So for each Gaussian the gradients have to be calculated and then these vectors are concatenated, resulting in a $2DN$ dimensional features vector, where D is the dimension of the SIFT feature and N the number of Gaussians. Thus, with increasing numbers of Gaussians the dimension of the feature vector increases rapidly and slows down the calculation. Experiments have shown, that the derivation to σ of the gradient vector (Equation 3.11) does not have much influence for writer identification and may also decrease the identification rate than when using only the derivation to μ for each Gaussian as feature vector. Thus, only Equation 3.10 is used for further evaluation, resulting in a DN dimensional feature vector, which has also the positive



(a) Sample SIFT features with rotation invariance.

(b) Sample SIFT features with rotation variance up to 180 degrees.

Figure 4.7: SIFT features calculated on the letters d and p. (a) When using the normal SIFT feature with rotation invariance, the red marked features generate identical descriptors. (b) When flipping the main orientation of the keypoint the histogram of the p is permuted, making both descriptors distinguishable. For visualization reasons the 128 dimension of the descriptor has been quantized to 8 dimensions.

side effect that the computational time for the generation of the feature vector as well as for the comparisons for writer identification is reduced by nearly one half. The next modification which is made to the original method from Perronin is that the normalization term in Equation 3.10 is eliminated. The feature vector of one image is generated by concatenating the following values for each Gaussian:

$$G_i^X = \frac{1}{\sqrt{w_i}} \sum_{t=1}^{T} \gamma_t(i) (\frac{x_t - \mu_i}{\sigma_i}). \tag{4.1}$$

The normalization term T in Equation 3.10 eliminates the effect that the number of SIFT features has an influence on the feature vector. Since a $L2$ normalization step is introduced by Perronin in [72], this normalization is no longer necessary. Afterwards the power normalization, which is proposed by Perronin et al. [72], is done to make the feature vector less sparse.

In [70] and [72] the Fisher vector for image classification is used and for both experimental setups they apply the Principal Component Analysis (PCA) to reduce the dimensions of the SIFT descriptor. They do not argue why they introduce this step, but it can be assumed that it is for the reduction of calculation time without losing performance. Since the dimension of the feature vector for one image depends linearly on the dimension of the local features used the dimension of the feature vector can be reduced drastically. When using all 128 dimensions of the SIFT features and 50 Gaussians the dimension of the feature vector describing the image is 12800 (taking the original formula). When taking the first 64 components of the PCA, like proposed in [72], the dimension is reduced to 6400. This reduces the computation time for the generation of the feature vector as well as the time for comparing feature vectors.

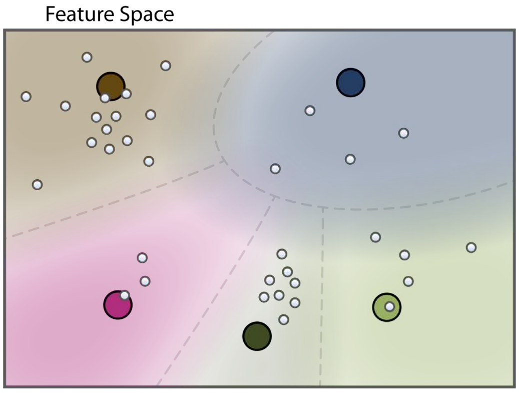


Figure 4.8: Difference of the partitioning of the feature space when using GMMs or $k - means$. When using the $k - means$ (dashed lines) the borders between the clusters are strict and no difference is made between features at the border of the cluster or directly at the center. When using GMMs the borders are smoother and features at the border have also influence to other Gaussians.

Parameter Evaluation

Like for the BOW method a parameter evaluation is carried out to find the optimal set of parameters for this method. The main free parameters for this approach are the use of the rotation variant SIFT descriptors, the minimum size of the SIFT features which is allowed, the number of Gaussians used, and the number of PCA components. These four parameters have been evaluated on their performance. The use of the rotation variant descriptor is a binary decision whereas for the number of Gaussians 5, 10, 20, 30, 40, 50, and 60 are used. For the size of the SIFT feature the maximum is 100 pixels and the minimum size has been varied between 0 and 60 pixels. The number of components of the PCA have been evaluated on 32, 64, 96, and without applying the PCA. For evaluation, the GMM was created on the TrigraphSlant dataset, like for the parameter evaluation of the BOW method, and for testing the ICDAR 2011 cropped dataset has been used. The hard criterion, more precise the $Top1$, $Top2$, $Top3$, and $Top4$ hard criteria, has been taken to calculate the accuracy. Figure 4.9 shows the mean accuracy on the evaluation dataset for the rotation invariant SIFT descriptor with different settings of filter size, Gaussians and PCA components. Figure 4.10 shows the same plot using the rotation variant SIFT descriptor. The detailed plots for all evaluations can be found in Appendix B. It can be seen, that a filter size of more than 30 decreases the performance of the method for both SIFT features dramatically. Too much information is lost if these features are skipped. When using a filter size of 30, the performance is only comparable to the ones with lower filter size if a lower number of Gaussians is chosen. Also, when taking too few Gaussians, in this case 5, the performance of the method is lower than when taking a higher number of Gaussians. The same effect happens when the number of Gaussians is set too high, it can be seen that for nearly all filter sizes the performance is the worst when using 70 Gaussians. These properties are the same for the rotation variant as well as for rotation invariant SIFT features. Furthermore, it can be seen that the different numbers of PCA components taken does not have a strong influence on the performance of the method proposed. When taking 32 and 64 components and the rotation invariant descriptors the performance is on average lower than when taking 96 components or without applying the PCA. When taking the rotation variant version of the SIFT features the performance with 32 components is overall the worst, whereas when taking 64 components the performance is still comparable to the other two combinations. When the filter size is set too high, in these evaluations 30 or above, the information less introduced by the PCA influences the performance of the method, since there are also too few features to generate a reliable description of the handwriting.

The highest performance of all the values evaluated is achieved when using the rotation variant descriptor with a filter size of 20, 10 Gaussians, and the first 96 PCA components. The second highest performance has the same parameters, except that the rotation invariant feature is taken. These two combinations are taken for further evaluation, additionally the combination of both features types with the filter size of 10, 40 Gaussians and 64 PCA components are taken since this combinations has also good performance and its values differ from the best ones.

4.3 Writer Identification Using Convolutional Neural Network

In Fiel and Sablatnig [26] the methodology for writer identification using a CNN is presented. Since the network requires the same size for all input images a preprocessing steps are introduced. First the text lines are detected, for the CVL-DB and for the IAM-DB word images are provided by the authors and thus this step is not necessary for these two databases. The text lines are then skewed to a horizontal direction and the handwritten words are size normalized. Then a sliding window is used to cut out squared images which are then used as input for the CNN. An accumulated feature vector for all images is then used for the identification of the writer respectively for the retrieval of pages with similar handwriting. Since the CVL-DB is the only dataset used which is not binarized, a binarization is performed. Because the dataset contains only scanned pages without any noise, a global threshold leads to a good binarization and thus the method of Otsu [68] is used.

The first step of the binarized images is the text line detection for the datasets which does not provide line or word images. For this task the method of Diem et al. [16] is used. This algorithm uses LPP for grouping the characters to words. The text lines are then detected by globally minimizing the distances of all words. The distance of two words is weighted by the angle between two words to avoid false merges because of low line spacing. Figure 4.11 shows the merging of the words. Since the angle between “and” and “Ocean” is too high this false merge is avoided.

The text lines now have to be deskewed by using the angles of the lower and upper profile lines. To generate the profile lines a simple algorithm is used: for each column of the image the first and the last foreground pixels are searched. If there is no foreground pixel in this image column then this column is skipped for the calculation. To eliminate the influence of the height

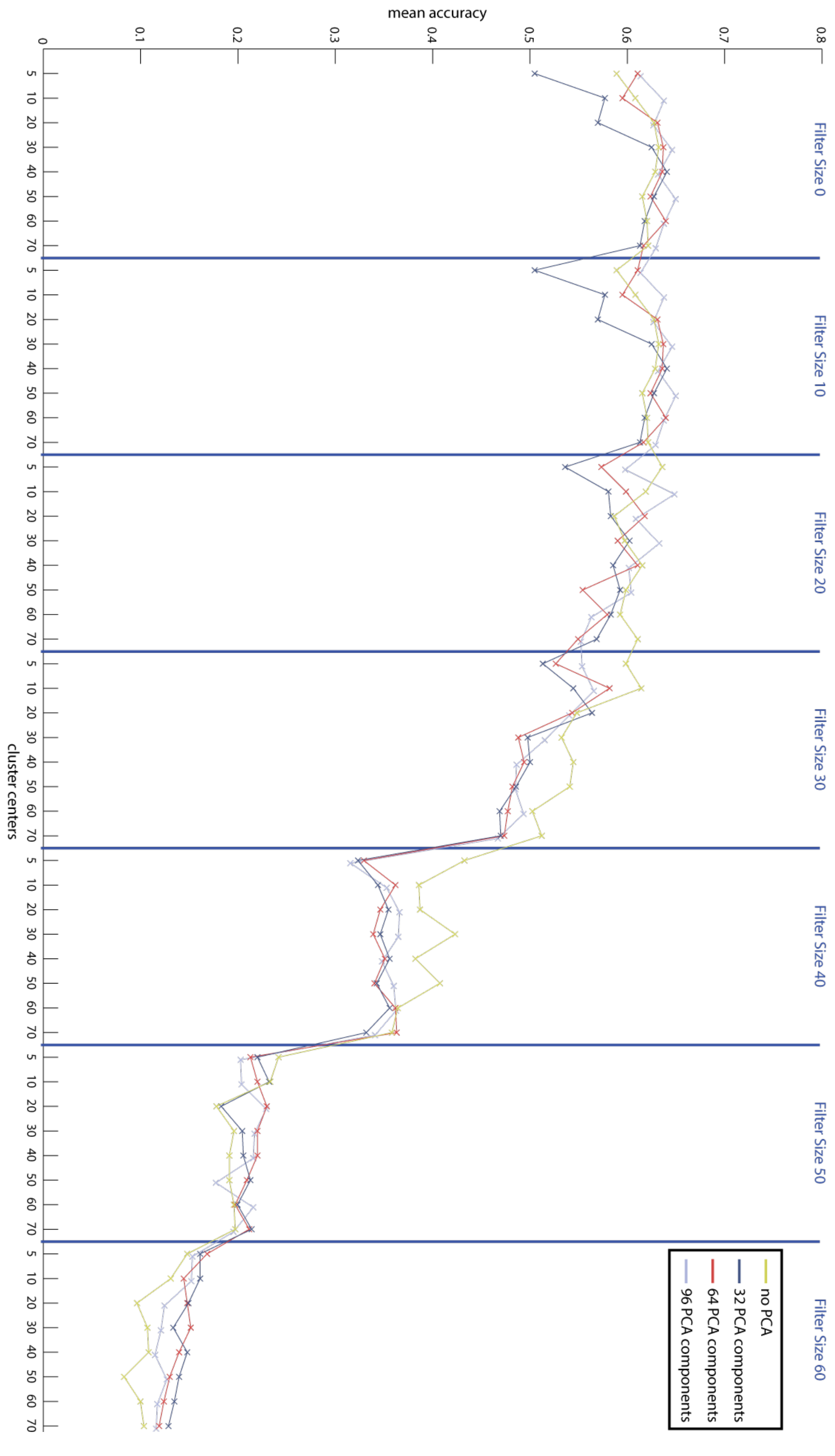


Figure 4.9: Mean accuracy of the hard Top-1, Top-2, Top-3, and Top-4 criterion on the ICDAR 2011 cropped dataset using the unmodified SIFT descriptor and different parameters.

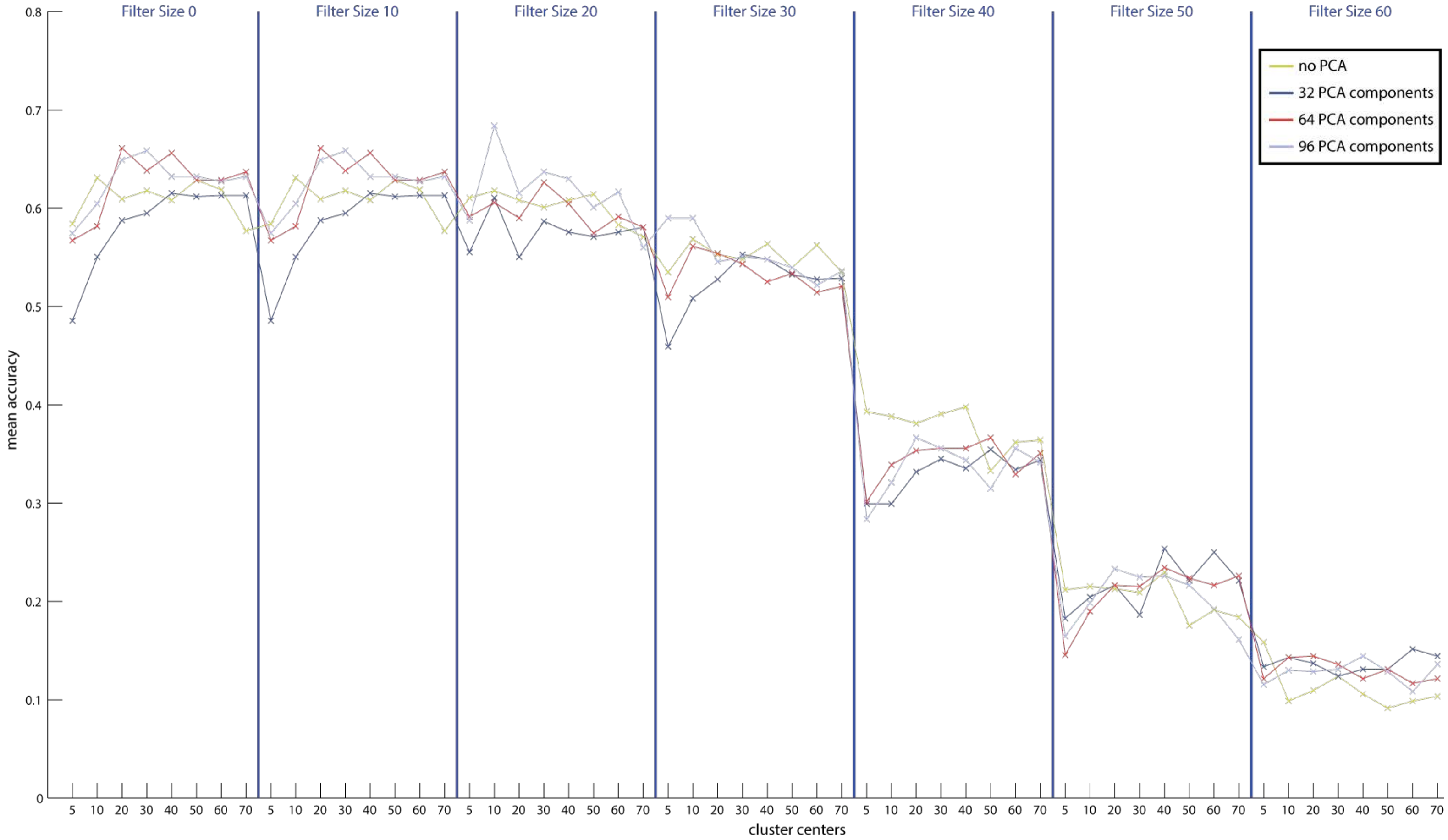


Figure 4.10: Mean accuracy of the hard Top-1, Top-2, Top-3, and Top-4 criterion on the ICDAR 2011 cropped dataset using the rotation variant SIFT descriptor and different parameters.

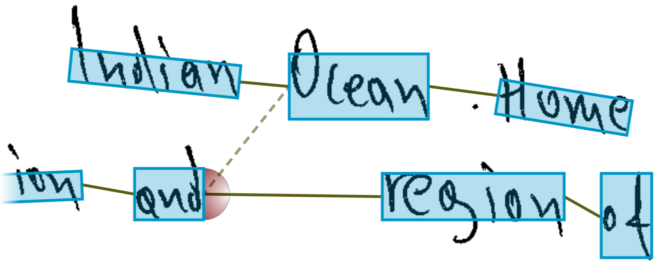
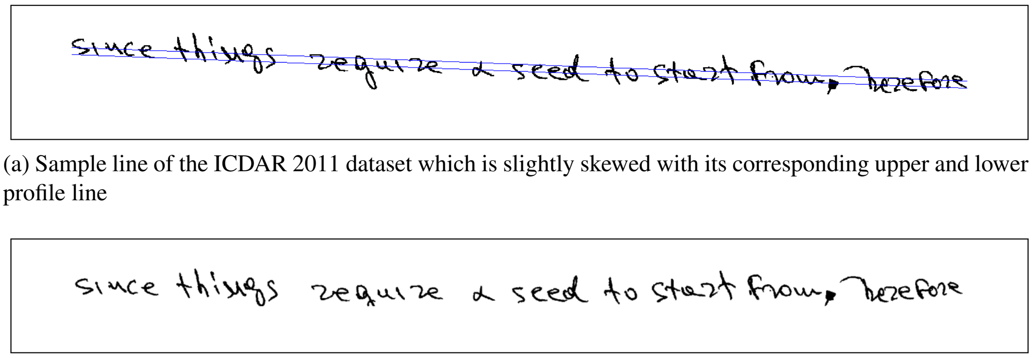


Figure 4.11: Merging of the words into text lines. The distances weighted with the angle between all words are minimized. The dashed lines shows a false word merge which was avoided (Figure taken from [16]),

of the writing to the identification, the size of the handwriting is normalized. The x-height, which has been calculated while deskewing, is normalized to half of the size of the required height by the CNN. Half of the height has been chosen to ensure that the ascenders and the descender of the handwriting have sufficient space to be present in the image. Figure 4.12 shows a sample line of the ICDAR 2011 dataset which is also slightly skewed. The corresponding upper and lower profile is generated and two lines are fitted through these points. Then the line is skewed by the mean angle of the two profile lines and as last step the x-height is normalized to a fixed value.


since things require a seed to start from, therefore
since things require a seed to start from, therefore

(a) Sample line of the ICDAR 2011 dataset which is slightly skewed with its corresponding upper and lower profile line

(b) Deskewed line and the x-height is normalized to 20 pixels.

Figure 4.12: Pre-processing steps of the CNN approach including deskewing of the text line and height noramlization.

For the generation of the feature vector, which is then used for writer identification or retrieval, a CNN is used. For this work a well-known model, namely the “caffenet”, which is part of the “Caffe - Deep learning framework”[1] [38], is used. The design of the network is presented in Figure 4.13. It consists of five convolution layers which are using kernel sizes of 11 to 3, followed by three fully connected layers. Since the last layer of the network is for labeling the input images to the correct class, this layer is cut off and the second last layer is used as a feature

[1] http://caffe.berkeleyvision.org - accessed July 2015

vector. This is done because a classification is not requested, instead the result of the network should be a vector which can be compared to other classes. The classification can be used, if a database of known writers is available and only the handwriting of these writers have to be identified, but this is not the case for the evaluation scheme presented in Section 2.1.7.

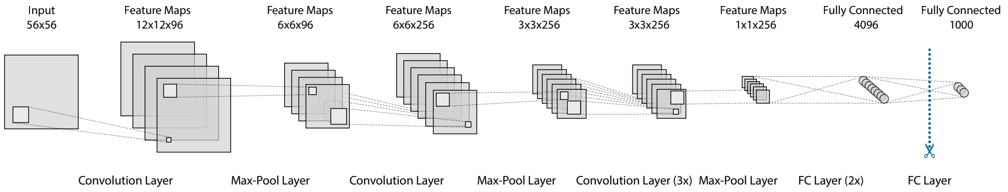


Figure 4.13: Design of the CNN, the “caffenet” of the “Caffe - Deep learning framework”

The CNN has to be trained beforehand. To ensure the independence between the training images and the test set, the CNN is trained on the IAM-DB dataset. The last layer consists of 1000 neurons, which is actually more than writers in the IAM-DB, but experiments have shown that the performance is dropping when using 657 neurons which equals the number of writers in the dataset. Since CNNs have to be trained on a large amount of data to achieve a good performance (e.g. for the ILSVRC 2014 the trainings set contained 1.2 Mio images for 1000 classes), the trainings set has to be enlarged artificially. This is done by rotating each image patch which has been cut out using the sliding window from -25 to 25 degrees with a 5 degrees step. This rotation of the image also may have a positive effect on the writer identification. Figure 4.14 shows some portions which are used as trainings set for one writer of the IAM-DB. The CNNs are invariant against translations and various forms of distortions, but not invariant against rotations. Writing with different slants can be seen as some kind of rotation of the characters and the rotation of the training images can make the net invariant to this kind of distortion. This property has to be confirmed in future work. With the rotation of the image, the trainings set consists of more than 2.3 Mio image patches. Due to the properties of the IAM-DB these patches are not equally distributed between the writers, thus each writer has at most 7700 patches (700 images patches which are rotated 10 times) in the trainings set.

To generate the feature vector for a complete document image, all images patches which are extracted on the handwriting are fed into the CNN. The last layer of the neural network is cut off, since it is responsible for the classification which is not needed for this task, and the output of second last layer is used as output of the CNN. Since the last layer has 4096 neurons, the activation of these neurons are used as feature vector of this patch. The mean values of all the vectors from the image patches are then calculated and used as feature vector for the writer identification or retrieval. Experiments showed that the χ^2-distance is a good similarity measurement for the features vectors generated with CNNs.

This method is developed rather as a proof of concept, future work will include the design of a network architecture which is better suited for this problem and also the effect of different pre-processing steps will be investigated. It is the first attempt to bring deep learning to the field of writer identification and writer retrieval.

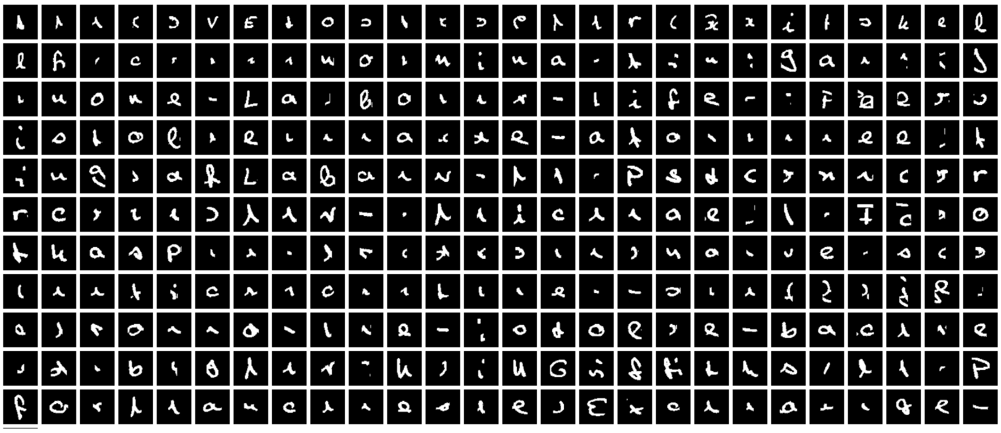


Figure 4.14: Image patches of one writer of the IAM-DB which are used in the trainings set. These patches are then also rotated from -25 to 25 degrees in 5 degree steps.

4.4 Summary

In this chapter three different methods for writer identification and writer retrieval based on the concepts presented in Chapter 3 have been presented. The first two methods are working with SIFT features which are clustered and then an occurrence histogram is generated or the gradient of the probability function is calculated. The most important parameters of these methods have been determined by empirical evaluation. The third approach is based on CNN. The CNN has to be trained beforehand on small image patches of the handwriting and when identifying a writer, each image patch is processed. For each image patch the activation of the neurons in the second last layer is used and for the generation of the feature vector for the writer, these activations are averaged. Two methods use the χ^2 distance to calculate the similarity between two handwritings and for the Fisher Vector the cosine distance is used.

CHAPTER 5

Evaluation

The evaluation is carried out on the different datasets presented in Section 2.1. For generating the vocabulary the TrigraphSlant dataset is used for the BOW and the Fisher vector approach, for training of the CNN the IAM-DB is used to ensure independence from the evaluation sets. Parameters have been set as described in the particular section of the Methodology. The evaluation method is as described in Section 2.1.7, so the hard and soft criterion from the ICDAR competitions are used, as well as the retrieval criterion of the CVL-DB. The last part of this section contains the comparison of the results.

5.1 Evaluation of the BOW Method

In Section 4.1 the size for the visual vocabulary was empirically evaluated and the best performance was achieved when using 200 cluster centers for *k-means* using a filter size for the SIFT features of 10 or 20. The modification of the SIFT features, which has been presented in Section 4.2 can also be applied for the BOW approach. So an additional parameter evaluation is carried out to show the performance of the method proposed using the rotation variant features and the effects of applying a PCA beforehand. Figure 5.1 and Figure 5.2 show the mean accuracy of the hard Top-1, Top-2, Top-3, and Top-4 criterion on the ICDAR 2011 cropped dataset for the unmodified SIFT features as well as for the modified. The detailed results for each parameter combination are shown in Appendix A. It can be seen that there is only a slight difference between the modified and the unmodified SIFT features. In contrast to the Fisher vector approach, applying the PCA has no positive effect on the performance. Since the dimensions of the feature vector are already low when using BOW, PCA is not used in the evaluation. For the rotational variant SIFT features the best performance is achieved when using 300 cluster centers and thus for further evaluation the size of the cluster centers is 250 and 300 for both types of SIFT features. In all tables they are named ROTVAR-250 respectively ROTVAR-300 for the modified features and the corresponding size of the visual vocabulary and INVAR-250 and INVAR-300.

The first evaluation is carried out on the ICDAR 2011 dataset. Table 5.1 shows the evaluation of the soft criteria on both datasets compared with the best ranked participants of the

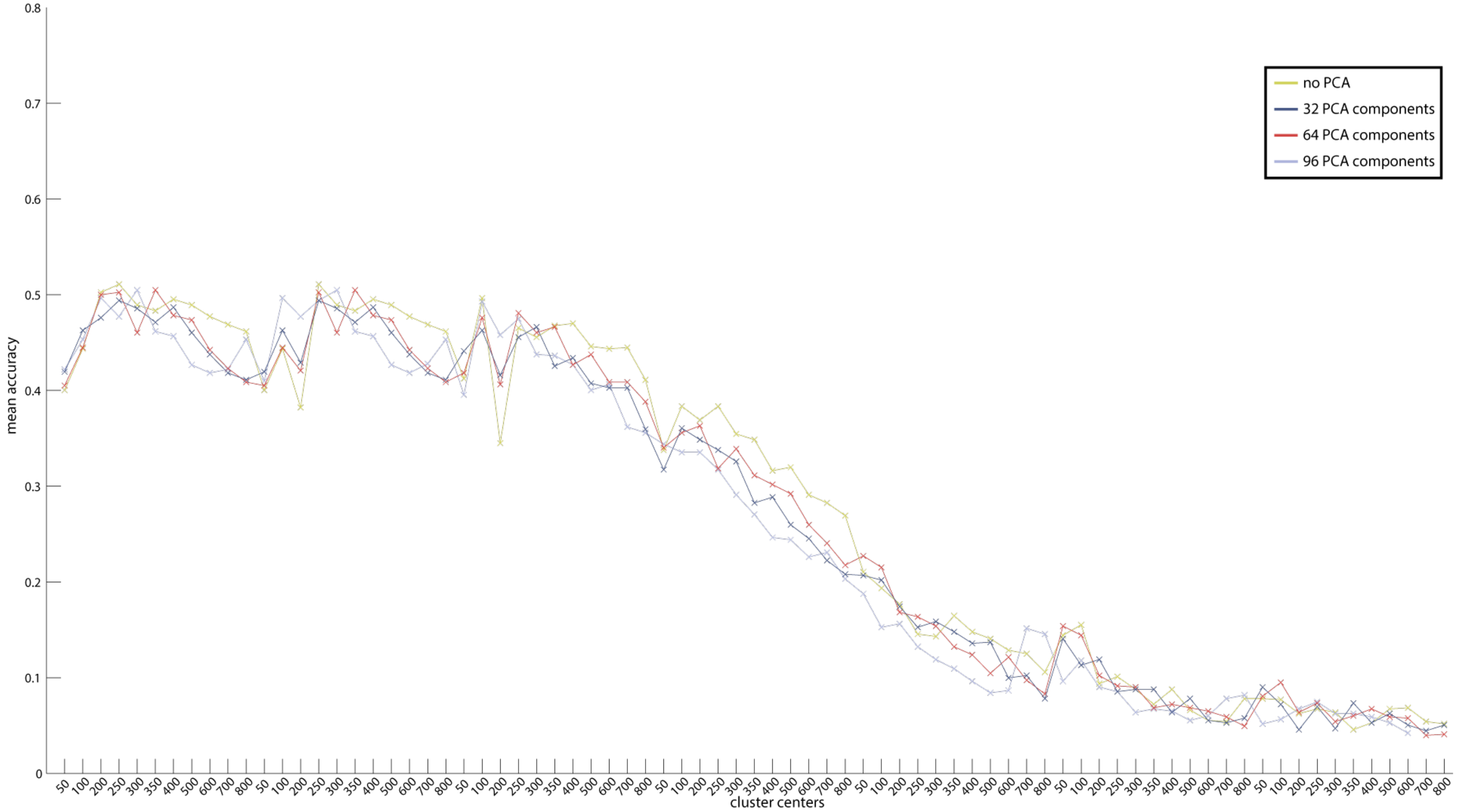


Figure 5.1: Mean accuracy of the hard Top-1, Top-2, Top-3, and Top-4 criterion on the ICDAR 2011 cropped dataset with different parameters using the modified SIFT features.

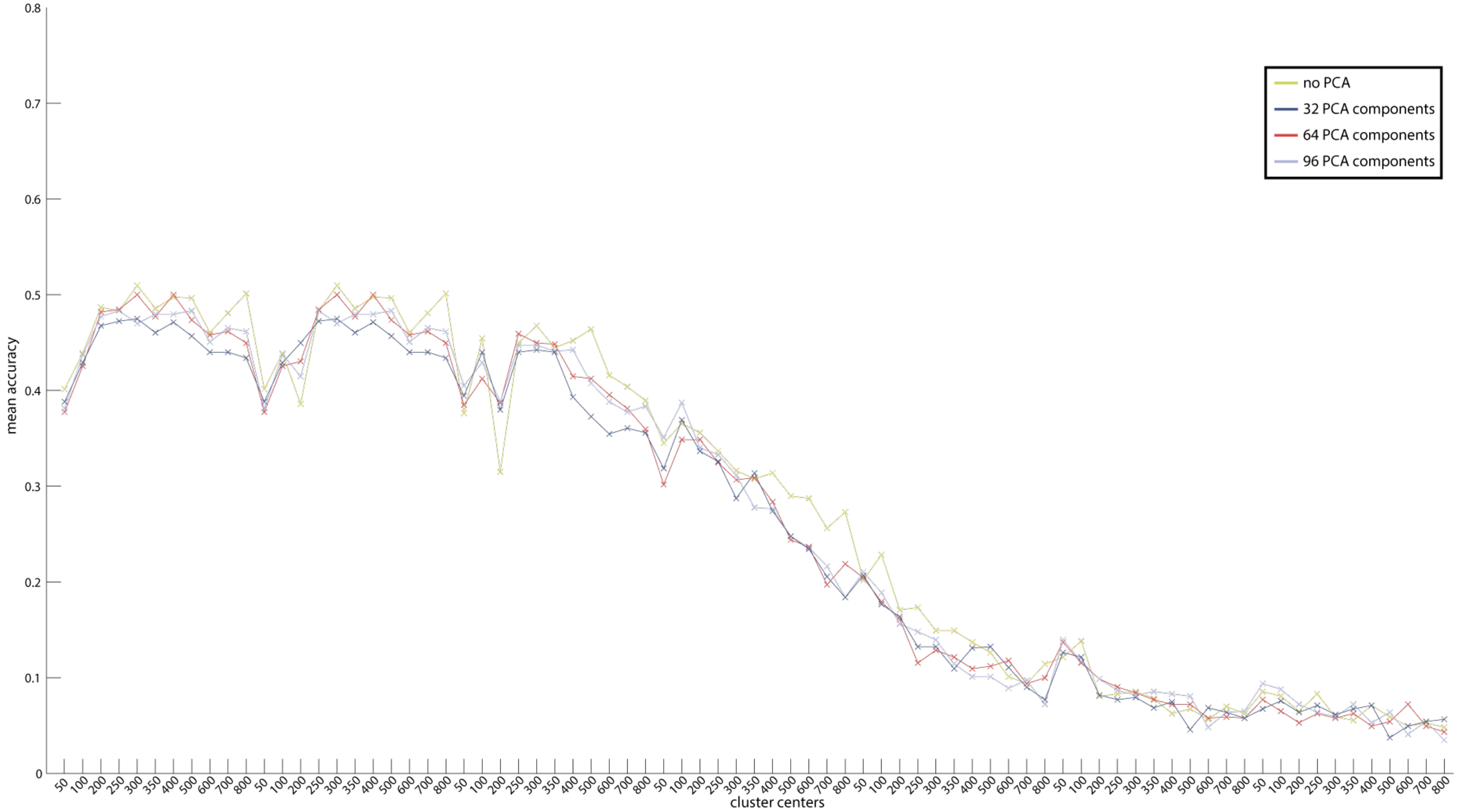


Figure 5.2: Mean accuracy of the hard Top-1, Top-2, Top-3, and Top-4 criterion on the ICDAR 2011 cropped dataset with different parameters using the rotation variant SIFT features.

competition. For the complete dataset all parameter combinations achieve 100% for all criteria. The performance on the cropped dataset is slightly lower than the performance of the other methods. Only for the Top-7 criterion the same performance is achieved. For the Top-1 criterion the highest difference is 4.8%.

complete dataset				
	Top 1	Top 2	Top 5	Top 7
Tsinghua	98.6	**100.0**	**100.0**	**100.0**
MCS-NUST	99.0	99.5	99.5	99.5
Tebessa C	98.6	**100.0**	**100.0**	**100.0**
InVar-250	**100.0**	**100.0**	**100.0**	**100.0**
RotVar-250	**100.0**	**100.0**	**100.0**	**100.0**
InVar-300	**100.0**	**100.0**	**100.0**	**100.0**
RotVar-300	**100.0**	**100.0**	**100.0**	**100.0**
cropped dataset				
	Top 1	Top 2	Top 5	Top 7
Tsinghua	**90.9**	93.8	**98.6**	**99.5**
MCS-NUST	82.2	91.8	96.6	97.6
Tebessa C	87.5	92.8	97.6	**99.5**
InVar-250	85.1	**94.2**	96.6	**99.5**
RotVar-250	85.1	92.3	98.1	99.0
InVar-300	82.7	92.3	96.6	**99.5**
RotVar-300	86.1	92.3	96.6	98.6

Table 5.1: The soft criterion evaluation results on the ICDAR 2011 dataset (in %)

The results for the hard criterion is shown in Table 5.2. On the complete dataset the methods proposed achieve the highest performance. The highest difference between one parameter set and the best performing participant is 6.3% for the Top-5 and 8.2% for the Top-7 criterion. On the cropped dataset the results of the method proposed are lower than the results of the participants. The good performance on the complete dataset and the bad performance on the cropped dataset show that the method does have problems if there are only a few text lines in the document image.

The next evaluation is carried out on the ICFHR 2012 competition dataset. Table 5.3 shows the soft evaluation on this dataset. The accuracy on the Top-1 criterion is lower by 5.7% compared to the other methods. This difference slightly decreases for the other criteria. These results indicate that the method proposed has a weakness when there are too many writers in the dataset. Instead of 26 writers in the previous experiment on the ICDAR dataset, the ICFHR dataset has 100 writers.

In Table 5.4 the results for the hard evaluation are presented. The performance of the method proposed is worse compared to the other methods: For the Top-2 criterion the difference between the accuracy of the best performing parameter set and the highest performing method is nearly 20%, for the Top-3 criterion this difference is even higher with 27.8%. The origin of this performance loss is that the ICFHR dataset contains English and Greek texts. Each writer contributed

complete dataset			
	Top 2	Top 5	Top 7
Tsinghua	95.2	84.1	41.4
MCS-NUST	93.3	78.9	39.9
Tebessa C	**97.1**	81.3	50.0
InVar-250	**97.1**	88.5	51.9
RotVar-250	96.6	88.5	51.4
InVar-300	96.6	89.4	**58.2**
RotVar-300	96.6	**90.4**	47.6
cropped dataset			
	Top 2	Top 5	Top 7
Tsinghua	**79.8**	**48.6**	12.5
MCS-NUST	71.6	35.6	11.1
Tebessa C	76.0	34.1	**14.4**
InVar-250	71.2	39.9	8.2
RotVar-250	69.7	31.3	7.2
InVar-300	69.2	36.5	7.2
RotVar-300	76.4	34.6	6.7

Table 5.2: The hard criterion evaluation results on the ICDAR 2011 dataset (in %)

	Top 1	Top 2	Top 5	Top 10
Tsinghua	92.8	95.8	97.8	98.3
Tebessa A	92.3	96,5	**98.8**	99.0
Tebessa C	**94.5**	**97.3**	**99.3**	**99.3**
InVar-250	84.5	92.0	96.3	97.3
RotVar-250	88.0	93.0	96.0	97.5
InVar-300	87.5	93.0	97.5	98.5
RotVar-300	88.8	92.8	96.8	98.3

Table 5.3: The soft criterion evaluation results on the ICFHR 2012 dataset (in %)

4 pages to the dataset, so for the Top-2 criterion one page written by the same writer in the other language has to be found and for the Top-3 criterion both pages have to be found. The method proposed is no longer able to identify a writer, if not the same alphabet is used (even though the Latin alphabet is derived from the Greek one). Again, the unmodified SIFT features perform better than the rotation variant features.

The next experiment is carried out on the ICDAR 2013 database. The results for this evaluation are shown in Table 5.5. Again, the method proposed has a lower performance than the participants of the competition. For the Top-1 criterion the difference is 9%, for the Top-2 5.9%, for the Top-5 4.3%, and for the Top-10 2.9%. The differences for the hard criterion are higher since the dataset again contains texts written in Greek and in English. The winner of the competition (CS-UMD-a) has the worst performance concerning the hard Top-2 and Top-3 criteria.

	Top 2	Top 3
Tsinghua	51.5	27.3
Tebessa A	57.5	29.3
Tebessa C	**65.0**	**37.8**
InVar-250	35.3	10.0
RotVar-250	32.0	10.0
InVar-300	35.3	9.0
RotVar-300	27.3	8.0

Table 5.4: The hard criterion evaluation results on the ICFHR 2012 dataset (in %)

This result shows once more that the method proposed has difficulties when there are many writers in the dataset.

	soft criterion				hard criterion	
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3
CS-UMD-a	**95.1**	**97.7**	**98.6**	99.1	19.6	7.1
CS-UMD-b	95.0	97.2	**98.6**	**99.2**	20.2	8.4
HIT-ICG	94.8	96.7	98.0	98.3	**63.2**	**36.5**
InVar-250	77.1	84.5	92.2	95.3	31.3	13.0
RotVar-250	84.1	89.9	93.4	95.8	34.0	11.6
InVar-300	83.0	89.6	94.3	96.3	34.3	12.9
RotVar-300	86.1	91.8	94.3	96.4	33.0	11.7

Table 5.5: Evaluation of the soft and hard criteria on the ICDAR 2013 dataset (in %)

Table 5.6 shows the results of the evaluation on the CVL-DB. This time one of the parameter sets has the highest performance for 6 out of 7 criteria. The RotVar-300 parameter set has the best performance for all the soft criteria. The other three parameter sets perform only slightly worse and the difference to the other methods is marginal. For the hard criterion only Tsinghua achieved a better accuracy on the Top-3 criterion. The difference for the Top-4 criterion, which means that all other pages of the same writer have to be ranked first, is 3.8%. These results show that if there is enough text present in the document image the method proposed performs better than other state-of-the-art methods. The CVL-DB contains nearly full written pages written in Latin characters (English and German), whereas the previous datasets contained only a short paragraph of text.

Table 5.7 shows the evaluation with the retrieval criterion on the CVL-DB. It can be seen that again the RotVar-300 parameter set has the best performance on all three criteria, but compared to the other parameter sets and the other methods the improvement is again only marginal. For all evaluations on the CVL-DB, the original SIFT features perform better than the modified features. Since this is also the case for the ICDAR 2013 dataset it can be said, that the rotation variant SIFT features have a better performance if there are more writers in the dataset.

The last experiments are on the Glagolitic database to show that the method proposed is also capable to identify a writer if the handwriting is in a different script and on historic documents.

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
Tsinghua	97.7	98.3	99.0	99.1	95.3	**94.5**	73.0
Tebessa C	97.6	97.9	98.3	98.5	94.3	88.2	73.0
InVar-250	97.0	97.7	98.4	99.0	93.6	86.2	68.8
RotVar-250	**97.9**	98.4	99.0	99.2	95.2	90.1	75.8
InVar-300	97.3	98.1	98.6	99.1	93.7	87.5	71.0
RotVar-300	**97.9**	**98.8**	**99.2**	**99.3**	**95.6**	91.2	**76.8**

Table 5.6: Evaluation results of the soft and hard criteria on the CVL-DB (in %)

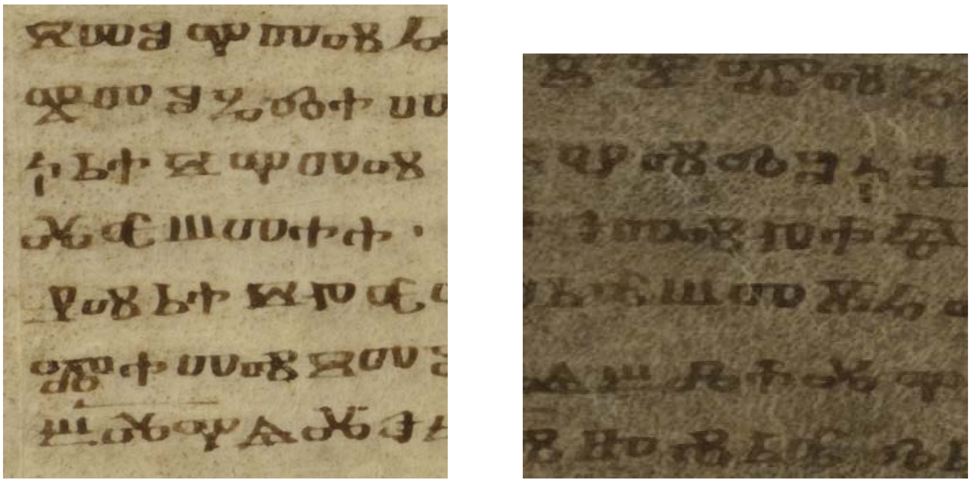
	Top 2	Top 3	Top 4
Tsinghua	96.8	94.5	90.2
Tebessa C	96.1	94.2	90.0
InAr-250	95.7	93.2	88.2
RotVar-250	96.8	95.1	91.1
InVar-300	95.9	93.7	89.1
RotVar-300	**97.2**	**95.7**	**91.6**

Table 5.7: The retrieval criterion evaluation results on the CVL-DB (in %)

The documents of this database have been evaluated using the same parameter sets and the same vocabularies. The visual vocabulary was not generated with Glagolitic characters. Furthermore, when dealing with historic documents noise like background clutter and bleed through ink can occur and thus a second experiment is made with the binarized version of these documents. This is done to eliminate this noise and also to avoid that the method does not depend on the color of the background. The color of the background is influenced by the storage conditions of the documents. Documents of the same codex can have a different background, which is shown in Figure 5.3. In Figure 5.3 (a) the handwriting has a much higher contrast compared to the image portion shown in (b), thus the SIFT descriptors differ for both portions. This effect can be eliminated with binarization.

Table 5.8 shows the results on the Glagolitic dataset with grayscale images. Since the images are copyrighted they are not published and thus no other method has been evaluated on this dataset. The rotation variant SIFT features perform better than the unmodified features. For the Top-1 criterion the performance gain is 3.9 respectively 2.6%. Overall the best performance is achieved when using 250 cluster centers.

The results on the binarized Glagolitic dataset are shown in Table 5.9. The results for all criteria are nearly similar for all four parameter sets. The performance gain when using the modified SIFT features can no longer be observed. Only for the Top-4 criterion the unmodified SIFT features have a higher performance than the unmodified, but the gain is only 0.8%. Compared to the grayscale dataset the performance for the soft evaluation is nearly the same. The performance gain when using binarization as pre-processing step is at maximum 2.5% for the Top-2 criterion. But when looking at the hard criterion, then the performance on the binarized



(a) (b)

Figure 5.3: Two portions of Codex Clozianus. (a) Folio 8 recto was stored at the City Museum in Trento, Italy. (b) Folio 3 verso, stored at the Ferdinandeum Museum in Innsbruck, Austria.

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
InVar-250	95.3	97.8	99.2	99.4	88.9	83.9	80.3
RotVar-250	**97.0**	98.3	**99.4**	**99.7**	**91.4**	**86.1**	**82.0**
InVar-300	95.8	98.3	99.2	**99.7**	90.3	84.2	81.4
RotVar-300	96.7	**98.9**	**99.4**	**99.7**	**91.4**	85.9	80.6

Table 5.8: Evaluation of the soft and hard criteria on the Glagolitc database with grayscale images (in %)

dataset is remarkable higher than on the grayscale dataset. For the Top-2 criterion the maximum difference is 6.9% for the Top-2, 9.2% for the Top-3 and above 20% for the Top-4.

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
InVar-250	**97.8**	98.9	**99.7**	99.7	**95.8**	93.1	**92.2**
RotVar-250	97.5	**99.2**	**99.7**	99.7	**95.8**	**94.2**	91.4
InVar-300	97.5	**99.2**	**99.7**	99.7	**95.8**	93.1	90.8
RotVar-300	96.9	**99.2**	**99.7**	**100.0**	95.0	93.1	91.1

Table 5.9: Evaluation of the soft and hard criteria on the Glagolitc database with binary images (in %)

For all evaluations the best vocabulary size is 250 when using the unmodified SIFT features and 300 when using the rotation variant features. When dealing with a dataset with many writer, like ICDAR 2013 or CVL-DB, then the modified features tend to have a better performance.

5.2 Evaluation of the Fisher Vector Method

The Fisher vector method is evaluated on the same five datasets, namely the cropped and full ICDAR 2011 dataset, then the ICFHR 2012 dataset, the ICDAR 2013 dataset, the CVL-DB, and the Glagolitic Database. Again, the performance is compared with the participants of the competition and for the CVL-DB with the results written in the paper [44]. As described in Section 4.2, four different parameter sets are evaluated. These four sets are:

- INVAR-F20-PCA96-GAUSS10: original SIFT descriptor (rotation invariant), filter size 20, 96 PCA components, and 10 Gaussians.
- ROTVAR-F20-PCA96-GAUSS10: modified SIFT descriptor (rotation variant), filter size 20, 96 PCA components, and 10 Gaussians.
- INVAR-F10-PCA64-GAUSS40: original SIFT descriptor (rotation invariant), filter size 10, 64 PCA components, and 40 Gaussians.
- ROTVAR-F10-PCA64-GAUSS40: modified SIFT descriptor (rotation variant), filter size 10, 64 PCA components, and 40 Gaussians.

The first evaluation is on the ICDAR 2011 dataset. Table 5.10 shows the result of the method proposed with the chosen parameter set on the complete and cropped dataset. It can be seen that on the complete dataset nearly every criterion achieves 100%, except for one parameter set which misses one page. On the cropped dataset the proposed method also outperforms the participants of the competition. The ROTVAR-F20-PCA96-GAUSS10 parameter set achieves best performance for all criteria, but also the other parameter sets perform better than the competitors on about the half of the criteria. The highest performance gain in comparison with the other methods is for the Top 2 criterion with 4.8%. Also the proposed method, with its best parameter set, is the only one which achieves 100% for the Top-5 and Top-7 criteria. In Table 5.11 the results of the evaluation on both datasets are shown. Again, for each criterion the proposed method performs best, but this time with multiple parameter sets. The best parameter set on the complete dataset is INVAR-F10-PCA64-GAUSS40 although it has only the second best performance for the Top-1 criterion. For the Top-5 and Top-7 criterion it performs best with 92.3% respectively 64.4%. The performance gain for the Top-7 criterion, compared to the competitors, is remarkable with 14.4% but also for the Top-5 criterion the performance gain is 8.2%. On the cropped dataset the best performance is achieved with the ROTVAR-F20-PCA96-GAUSS10 parameter set. For the Top-2 criterion the result of the rotation invariant features with the same filter size, Gaussians, and PCA components is better, but for the other two criteria the accuracy is higher by 2.4 respectively 2.9%. Compared to the participants of the competition, the results are outperformed by 7,2% for the Top-1 criterion, 14,9% for the Top 5 criterion and 10.6% for the Top-7 criterion.

The second evaluation is carried out on the ICFHR 2012 dataset. The results for the soft criterion are shown in Table 5.12. Except for the Top-5 criterion at least one of the parameter sets performed best. The overall highest accuracy was achieved using the ROTVAR-F10-PCA64-GAUSS40 parameter set, but the difference to the other sets is only slight, especially for the Top-5 and Top-10 criteria.

complete dataset				
	Top 1	Top 2	Top 5	Top 7
Tsinghua	98.6	**100.0**	**100.0**	**100.0**
MCS-NUST	99.0	99.5	99.5	99.5
Tebessa C	98.6	**100.0**	**100.0**	**100.0**
InVar-F20-Pca96-Gauss10	99.5	99.5	99.5	**100.0**
RotVar-F20-Pca96-Gauss10	**100.0**	**100.0**	**100.0**	**100.0**
InVar-F10-Pca64-Gauss40	**100.0**	**100.0**	**100.0**	**100.0**
RotVar-F10-Pca64-Gauss40	**100.0**	**100.0**	**100.0**	**100.0**
cropped dataset				
	Top 1	Top 2	Top 5	Top 7
Tsinghua	90.9	93.8	98.6	99.5
MCS-NUST	82.2	91.8	96.6	97.6
Tebessa C	87.5	92.8	97.6	99.5
InVar-F20-Pca96-Gauss10	92.8	95.7	97.6	99.0
RotVar-F20-Pca96-Gauss10	**94.2**	**98.6**	**100.0**	**100.0**
InVar-F10-Pca64-Gauss40	90.4	93.3	97.6	99.0
RotVar-F10-Pca64-Gauss40	93.8	95.2	98.6	99.0

Table 5.10: The soft criterion evaluation results on the ICDAR 2011 dataset (in %)

complete dataset			
	Top 2	Top 5	Top 7
Tsinghua	95.2	84.1	41.4
MCS-NUST	93.3	78.9	39.9
Tebessa C	97.1	81.3	50.0
InVar-F20-Pca96-Gauss10	**98.6**	90.9	61.5
RotVar-F20-Pca96-Gauss10	97.6	**92.3**	56.7
InVar-F10-Pca64-Gauss40	98.1	**92.3**	**64.4**
RotVar-F10-Pca64-Gauss40	96.2	**92.3**	60.1
cropped dataset			
	Top 2	Top 5	Top 7
Tsinghua	79.8	48.6	12.5
MCS-NUST	71.6	35.6	11.1
Tebessa C	76.0	34.1	14.4
InVar-F20-Pca96-Gauss10	**87.0**	61.1	20.7
RotVar-F20-Pca96-Gauss10	86.1	**63.5**	**25.0**
InVar-F10-Pca64-Gauss40	84.6	61.5	17.8
RotVar-F10-Pca64-Gauss40	86.1	61.5	21.2

Table 5.11: The hard criterion evaluation results on the ICDAR 2011 dataset (in %)

	Top 1	Top 2	Top 5	Top 10
Tsinghua	92.8	95.8	97.8	98.3
Tebessa A	92.3	96,5	98.8	99.0
Tebessa C	94.5	97.3	**99.3**	99.3
InVar-F20-Pca96-Gauss10	93.0	96.8	98.5	99.0
RotVar-F20-Pca96-Gauss10	95.5	97.3	99.0	**99.5**
InVar-F10-Pca64-Gauss40	95.3	96.8	98.3	99.3
RotVar-F10-Pca64-Gauss40	**96.3**	**98.0**	99.0	99.0

Table 5.12: The soft criterion evaluation results on the ICFHR 2012 dataset (in %)

Table 5.13 shows the evaluation of the hard criterion. This time the method proposed performs worse than the best competitor. The best accuracy is achieved using the InVar-F10-Pca64-Gauss40 parameter set, but still the gap to *Tebessa C* is 18% for the Top-2 and 17.3% for the Top-3 criterion. The reason for this result is the same like for the BOW method: the ICFHR 2012 dataset consists of four pages for each writer, two written in English and two written in Greek. The method proposed is not able to identify the writer correctly when the reference document and the query document are written with different alphabets. So the performance for the Top-1 criterion is good where one page with the same alphabet can still be found. For the Top-2 criterion at least one page written in the other alphabet has to be found and for the Top-3 criterion both other pages have to be found. This can also be seen at the values in Table 5.14. In this evaluation the reference document is only compared with the pages written in the same language and the corresponding page of the same writer has to be found. The proposed method performs best for each criterion for both languages. For the Top-1 criterion, which is finding the missing document in the most similar page, the accuracy is raised by 2% when using the RotVar-F10-Pca64-Gauss40 parameter set. When comparing the soft result for each language and the soft results for the complete dataset only the result for the Top-1 and Top-2 criterion is better on the complete dataset than on either one of the only English or Greek dataset. This means, that for these criteria at least one page written in the other alphabet was found. These results show that the method proposed is able to do the identification task on different alphabets then used for generating the visual vocabulary, but is not able to handle multiple alphabets in the evaluation dataset. Remarkable for the hard criterion on the complete dataset is, that the rotation invariant SIFT features perform up to 7.5% better on the Top-1 and 4.2% better on the Top-3 criterion than the modified SIFT feature. One explanation for this behavior is that the upper and lower contour of the handwriting differs for each writer when writing in a different alphabet and thus the modification has a bad influence on the performance of the method.

Next evaluation has been carried out on the ICDAR 2013 dataset and is shown in Table 5.15. Again, for the soft criterion the method proposed performed best. This time the parameter set RotVar-F10-Pca64-Gauss40 has the best performance on all four criteria. The distance to the winner of the competition, CS-UMD-a, is only 1.7% on the Top-1 criterion and 0.3% for the other criteria. For the hard criteria the performance is again worse than one of the participants. The performance of all parameter sets is about 20% worse for the Top-2 criterion and about 13% for the Top-3 criterion. The reason for this is the same like for the ICFHR 2012 dataset.

	Top 2	Top 3
Tsinghua	51.5	27.3
Tebessa A	57.5	29.3
Tebessa C	**65.0**	**37.8**
InVar-F20-Pca96-Gauss10	43.5	16.8
RotVar-F20-Pca96-Gauss10	37.8	16.0
InVar-F10-Pca64-Gauss40	47.0	20.5
RotVar-F10-Pca64-Gauss40	39.5	16.3

Table 5.13: The hard criterion evaluation results on the ICFHR 2012 dataset (in %)

only English pages				
	Top 1	Top 2	Top 5	Top 10
Tsinghua	94,0	94,5	95,5	98,0
Tebessa A	89,5	96,0	97,0	98,5
Tebessa C	91,5	95,5	97,5	98,0
InVar-F20-Pca96-Gauss10	91.0	95.5	97.5	98.5
RotVar-F20-Pca96-Gauss10	95.5	**97.5**	**99.0**	**99.0**
InVar-F10-Pca64-Gauss40	94.0	95.5	97.5	98.5
RotVar-F10-Pca64-Gauss40	**96.0**	97.0	98.0	98.5
only Greek pages				
Tsinghua	90,0	94,0	98,5	99,0
Tebessa A	92,0	95,0	98,5	99,0
Tebessa C	93,5	97,0	99,5	99,5
InVar-F20-Pca96-Gauss10	93.5	96.5	98.5	99.0
RotVar-F20-Pca96-Gauss10	**95.5**	97.0	99.0	**100.0**
InVar-F10-Pca64-Gauss40	94.5	96.5	99.5	**100.0**
RotVar-F10-Pca64-Gauss40	**95.5**	**98.5**	**100.0**	**100.0**

Table 5.14: The soft criterion evaluation results for each language on the ICFHR 2012 dataset (in %)

Again, the writers who contributed to this dataset had to copy four pages, two in English and two in Greek. The method proposed is dependent on the same alphabet and thus the performance drops. The winner of the competition achieves even worse result since their method, as described in Section 2.2, is generating the features directly on the shape of the characters and thus it is highly dependent on the alphabet.

Table 5.16 shows the evaluation results on the CVL-DB. For all criteria, soft and hard, the proposed method has the highest accuracy. The difference between the two parameter sets for the rotation variant and rotation invariant SIFT features is only marginal. There is also only a slight difference between the proposed method and the other methods for all criteria except for the Top-4 hard criterion, here the performance of the method proposed is 12.7% higher. Table 5.17 shows the evaluation of the retrieval criterion. The method proposed outperforms the other

	soft criterion				hard criterion	
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3
CS-UMD-a	95.1	97.7	98.6	99.1	19.6	7.1
CS-UMD-b	95.0	97.2	98.6	99.2	20.2	8.4
HIT-ICG	94.8	96.7	98.0	98.3	**63.2**	**36.5**
InVar-F20-Pca96-Gauss10	92.3	95.1	97.5	98.9	44.7	22.0
RotVar-F20-Pca96-Gauss10	95.3	96.9	**98.9**	99.3	44.2	22.0
InVar-F10-Pca64-Gauss40	93.2	96.2	98.3	99.0	44.3	23.5
RotVar-F10-Pca64-Gauss40	**96.8**	**98.0**	**98.9**	**99.4**	42.3	23.1

Table 5.15: Evaluation of the soft and hard criteria on the ICDAR 2013 dataset (in %)

methods, but it can be seen that 3 out of 4 parameter sets have exactly the same performance on all three criteria. This means that, at least on this dataset, all datasets can be used for retrieval and the different parameters deliver only a slightly varied sorting according to the similarity of the handwriting.

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
Tsinghua	97.7	98.3	99.0	99.1	95.3	94.5	73.0
Tebessa C	97.6	97.9	98.3	98.5	94.3	88.2	73.9
InVar-F20-Pca96-Gauss10	98.2	98.7	99.0	99.2	96.2	93.2	82.3
RotVar-F20-Pca96-Gauss10	98.8	99.1	99.2	99.2	97.4	94.9	86.6
InVar-F10-Pca64-Gauss40	98.9	99.0	99.3	99.5	96.9	94.0	85.7
RotVar-F10-Pca64-Gauss40	**99.1**	**99.3**	**99.5**	**99.6**	**98.0**	**96.1**	**89.6**

Table 5.16: Evaluation results of the soft and hard criteria on the CVL-DB (in %)

	Top 2	Top 3	Top 4
Tsinghua	96.8	94.5	90.2
Tebessa C	96.1	94.2	90.0
InVar-F20-Pca96-Gauss10	97.4	96.3	93.3
RotVar-F20-Pca96-Gauss10	**98.2**	**97.4**	**95.0**
InVar-F10-Pca64-Gauss40	**98.2**	**97.4**	**95.0**
RotVar-F10-Pca64-Gauss40	**98.2**	**97.4**	**95.0**

Table 5.17: The retrieval criterion evaluation results on the CVL-DB (in %)

The last experiments are carried out on the Glagolitic dataset. For reasons which have been already explained in Section 5.1 the evaluation is also carried out on the binarized dataset. Like for the BOW approach the same parameter set is used and again no new GMMs are generated with Glagolitic characters. Table 5.18 shows the evaluation results on this dataset on the grayscale images. Only the results of the method proposed are listed due to copyright issues of the dataset no other methods have been evaluated on this dataset. It can be seen that, except for the Top-1 criterion, the rotation variant SIFT features have a higher performance than the unmodified features. The ROTVAR-F10-PCA64-GAUSS40 has the overall best performance, for the Top-1 criterion it is only 0.5% worse than the accuracy of the best performing parameter set. In general, the parameter with the higher number Gaussians and less PCA components have a higher accuracy than the other two parameter sets. This can be for two reasons: first the feature space is described in more detail with 40 Gaussians, which may be necessary for Glagolitic characters and second, due to the PCA the information of the SIFT features is decreased which can reduce the influence of the background. Table 5.19 shows the result on the binary images of the dataset. It can be seen that again the rotation variant SIFT features perform better, but this time the two different parameter sets perform nearly equally good. This means that for the evaluation on the gray scale images the influence of the background was reduced using the PCA.

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
INVAR-F20- -PCA96-GAUSS10	94.2	97.5	99.2	99.4	89.5	82.5	79.8
ROTVAR-F20- -PCA96-GAUSS10	95.3	98.1	99.4	99.7	91.4	86.1	83.7
INVAR-F10- -PCA64-GAUSS40	**97.2**	98.6	99.7	99.7	**94.7**	87.8	82.0
ROTVAR-F10- -PCA64-GAUSS40	96.7	**98.9**	**99.7**	**99.7**	**94.7**	**90.6**	**87.3**

Table 5.18: Evaluation of the soft and hard criteria on the Glagolitc database with grayscale images (in %)

Since the distribution of the number of pages for each writer is not equal and Writer 1 has by far the largest contingent of documents in the dataset an additional experiment is carried out. This experiment shows that the method is not only performing well for Writer 1 (where it achieves 100% as experiments showed), but also for the other writers. This time the documents from Writer 1 are skipped as reference documents but remain in the dataset. So the documents of the other writers still have the possibility that they are assigned to Writer 1 because the document with the smallest distance originate from Writer 1. The evaluation on the dataset with grayscale images can be seen in Table 5.20. The performance is dropping for each criterion, but only for 3.2% on maximum for the best performing parameter set. It is remarkable that the rotation variant descriptors this time perform up to 18% better on the hard Top-3 and Top-4 criteria. This means that the rotation variance is necessary for a retrieval, since only the last criteria are affected. Table 5.21 shows the results on the dataset with binary images where the pages of Writer 1 has been skipped as reference documents. Again, compared to the grayscale dataset the

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
InVar-F20- -Pca96-Gauss10	98.6	99.7	**100.0**	**100.0**	96.7	95.3	93.6
RotVar-F20- -Pca96-Gauss10	**99.4**	99.7	**100.0**	**100.0**	98.3	96.7	95.6
InVar-F10- -Pca64-Gauss40	98.3	**100.0**	**100.0**	**100.0**	97.2	96.1	94.7
RotVar-F10- -Pca64-Gauss40	**99.4**	**100.0**	**100.0**	**100.0**	**98.6**	**96.9**	**95.8**

Table 5.19: Evaluation of the soft and hard criteria on the Glagolitic database with binary images (in %)

overall performance of all parameter sets increases. This time the accuracy of the rotation variant and invariant parameter sets are again similar, but still the rotation variant sets outperform the unmodified SIFT features.

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
InVar-F20- -Pca96-Gauss10	87.7	94.8	98.7	99.4	77.3	61.7	55.8
RotVar-F20- -Pca96-Gauss10	90.3	96.1	99.4	**100.0**	81.2	68.8	63.0
InVar-F10- -Pca64-Gauss40	**94.2**	97.4	**100.0**	**100.0**	**89.0**	72.7	59.7
RotVar-F10- -Pca64-Gauss40	93.5	**98.1**	**100.0**	**100.0**	**89.0**	**79.2**	**72.1**

Table 5.20: Evaluation of the soft and hard criteria on the Glagolitc database with grayscale images (in %) with skipping the pages of Writer 1 as reference document

The evaluation on all datasets shows, that the modified SIFT features perform better than the unmodified features when using the Fisher vector. 40 Gaussians have a better performance than 10 Gaussians and when using 40 Gaussians the filter size is set to 10. For the unmodified SIFT features also the 40 Gaussians have the best performance, also with a filter size of 10.

5.3 Evaluation of the Convolutional Neural Network

As The CNN was trained on the IAM-DB, since sufficient trainings patches can be extracted from this dataset. The first evaluation is on the ICDAR 2011 dataset. The values for the hard criterion are, like in the competition, 2, 3, and 4 and for the soft criterion 1, 2, 5, and 7. The result of the soft criterion on the complete dataset and the cropped dataset are presented in Table

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
InVar-F20- -Pca96-Gauss10	97.4	99.3	**100.0**	**100.0**	93.5	91.5	87.6
RotVar-F20- -Pca96-Gauss10	98.7	99.3	**100.0**	**100.0**	**96.7**	92.8	90.2
InVar-F10- -Pca64-Gauss40	96.1	**100.0**	**100.0**	**100.0**	94.8	92.2	89.5
RotVar-F10- -Pca64-Gauss40	98.7	**100.0**	**100.0**	**100.0**	**96.7**	**93.5**	**90.8**

Table 5.21: Evaluation of the soft and hard criteria on the Glagolitc database with binary images (in %) with skipping the pages of Writer 1 as reference document

5.22 the other values are the best three participants of this competition. It can be seen that on the complete dataset all methods have a performance above 98%. The proposed method with the CNN has still the best performance with 99.5% at the Top-1 criterion, which is actually the identification of the writer. The explanations for the performance of all methods are that there are only 26 writers in the dataset and the text which has to be copied by the participants consists of multiple lines. So nearly all values in this part of the table are 100% when regarding the Top-2 to Top-7 results. The influence of the text length can be seen in the second part of the table. The cropped dataset consists only of one or two lines of the complete pages. The performance for all participants drop, but still the proposed method has the best performance on the Top-1 criterion. For the Top-5 criterion the method of Tsinghua has the best performance and the performance on the Top-7 criterion are equally high for two participants and the proposed method.

complete dataset				
	Top 1	Top 2	Top 5	Top 7
Tsinghua	98.6	**100.0**	**100.0**	**100.0**
MCS-NUST	99.0	99.5	99.5	99.5
Tebessa C	98.6	**100.0**	**100.0**	**100.0**
proposed method	**99.5**	**100.0**	**100.0**	**100.0**
cropped dataset				
	Top 1	Top 2	Top 5	Top 7
Tsinghua	90.9	93.8	**98.6**	**99.5**
MCS-NUST	82.2	91.8	96.6	97.6
Tebessa C	87.5	92.8	97.6	**99.5**
proposed method	**94.7**	**97.6**	98.1	**99.5**

Table 5.22: The soft criterion evaluation results on the ICDAR 2011 dataset (in %)

The evaluation of the hard criterion on this dataset is shown in Table 5.23, again for the complete and also for the cropped dataset. On the complete dataset the proposed method has the best performance of all four methods. Still the accuracies are above 93%, which originates from the number of writers and the amount of text in the document images. For the Top-7 criterion, which means finding all other 7 pages of the writer of the reference document the performance is still 52.4%. On the cropped dataset the performance of all methods decreases drastically. For the Top-5 criterion the performance drops to 53.8% and for the Top-7 to 14.4%. The Top-7 criterion is the only one where the proposed method has not the best performance.

complete dataset			
	Top 2	Top 5	Top 7
Tsinghua	95.2	84.1	41.4
MCS-NUST	93.3	78.9	39.9
Tebessa C	97.1	81.3	50.0
proposed method	**98.6**	**87.0**	**52.4**
cropped dataset			
	Top 2	Top 5	Top 7
Tsinghua	79.8	48.6	12.5
MCS-NUST	71.6	35.6	11.1
Tebessa C	76.0	34.1	**14.4**
proposed method	**84.6**	**53.8**	10.1

Table 5.23: The hard criterion evaluation results on the ICDAR 2011 dataset (in %)

The next evaluation has been carried out on the ICDAR 2013 dataset. 250 writers contributed to this dataset and also there are only four lines of text present in the document image. Table 5.24 shows the evaluation of all criteria on this dataset. It can be seen that the proposed method is unable to compete with the methods of the participants of the contest. One of the problems of the proposed method on this dataset is that the line segmentation does not return perfect results, ascender or descenders are cut off however they are discriminative features for writer identification. Also normalization problems are the cause of these results, which are also influenced by the performance of the line segmentation. For the Top-3 criterion the task is to find all the images of the same writer. Since all writers have two English and two Greek texts in the dataset, this means that also both documents written in the other language have to be found. “CS-UMD-a”, which is actually the winning method, has only a performance of 7.1% for this criterion whereas “HIT-ICG” has a performance of 36.5%. This is because “CS-UMD-a” generates a codebook with exemplar letters for each writer. Since English and Greek have a different alphabet the identification rate of this method drops dramatically when a text is written in a language with a different alphabet.

The last evaluation of the method is on the CVL-DB. The results for the hard and soft criterion are shown in Table 5.25. For all the soft criteria the proposed method performed best on this dataset. Since the texts which have to be copied by the writers for this dataset consists of multiple text lines, all the methods have a performance of more than 97%. For the hard criterion the proposed method perform best for the Top-2 and the Top-4 criterion, the improvement here is 6.9%. The task for Top-4 is to find all other four pages of the same writer as the reference document in the dataset.

	soft criterion				hard criterion	
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3
CS-UMD-a	**95.1**	**97.7**	98.6	99.1	19.6	7.1
CS-UMD-b	95.0	97.2	**98.6**	**99.2**	20.2	8.4
HIT-ICG	94.8	96.7	98.0	98.3	**63.2**	**36.5**
proposed method	88.5	92.2	96.0	98.3	40.5	15.8

Table 5.24: Evaluation of the soft and hard criteria on the ICDAR 2013 dataset (in %)

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
Tsinghua	97.7	98.3	99.0	99.1	95.3	**94.5**	73.0
Tebessa C	97.6	97.9	98.3	98.5	94.3	88.2	73.0
proposed method	**98.9**	**99.0**	**99.3**	**99.5**	**97.6**	93.3	**79.9**

Table 5.25: Evaluation results of the soft and hard criteria on the CVL-DB (in %)

On this dataset also the retrieval is evaluated, because values for comparison are listened in [44]. These results are shown in Table 5.26. The proposed method achieves again higher retrieval rates than the other methods. 93.3% of the Top-4 retrieved documents belong to the same writer as the reference document.

	Top 2	Top 3	Top 4
Tsinghua	96.8	94.5	90.2
Tebessa C	96.1	94.2	90.0
proposed method	**98.3**	**96.9**	**93.3**

Table 5.26: The retrieval criterion evaluation results on the CVL-DB (in %)

Furthermore, the effect on the results of the rotation of the training data as described in Section 4.3 has been evaluated. The CNN has been trained without artificially enlarging the trainings dataset. These results are compared against the results when training on the enlarged dataset. The evaluations on the ICDAR 2011 datasets are shown in Table 5.27 and 5.28. It can be seen that especially the performance on the cropped dataset drops significantly. This can be explained by the amount of text in the document image. The fewer characters are present on the page the more precise description of the writer has to be encoded in the features. When training without the rotation of the image dataset, the CNN learns the rotation of the character as a feature for a specific writer. If in the test dataset multiple writers have a nearly equal slant the system may mix up these two writers and thus the precision is lower. Also the number of training images has an effect on the performance of the system. When training on the enlarged dataset the CNN is able to find features which are suited better for the task of writer identification. For the soft criterion on the complete dataset the performance is only slightly better when using the enlarged trainings set because the performance is already above 97% and there are enough characters on the complete page to find another page of the same writer. This can be seen in the results of the hard criterion, where all pages have to be written by the same writer. The performance drops rapidly if more pages have to be found. So finding one page of the same writer can still be handled when training on the unrotated dataset, but when trying to find more or even all pages of one writer the system does not return reliable results.

complete dataset				
	Top 1	Top 2	Top 5	Top 7
with rotation	99.5	**100.0**	**100.0**	**100.0**
without rotation	97.6	98.6	**100.0**	**100.0**
cropped dataset				
	Top 1	Top 2	Top 5	Top 7
with rotation	94.7	97.6	98.1	99.5
without rotation	79.8	87.0	94.2	98.1

Table 5.27: Comparison of the effect of the rotation on the trainings dataset on the soft criterion of the ICDAR 2011 dataset (in %)

complete dataset			
	Top 2	Top 5	Top 7
with rotation	98.6	87.0	52.4
without rotation	93.8	72.1	33.2
cropped dataset			
	Top 2	Top 5	Top 7
with rotation	84.6	53.8	10.1
without rotation	58.7	18.8	1.9

Table 5.28: Comparison of the effect of the rotation on the trainings dataset on the hard criterion of the ICDAR 2011 dataset (in %)

Table 5.29 shows the comparison of the performance of the different trainings sets on the ICDAR 2013 dataset. Again, the performance is dropping dramatically, however this time for both criteria. Whereas on the complete pages on the ICDAR 2011 dataset the maximal performance loss of the soft criterion is 1,9% on the ICDAR 2013 dataset the performance loss for the Top-1 criterion is 19.9%. This is due to the number of writers present in the dataset. For the hard criterion the accuracy is approximately only two thirds compared to a training on the rotated dataset.

	soft criterion				hard criterion	
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3
with rotation	88.5	92.2	96.0	98.3	40.5	15.8
without rotation	68.6	77.7	87.3	93.0	25.8	9.2

Table 5.29: Comparison of the effect of the rotation on the trainings dataset on the ICDAR 2013 dataset (in %)

The evaluation of the CVL-DB is shown in Table 5.30. Again, when using the soft criterion the performance drops nearly by 12%. Especially when using the Top-4 hard criterion the performance loss is more than 50%. For this criterion all pages of the same writer have to be found. Since the ICDAR 2011 and 2013 dataset contain Greek and English texts the difference of the performance on the CVL-DB of the system when trained with both datasets is not that high compared to the other datasets. In the CVL-DB only Latin scripts are used and thus the performance of the system is almost 80%. When training with the unrotated dataset the CNN is no longer able to discriminate between the different writers for all texts and thus the performance is dropping.

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
with rotation	98.9	99.0	99.3	99.5	97.6	93.3	79.9
without rotation	86.6	91.1	95.4	96.9	72.5	51.0	24.1

Table 5.30: Comparison of the effect of the rotation on the trainings dataset on the CVL-DB (in %)

5.4 Comparison of the Results

In this section the results of the three different methods are compared to each other. This is also done for each dataset presented in Section 2.1. For the BOW and for the Fisher vector the best parameter set is taken for the unmodified respectively modified SIFT features are taken. For the BOW approach the cluster size is set to 250 for the original features and 300 when using the rotation variant features. For the Fisher vector method 40 Gaussians are used for both features. The CNN method is trained using also the rotated images.

The first dataset is the ICDAR 2011 and the results from the different methods are shown in Table 5.31. For the soft criterion on the complete dataset nearly every criteria can be full filled by 100%. On the cropped dataset the Fisher vector method and the CNN method are best performing. The BOW method can only keep up on one criterion, the Top-7.

Tabe 5.32 shows the comparison of the results using the hard criterion. Again, the Fisher vector and the CNN approach performed best on this dataset. On the complete dataset the unmodified features perform better than the modified features. Since only 26 writers have contributed to this dataset, the original SIFT features have a better performance. The CNN method is able to keep up for the Top-2 criteria, but then is falling back in contrast to the Fisher vector approach.

The comparison of the results on the ICFHR dataset is shown in Table 5.33 and Table 5.34. There are no results available for the CNN approach. For the soft criterion the rotation variant feature perform better. The Fisher vector approach outperforms the BOW method.

For the hard evaluation the unmodified SIFT feature have a higher accuracy. The reason for this is the mixture between Greek and Latin characters, since documents written in the other language have to be found. The Fisher Vector has a much higher performance than the BOW

complete dataset				
	Top 1	Top 2	Top 5	Top 7
InVar-250	**100.0**	**100.0**	**100.0**	**100.0**
RotVar-300	**100.0**	**100.0**	**100.0**	**100.0**
InVar-F10-Pca64-Gauss40	**100.0**	**100.0**	**100.0**	**100.0**
RotVar-F10-Pca64-Gauss40	**100.0**	**100.0**	**100.0**	**100.0**
CNN	99.5	**100.0**	**100.0**	**100.0**
cropped dataset				
	Top 1	Top 2	Top 5	Top 7
InVar-250	85.1	94.2	96.6	**99.5**
RotVar-300	86.1	92.3	96.6	98.6
InVar-F10-Pca64-Gauss40	90.4	93.3	97.6	99.0
RotVar-F10-Pca64-Gauss40	93.8	95.2	**98.6**	99.0
CNN	**94.7**	**97.6**	98.1	**99.5**

Table 5.31: The soft criterion evaluation results on the ICDAR 2011 dataset (in %)

complete dataset			
	Top 2	Top 5	Top 7
InVar-250	97.1	88.5	51.9
RotVar-300	96.6	90.4	47.6
InVar-F10-Pca64-Gauss40	98.1	**92.3**	**64.4**
RotVar-F10-Pca64-Gauss40	96.2	**92.3**	60.1
CNN	**98.6**	87.0	52.4
cropped dataset			
	Top 2	Top 5	Top 7
InVar-250	71.2	39.9	8.2
RotVar-300	76.4	34.6	6.7
InVar-F10-Pca64-Gauss40	84.6	**61.5**	17.8
RotVar-F10-Pca64-Gauss40	**86.1**	**61.5**	**21.2**
CNN	84.6	53.8	10.1

Table 5.32: The hard criterion evaluation results on the ICDAR 2011 dataset (in %)

	Top 1	Top 2	Top 5	Top 10
InVar-250	84.5	92.0	96.3	97.3
RotVar-300	88.8	92.8	96.8	98.3
InVar-F10-Pca64-Gauss40	95.3	96.8	98.3	99.3
RotVar-F10-Pca64-Gauss40	**96.3**	**98.0**	**99.0**	**99.0**

Table 5.33: The soft criterion evaluation results on the ICFHR 2012 dataset (in %)

approach. For the Top-2 criterion the difference is nearly 22% and for the Top-3 criterion it is still 10.5%.

	Top 2	Top 3
InVar-250	35.3	10.0
RotVar-300	27.3	8.0
InVar-F10-Pca64-Gauss40	**47.0**	**20.5**
RotVar-F10-Pca64-Gauss40	39.5	16.3

Table 5.34: The hard criterion evaluation results on the ICFHR 2012 dataset (in %)

The results on the ICDAR 2013 are listened in Table 5.35. Again, for the soft evaluation the Fisher vector approach with the parameter set RotVar-F10-Pca64-Gauss40 has the best performance and for the hard evaluation the parameter set InVar-F10-Pca64-Gauss40 performs best. Again, this has to do with the two different alphabets which are mixed up in this dataset. The CNN method has a higher performance than the BOW approach. For the hard criterion the accuracy of the Fisher vector are ca. 10-13% higher than the accuracy of the BOW.

	soft criterion				hard criterion	
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3
InVar-250	77.1	84.5	92.2	95.3	31.3	13.0
RotVar-300	86.1	91.8	94.3	96.4	33.0	11.7
InVar-F10-Pca64-Gauss40	93.2	96.2	98.3	99.0	**44.3**	**23.5**
RotVar-F10-Pca64-Gauss40	**96.8**	**98.0**	**98.9**	**99.4**	42.3	23.1
CNN	88.5	92.2	96.0	98.3	40.5	15.8

Table 5.35: Evaluation of the soft and hard criteria on the ICDAR 2013 dataset (in %)

For the CVL-DB the results of the different methods look similar and are presented in Table 5.36. The only difference is that this time also the rotation variant features perform best for the hard criterion. The CNN has also a higher accuracy than the BOW methods, but the difference is very small. For the Top-4 hard criterion, which is finding all other four pages written by the same writer, the method using the Fisher vector has a 9.7% higher accuracy than the CNN and 12.8% higher than the BOW.

For the retrieval criteria the results are slightly different and are shown in Table 5.37. Both Fisher vector approaches perform equal, but the CNN method is able to perform slightly better

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
InVar-250	97.0	97.7	98.4	99.0	93.6	86.2	68.8
RotVar-300	97.9	98.8	99.2	99.3	95.6	91.2	76.8
InVar-F10- -Pca64-Gauss40	98.9	99.0	99.3	99.5	96.9	94.0	85.7
RotVar-F10- -Pca64-Gauss40	**99.1**	**99.3**	**99.5**	**99.6**	**98.0**	**96.1**	**89.6**
CNN	98.9	99.0	99.3	99.5	97.6	93.3	79.9

Table 5.36: Evaluation results of the soft and hard criteria on the CVL-DB (in %)

on the Top-2 criterion. The difference to the BOW method is no longer that high.

	Top 2	Top 3	Top 4
Inar-250	95.7	93.2	88.2
RotVar-300	97.2	95.7	91.6
InVar-F10-Pca64-Gauss40	98.2	**97.4**	**95.0**
RotVar-F10-Pca64-Gauss40	98.2	**97.4**	**95.0**
CNN	**98.3**	96.9	93.3

Table 5.37: The retrieval criterion evaluation results on the CVL-DB (in %)

The Glagolitic database has only been evaluated with the Fisher vector and the BOW method, since the CNN method would require a new training on the dataset using the same characters. There is not enough training data available and thus this method is skipped. Table 5.38 shows the results of both methods on the grayscale images of the Glagolitic database. The Fisher vector method performs better than the BOW approach, but the difference between these two methods is marginal for the soft criterion. For the hard criterion the difference raises to nearly 7%.

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
InVar-250	95.3	97.8	99.2	99.4	88.9	83.9	80.3
RotVar-300	96.7	**98.9**	99.4	**99.7**	91.4	85.9	80.6
InVar-F10- -Pca64-Gauss40	**97.2**	98.6	**99.7**	**99.7**	**94.7**	87.8	82.0
RotVar-F10- -Pca64-Gauss40	96.7	**98.9**	**99.7**	**99.7**	**94.7**	**90.6**	**87.3**

Table 5.38: Evaluation of the soft and hard criteria on the Glagolitc database with grayscale images (in %)

On the binary dataset the Fisher vector approaches have also a higher accuracy than the BOW method. These results can be seen in Table 5.39. For most of the soft criteria both parameter sets of the Fisher vector achieve 100%, except for the Top-1 criterion. For the hard criterion the Fisher vector with the rotation variant features has a by about 3% better performance than the BOW approach.

	soft criterion				hard criterion		
	Top 1	Top 2	Top 5	Top 10	Top 2	Top 3	Top 4
InVar-250	97.8	98.9	99.7	99.7	95.8	93.1	92.2
RotVar-300	96.9	99.2	99.7	**100.0**	95.0	93.1	91.1
InVar-F10--Pca64-Gauss40	98.3	**100.0**	**100.0**	**100.0**	97.2	96.1	94.7
RotVar-F10--Pca64-Gauss40	**99.4**	**100.0**	**100.0**	**100.0**	**98.6**	**96.9**	**95.8**

Table 5.39: Evaluation of the soft and hard criteria on the Glagolitc database with binary images (in %)

5.5 Summary

In this chapter the three proposed methods for writer identification has been evaluated on commonly used datasets, namely the ICDAR 2011, ICDAR 2013, and the CVL-DB. The BOW method and the Fisher vector method has additionally been evaluated on the ICFHR 2012 and the Glagolitic database. In the comparison of the results it can be seen, that the Fisher vector has the best overall performance. The CNN method, even though it is only a proof of concept, has a better performance than the BOW method, which performs worst. It can also be seen that all three methods have problems if the alphabet changes within the dataset. The CNN method is still able to compete with the other methods, although in the trainings set no Greek characters are available. Surprisingly subsequent experiments, which are not presented, showed the when training with additional pages which are written in Greek there is no increase of performance. The experiments on the Glagolitic database show that the Fisher vector and the BOW approach can also be applied on historical data. Because of the non uniform background of this dataset, the contrast between the writing and the background is changes, which also influences the generation of the SIFT features. Thus, a preprocessing step is introduced to improve the performance. For both methods the visual vocabulary is generated on Latin scripts, also when dealing with Glagolitic characters. This means that the vocabulary is universally applicable for every alphabet.

CHAPTER 6

Conclusion and Future Work

This chapter gives an overview about the contribution of this work, followed by a summary of the methods developed within the scope of the work. Possible improvements, which will be done as future work, are proposed and incorporated in the respective summary.

The contribution of this work is the development of new methods for writer identification and writer retrieval. Three approaches are proposed, which originate from the field of object recognition. The methodology of this work is brought to the field of document image analysis and is extensively evaluated using several commonly used databases. While the approach for BOW is unchanged to the original method, the Fisher vector have been adopted by using only the gradients derived to μ and by skipping the normalization term in the equation. The normalization is done afterwards by using a L2-normalization of the vector. Furthermore, rotation variant SIFT descriptors, which are proposed by Diem and Sablatnig [17], are incorporated into the method. Various experiments have shown, that when using the Fisher vector approach the rotational variant descriptors perform better than the original one. This is because the features of the upper and lower profile of the writer are now distinguishable and the upper and lower profiles of the handwriting is a discriminative feature for each writer. Both approaches are bringing the local features to the field of writer identification. It has been shown that state of the art methods are using features which are calculated at character level. Since the image has to be segmented first, errors can be introduced in the pre-processing steps which are hard to correct later on. Also the use of the deep learning concept is brought to the field of document image analysis. It is shown that a simple adaption of the concepts to document images by using sliding windows of extracted and normalized text line images can be used for a successful identification of the writer.

To summarize the problem statement given in Chapter 1 were the goals of this thesis have been defined:

Is a reliable writer identification and retrieval method possible without being dependent on preprocessing steps?

Two of the three methods proposed in this thesis do not use pre-processing methods. Only if the document image has not only handwriting in it, a localization of the areas with handwriting has to be used but this was not covered in this work. Most of the state-of-the-art methods presented in

Chapter 2 do use a binarization step, text line detection, or deletion of the white spaces between the text lines which is not necessary for the two proposed methods. Evaluation showed that the method proposed outperforms or achieve a similar performance as the state-of-the-art on different databases. The evaluation on the historic database showed that an identification and retrieval of writers is also possible if the background is not uniform, even though the performance is increased when using a binarization step. The reason for this is that the proposed method is also learning the background since SIFT features calculate the gradient between handwriting and background. This can also be a desirable property, for example for the Stasi files where multiple document pages, which belong together and have been written at the same day and on the same type of paper, have to be found and the ones with a similar background have a slightly lower distance. The third method does depend on pre-processing steps.

Can the gray value information be exploited for this task?
Since SIFT features are used, the gray value information of the document image is already used for identification and retrieval. During a binarization step the additional information stored in the gray values is lost and cannot be used for identification. The gray value is responsible for learning background and also the color of the pen has an influence to the identification task. In the CVL-DB multiple users have written with different pen types and color (see Figure 1.3) and the proposed method outperforms the state of the art on this dataset. Thus, the gray value information is exploited for the two methods which are using SIFT features, but the influence is not so high that they favor the pen color instead of the writing style.

Does this method also work if only a few lines of handwritten text are present?
The evaluation on the ICDAR 2011 cropped dataset showed, that the Fisher vector and the CNN method have high performance. When using the Top-1 evaluation criterion the CNN method achieves 94.7% identification rate and the Fisher vector 93.8% which is an increase of nearly 4% compared to the state of the art. For the Top-2 hard criterion the best performance is achieved by the Fisher vector method with 61.5%, which is an increase of 21.6% compared to the participants. The BOW approach achieves only 76.4%. For the Top-7 criterion, which is finding all other pages of the same writer, the performance of the Fisher vector method is 21.2%, whereas the performance of the CNN drops to 10.1%. Since the cropped dataset only consists of two text lines one can say that the methods are capable of identifying a writer even if there is only a few handwritten text available. The ICFHR 2012 and ICDAR 2013 datasets contain four text lines on each page and the proposed method returns also good results for the identification tasks.

Can it be adopted easily for historic databases?
The experiments on the historic dataset showed that the BOW method, as well as the Fisher vector approach, can be applied to historic databases. Even if a different alphabet is used for the generation of the visual vocabulary, the performance is still high with 97.2%. This performance can still be improved when the images are binarized, but this may be a property of the Glagolitic dataset, since the background of the documents of one writer differ. If the evaluation dataset consists of multiple alphabets, the performance is dropping. The CNN method cannot easily be

adopted to new datasets, since it has to be trained on characters of an alphabet which are at least similar to the one used in the evaluation dataset.

To summarize this thesis, short descriptions of the three methods for writer identification and writer retrieval are given: The first method is based on SIFT features and a BOW approach. On a trainings set the SIFT features are calculated and all features of the trainings set are then clustered using *k-means*. This clustering divides the feature space and with an occurrence histogram new images of handwriting can be examined and is called BOW. On a new image the first step is to calculate again the SIFT features on the handwriting and then for the generation of the occurrence histogram for each of these features the nearest cluster center is searched using the Euclidean distance. The corresponding bin in the histogram of the nearest cluster center is then increased. So for each image of a handwriting a histogram is generated which describes the distribution of the SIFT features in feature space. These histograms are then used for writer identification or writer retrieval. The distance of each feature vector of an image is compared to precomputed features vectors of a known dataset using the χ^2 distance. The smaller this distance is, the more similar are the handwritings. The writer of the document with the smallest distances is then assigned as author of the new document. For writer retrieval the goal is to get the most similar document images according to the handwriting. Thus, the documents in the database are sorted according to the distance.

The second method which was presented was also based on SIFT features. Also a modification of the SIFT features, which makes the features variant to rotations up to 180 degrees, have been evaluated and showed an increase of performance. Instead of the BOW approach the Fisher vector is used for identification and retrieval. While the BOW approach divides the feature space in strict borders, for the Fisher vector a GMM is used for clustering. This leads to a more precise representation of the feature space. So for each coordinate in the feature space the probability that a particular feature was generated by a Gaussian can be calculated. The Fisher information can be calculated for each feature using the gradient vector which is the first derivation. Again, all SIFT features of a trainings set are clustered with a fixed number of Gaussians. As feature vector the gradient vectors for each of the Gaussians are concatenated. As performance improvement a L2 normalization and a power normalization are performed. When identifying a new writer, first the SIFT features are calculated on the handwriting and then the Fisher vector is generated. These feature vectors are then used for the writer identification and writer retrieval. Again, like when using the BOW method, the distance of the feature vector of an new images to the feature vectors stored in the database is calculated but this time using the cosine distance, as it is a natural distance measurement for the Fisher vector. The author of the document with the smallest distance is then assigned to the new document and for writer retrieval the documents in the database are sorted according to this distance. To reduce the computational complexity a PCA can be applied to the SIFT feature which reduces the dimension of the feature vector.

The last method presented is based on CNN. This time pre-processing steps were necessary. First, the image was binarized and then the text lines were searched. The binarization has been done using the Otsu method, since on the evaluation databases a global threshold lead to a nearly perfect segmentation. For the text line detection a method based on LPPs was used. The text lines were also deskewed using the lower and the upper contour points of the line. Then, with a sliding window approach, the text lines processed resulting in small portions of the handwriting containing only 1-2 characters. All images of the trainings set were then fed to a CNN. As architecture of the CNN the well known "Caffenet" was used. Since the last fully connected layer of a CNN does the classification, this layer is cut off and the activations of the neurons of the second last layer are used as feature vector for this particular sliding window. As feature vector for the complete page the mean vector of all image portions of the document image was used. When identifying a new writer, the pre-processing steps are executed and all image portions were again given to the CNN. Again, the mean vector of the second last layer was taken as feature vector. The distance of this vector to the feature vectors stored in a dataset was then calculated, using the χ^2-distance, and the author of the document with the smallest distance was assigned as writer to the new document. For writer retrieval, the documents in the dataset are sorted according to the distance. This method was only developed as a proof of concept and to the best knowledge of the author is the first who brings CNNs to the field of writer identification.

All three methods have been evaluated on four scientific datasets and the BOW words and Fisher vector methods have also been evaluated on a historic dataset. As scientific datasets the dataset of the ICDAR 2011, ICFHR 2012, and ICDAR 2013 writer identification competitions as well as the CVL-DB were used. The datasets comprise 26 to 310 different writers. As evaluation criteria the hard and soft criteria, which were introduced in the ICDAR 2011 writer identification competition, were used. Each document is used as reference document and the documents written by the same writer have to be found in the remaining dataset. The remaining documents in the dataset are sorted according to the distance to the reference document and for the soft criteria at least one document written by the same writer has to be in the first N documents. For the hard criterion all N documents have to be written by the same writer as the reference document. The ICDAR 2011 dataset contains of a complete and a cropped dataset, which only contains one or two text lines of the original dataset. For the complete dataset the identification (Top-1 criterion) was solved nearly perfect on the complete dataset, except the CNN method was not able to identify the writer of one document. On the cropped dataset the highest accuracy was achieved by the CNN method, which identified 94.7% of writers correctly. On the ICFHR dataset the best performance was achieved by the Fisher vector method with 96.3%, since this dataset contains texts written in Greek and English and the hard Top-2 criterion means that one document written in the other language has to be found, the performance is only 47% which is nearly 15% less than the best participant. This means that the methods have problems in identifying the writer of a document, when the document is written in a different alphabet. On the ICDAR 2013 dataset, the highest accuracy was 96.8% and on the CVL-DB the identification rate was 99.1%. On the historic dataset, which is written in Glagolitic script, the identification rate was 97.2% on the grayscale images and 99.4% on the binarized images. This means that both methods have a good performance without the pre-processing of the images, but in this case the pre-processing increases the performance. This is due the slightly adoption of the method to the background. Since the manuscripts in this dataset have been stored in two different locations the contrast of the handwriting to the background differs, which results in different gradients of the SIFT features.

To conclude, the three methods developed in the scope of this thesis have shown to have a high accuracy on the evaluation databases. Furthermore, with the modification on the SIFT features a higher performance was achieved when using the Fisher vector. This modification does not have a great impact when using the BOW approach. The advantage of the BOW and Fisher vector approaches are that they are computational relatively fast, since only the SIFT features have to be calculated and for the BOW the occurrences of the nearest cluster centers have to be counted. For the Fisher vector, the gradients of the Gaussians have to be calculated, which needs more computational power but is still achievable within 1-3 seconds. Calculating the distance between two feature vectors depends on their dimensions. Thus, the BOW, where the dimension of the feature vector equals the number of clusters chosen, is very fast since the feature vector has low dimensions. Experiments showed that the best performance was achieved when using 250 and 300 cluster centers. For the Fisher vector, the dimension is ND, where N is the number of Gaussians and D is the dimension of the local features. It has been shown that the best values for N are 10 respectively 40, and with the application of a PCA the dimension of the SIFT features can be reduced to 64. Future work for both methods include the development of new features, which describe the handwriting in more detail. Also the spatial relationship between the SIFT features can be exploited and may lead to a better performance. Future work may also include the selection of SIFT features which carry the relevant information for writer identification and retrieval. Deleting features which do not add information to this task increases the performance, since they have an equal influence to the resulting feature vector of the writer as other SIFT features.

The CNN method is, to the best knowledge of the author, the first method which brings the field of deep learning to the field of writer identification. The disadvantages of this approach is that it is very time consuming. The training phase of the network takes 30 or more hours and also identifying the writer of a new image is slow. The pre-processing steps can be processed quite fastly, but if the document images become more challenging the computational power of these steps may also increase. But each portion of the image has to be fed to the CNN resulting in multiple seconds identification time for a complete page. Future work for the CNN method includes the development of a new network architecture, which is able to represent the differences between different writers better. Also the pre-processing steps will be investigated in more detail. Instead of just simply taking a sliding window, experiments will be made which rely on single characters. Therefore, different character segmentation methods have to be evaluated. Another improvement will be the detection of image portions which do not have enough information for a writer identification. Sliding windows with only some meaningless strokes have to be filtered out, since they contribute equally to the feature vector of the whole page as image portions which carry the information about the writer. Taking the mean of all vectors of the second last layer at the end as feature vector could also be improved by applying a voting mechanism or by determining the importance of the particular portion of image to the identification and retrieval process.

An overall improvement of the writer identification process is to follow the idea of Djeddi et al. in [19]. They propose to use a retrieval mechanism before the identification takes place. The retrieval mechanism searches for the most similar pages in the database of known writers. The documents returned by the retrieval mechanism are then analyzed further to identify the writer of the particular handwriting. Since the search space is reduced drastically by the retrieval mechanism, a writer identification method can be used which does a comparison on higher levels, e.g. one to one comparisons of characters or even whole words which are significant for the writer in the reference document, which will give a boost to the identification rate.

CHAPTER 7

Acronyms and Symbols

BOW	Bag of Words
CNN	Convolutional Neural Network
CC	Connected Components
CVL-DB	CVL-Database
DOG	Difference Of Gaussians
GLCM	Gray Level Co-occurrences Matrices
GMM	Gaussian Mixture Model
GMSF	Grid Microstructure Features
IAM-DB	IAM Handwriting Database
ICDAR	International Conference on Document Analysis and Recognition
ICFHR	International Conference on Frontiers in Handwriting Recogntion
KAS	K-Adjacent Segments
MNIST	Modified NIST (National Institute of Standards and Technology)
ILSVRC	Imagenet Large Scale Visual Recognition Challenge
LBP	Local Binary Patterns
LPP	Local Projection Profiles
NN	Neural Network

PCA Principal Component Analysis

PDF Probability Density Function

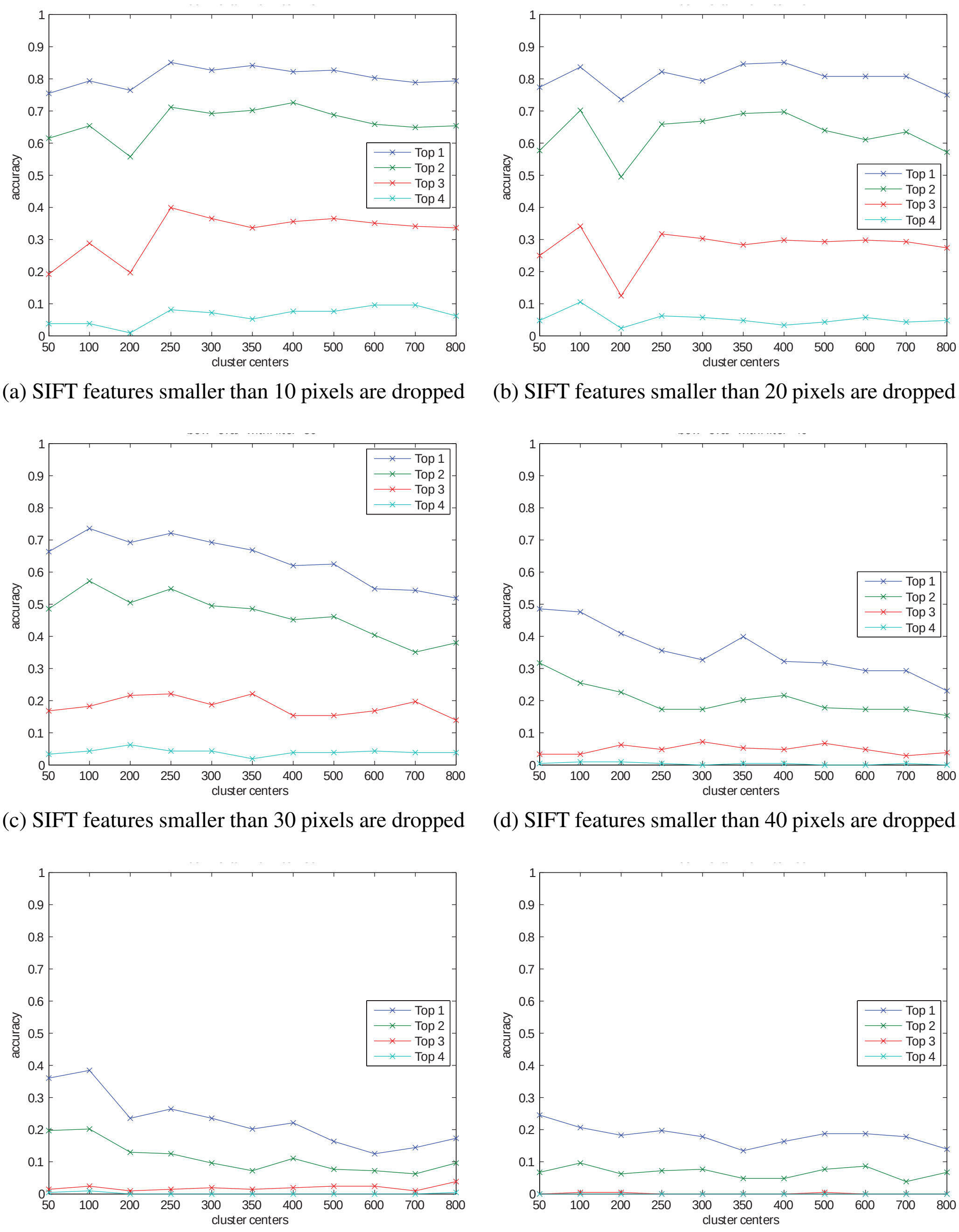
SIFT Scale Invariant Feature Transform

UBM Universal Background Model

APPENDIX A

Parameter Evaluation for the BOW Approach

In this appendix the parameters for the BOW method, which is presented in Chapter 4, are shown. The parameter evaluation is carried out to find the optimal parameter set for this method. This is done by evaluating the method with different parameter sets on a database and the highest performing set is taken for the evaluation. Since it might be possible that this set is only optimal for the evaluation dataset used, not only the best set is taken for further evaluations, but also another parameter set which has only a slightly worse performance and is also distinctive enough from the best set. The free parameters of the BOW method are the number of cluster centers and also the minimal size of the SIFT features used for generating the occurrence histogram. Also the application of the PCA has been evaluated on this method and two different SIFT features are used: First the original features proposed by Lowe [56] in Figures A.1 to A.4 and second modified features, which are proposed by Diem and Sablatnig [17], in Figures A.5 to A.8. The minimum size of the features was set to 0, 10, 20, 30, 40, 50, 60, and 70 and for the number of cluster centers the values 50, 100, 200, 250, 300, 350, 400, 500, 600, 700, and 800 were chosen. 32, 64, and 96 components of the PCA were used and also without the usage of the PCA.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

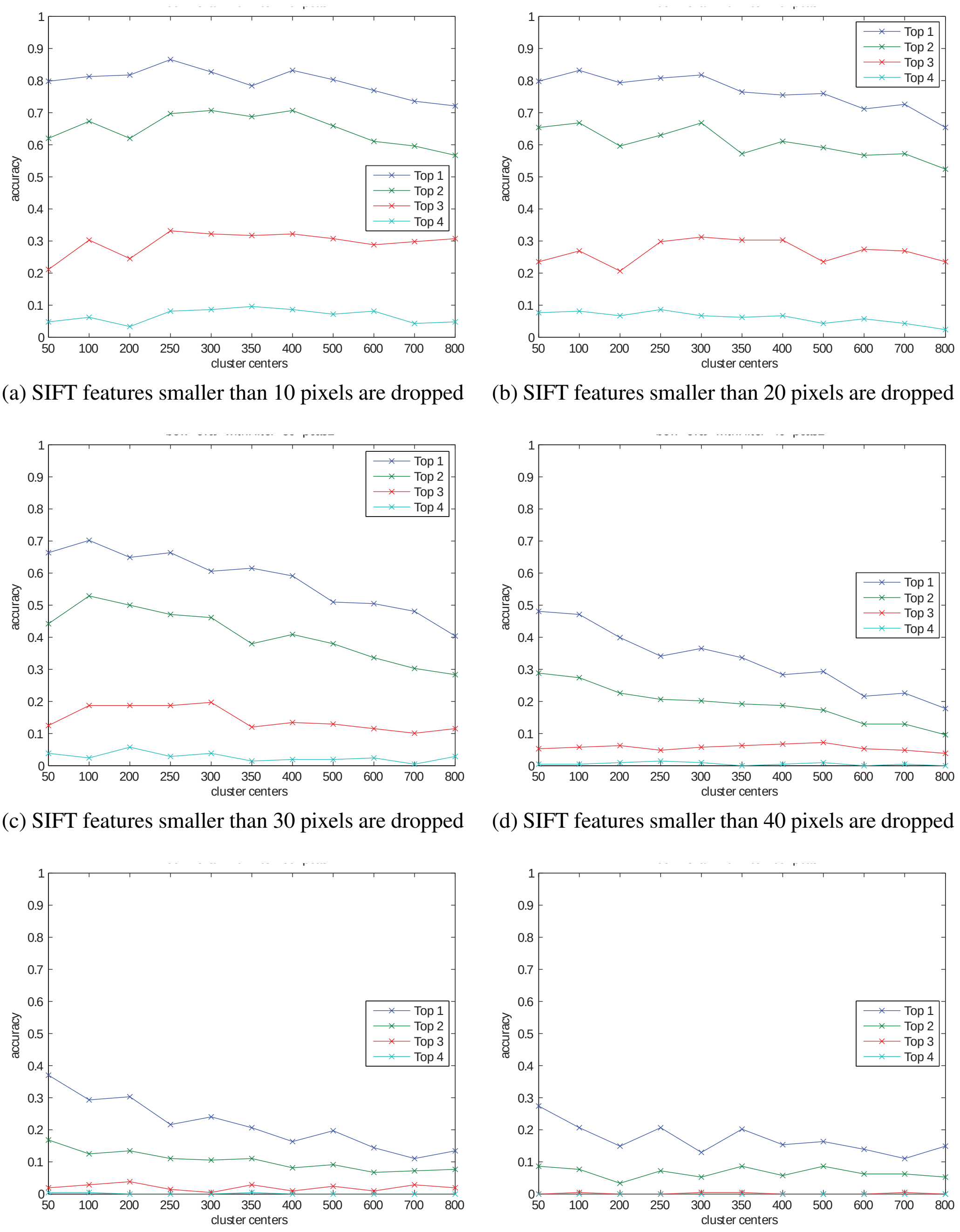
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure A.1: Parameter evaluation for the BOW approach using the unmodified SIFT features without applying a PCA. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with $Top\ 1$, $Top\ 2$, $Top\ 3$ and $Top\ 4$.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

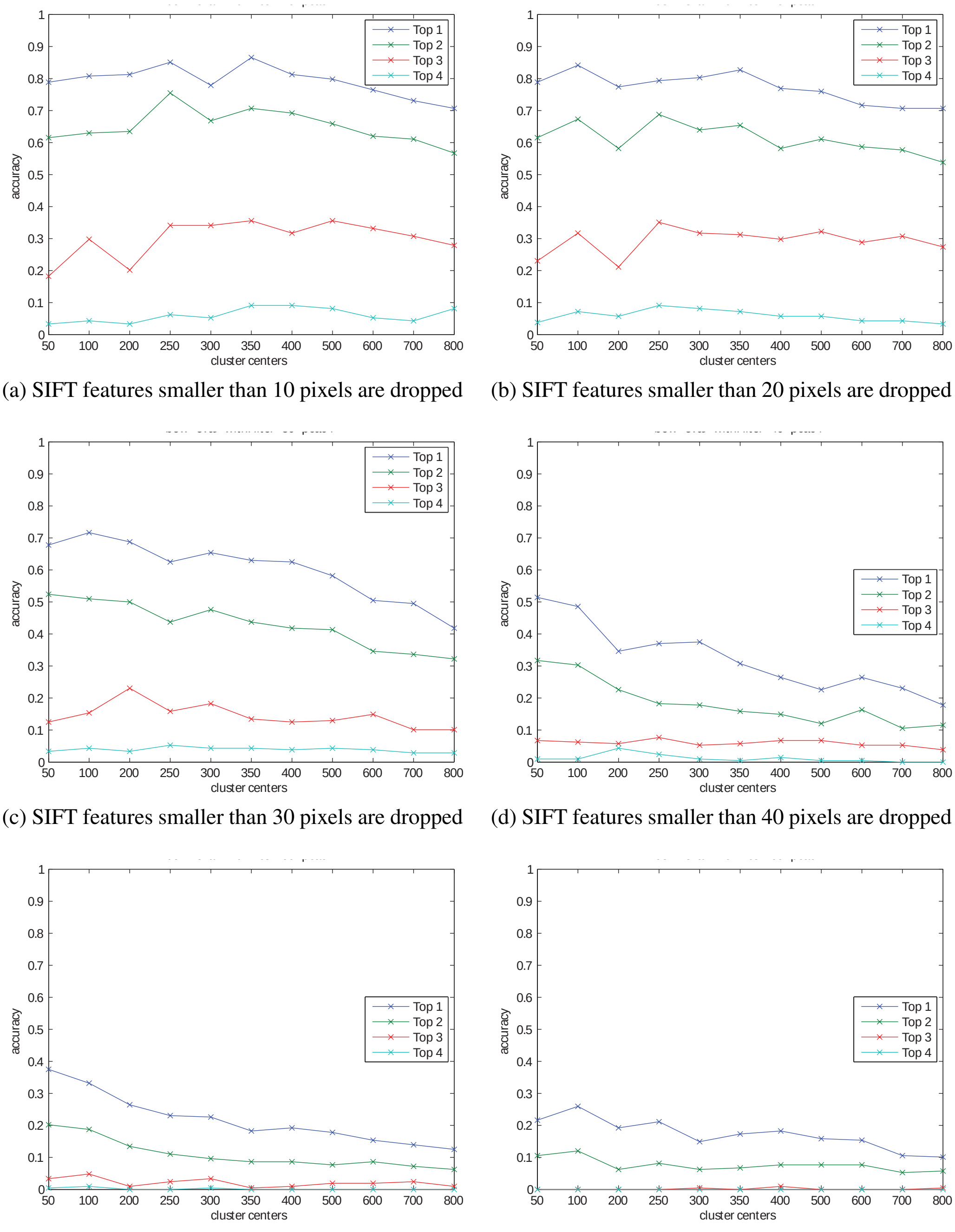
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure A.2: Parameter evaluation for the BOW approach using the unmodified SIFT features and taking the first 32 components of the PCA. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with Top 1, Top 2, Top 3 and Top 4.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

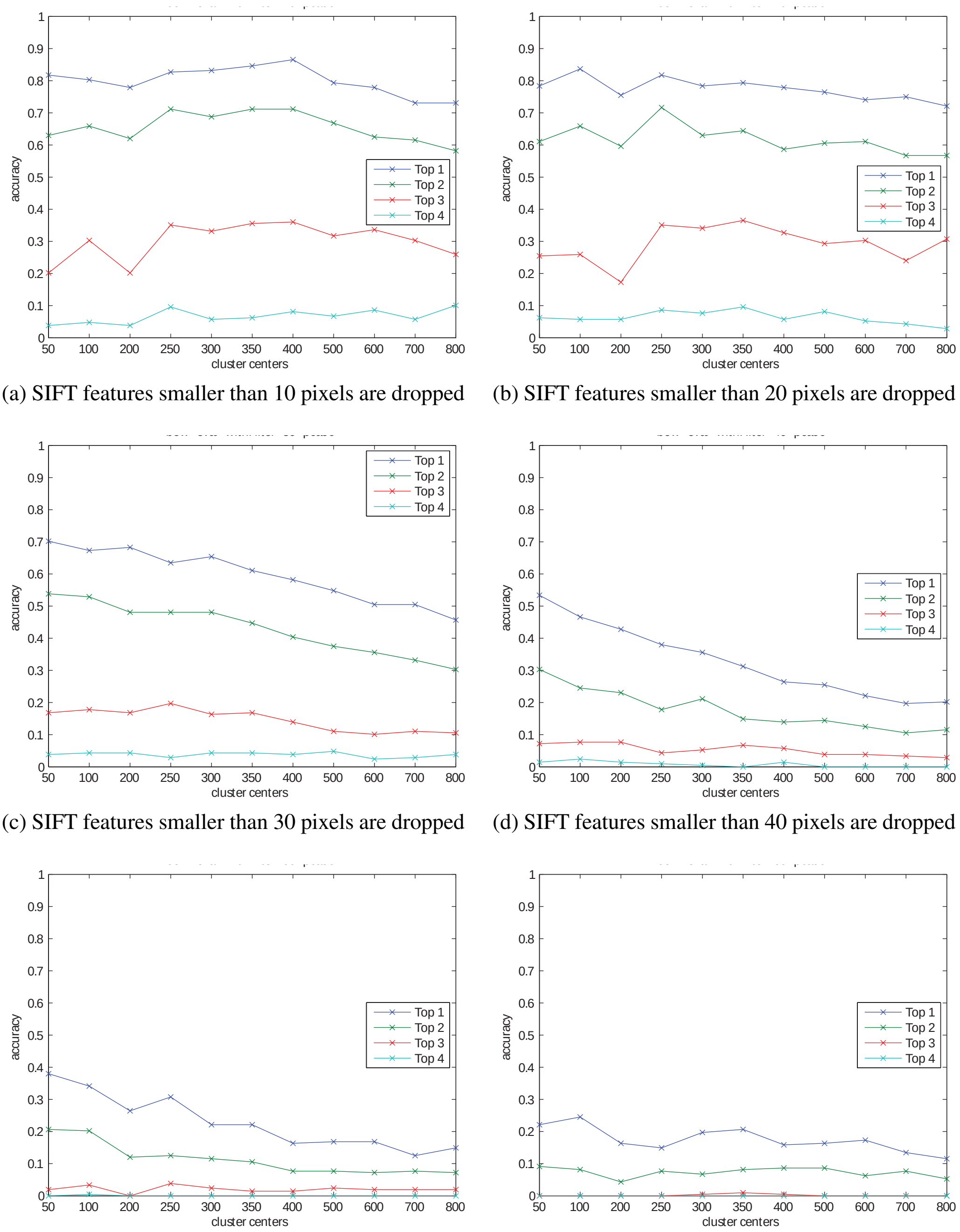
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure A.3: Parameter evaluation for the BOW approach using the unmodified SIFT features and taking the first 64 components of the PCA. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with Top 1, Top 2, Top 3 and Top 4.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

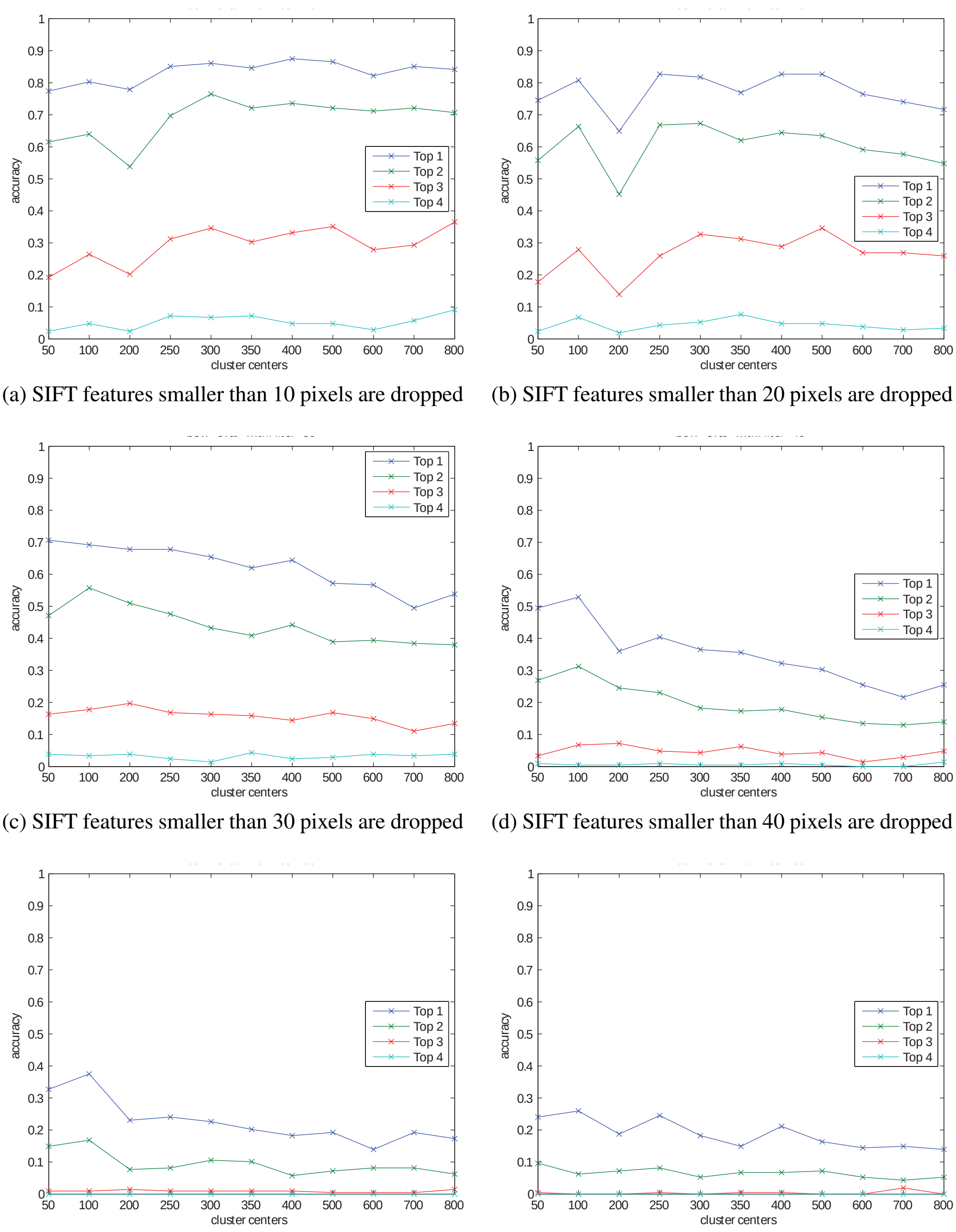
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure A.4: Parameter evaluation for the BOW approach using the unmodified SIFT features and taking the first 96 components of the PCA. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with $Top\ 1$, $Top\ 2$, $Top\ 3$ and $Top\ 4$.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

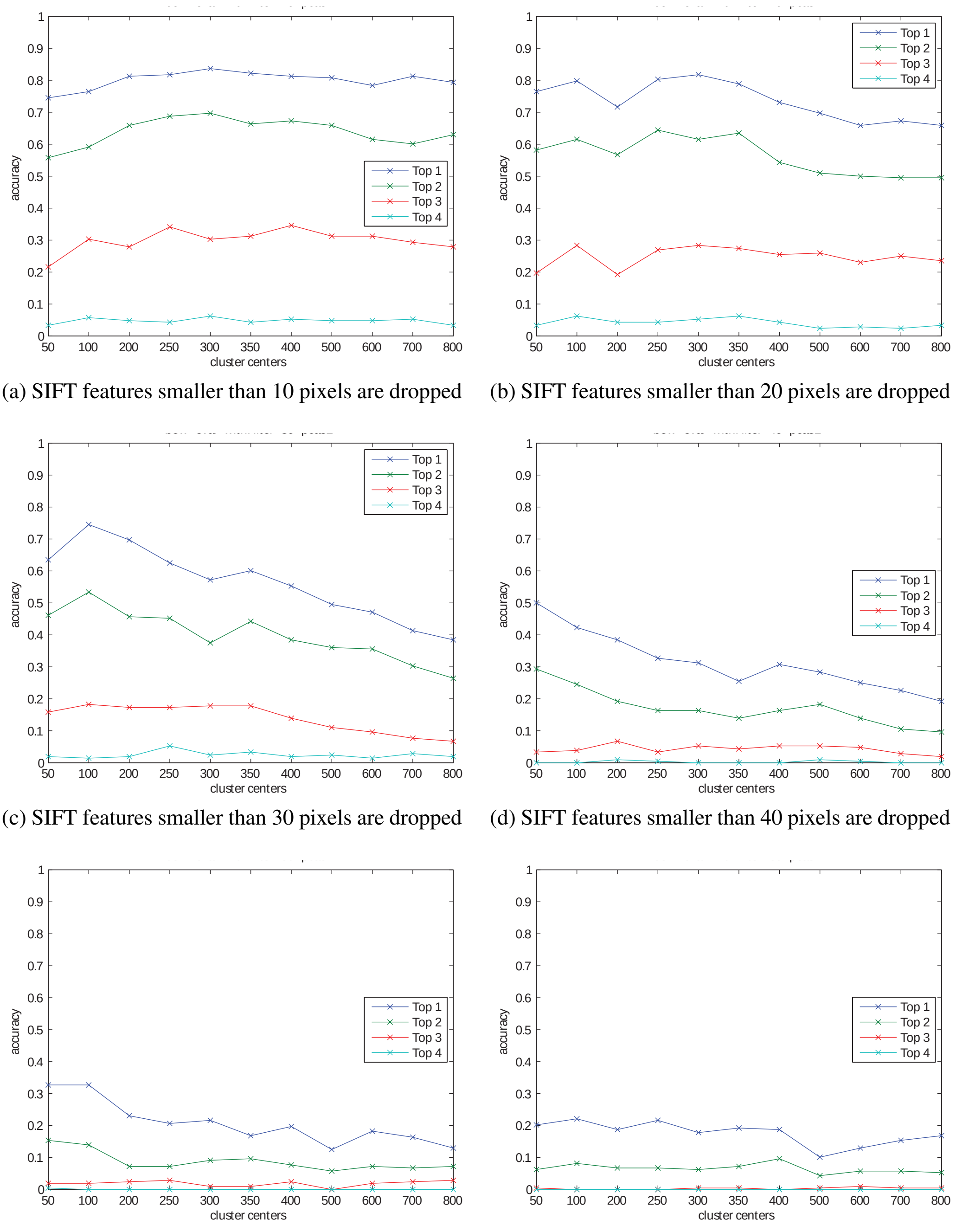
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure A.5: Parameter evaluation for the BOW approach using the rotation variant SIFT features without applying a PCA. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with $Top\ 1$, $Top\ 2$, $Top\ 3$ and $Top\ 4$.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

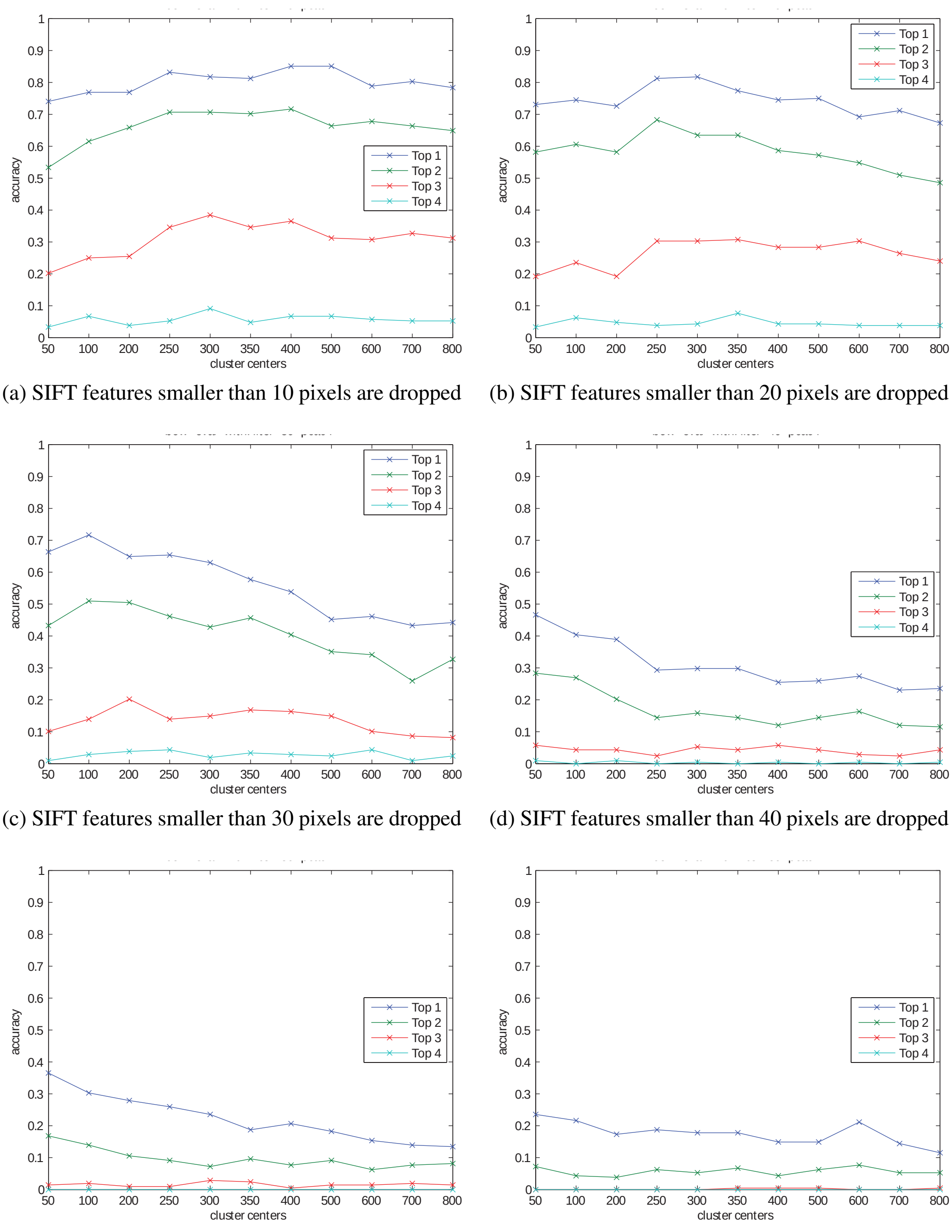
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure A.6: Parameter evaluation for the BOW approach using the rotation variant SIFT features and taking the first 32 components of the PCA. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with Top 1, Top 2, Top 3 and Top 4.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

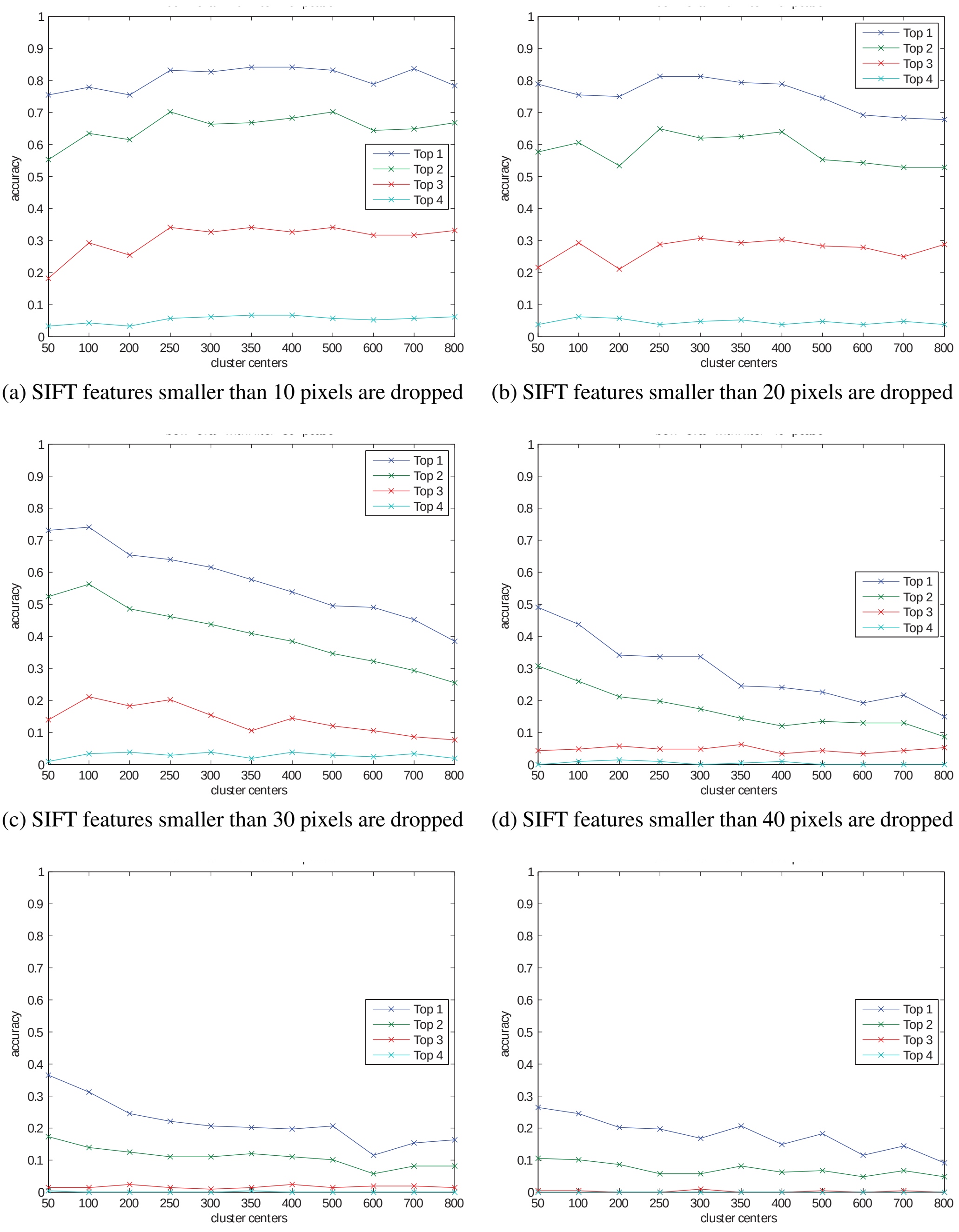
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure A.7: Parameter evaluation for the BOW approach using the rotation variant SIFT features and taking the first 64 components of the PCA. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with $Top\ 1$, $Top\ 2$, $Top\ 3$ and $Top\ 4$.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

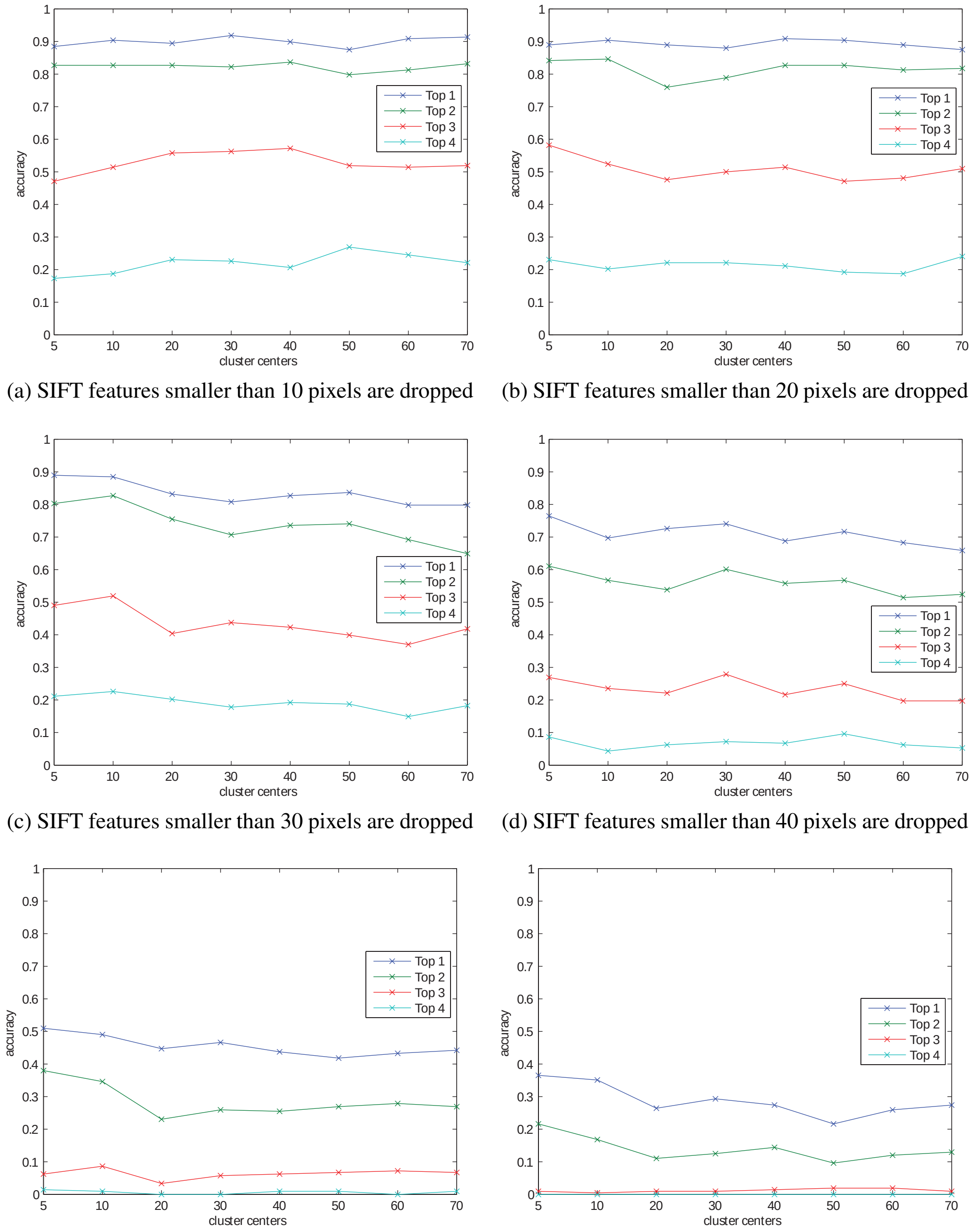
Figure A.8: Parameter evaluation for the BOW approach using the rotation variant SIFT features and taking the first 96 components of the PCA. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with $Top\ 1$, $Top\ 2$, $Top\ 3$ and $Top\ 4$.

It can be seen that the best results were achieved if no PCA is used, independent of using the rotation variant or the rotation invariant SIFT features. These results can be seen in Figure A.1 and Figure A.5. Further it can be seen in all evaluations that with increasing minimum size of the SIFT features the overall performance decreases. This means that for this approach small features carry valuable information for the identification of writers. Especially when filtering smaller SIFT features, increasing the number of cluster centers leads to a worse performance. When not filtering at all, the performance is nearly equal when changing the number of centers. The best parameter sets on the evaluation dataset is achieved when using a filter size of 10 and 250 cluster centers. The set has been chosen by determining the mean performance of the *Top 1* to *Top 4* criterion. When using the modified SIFT features, the best performance is achieved when using 300 cluster centers. These best two configurations are also evaluated when using the other SIFT descriptor resulting in four different parameter sets.

APPENDIX B

Parameter Evaluation for the Fisher Vector Approach

In this chapter the parameter evaluation for the Fisher vector is presented. The goal is, like for the parameter evaluation for the BOW approach, to find the set of parameters with the highest performance. The free parameters for the Fisher vector method are the same like for the BOW method, except that the number of cluster centers is replaced with the number of Gaussians. The values for filtering the small SIFT features as well as the settings for the PCA are the same like for the parameter evaluation of the BOW method. 5, 10, 20, 30, 40, 50, 60, and 70 are used as numbers of the Gaussians distributions.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

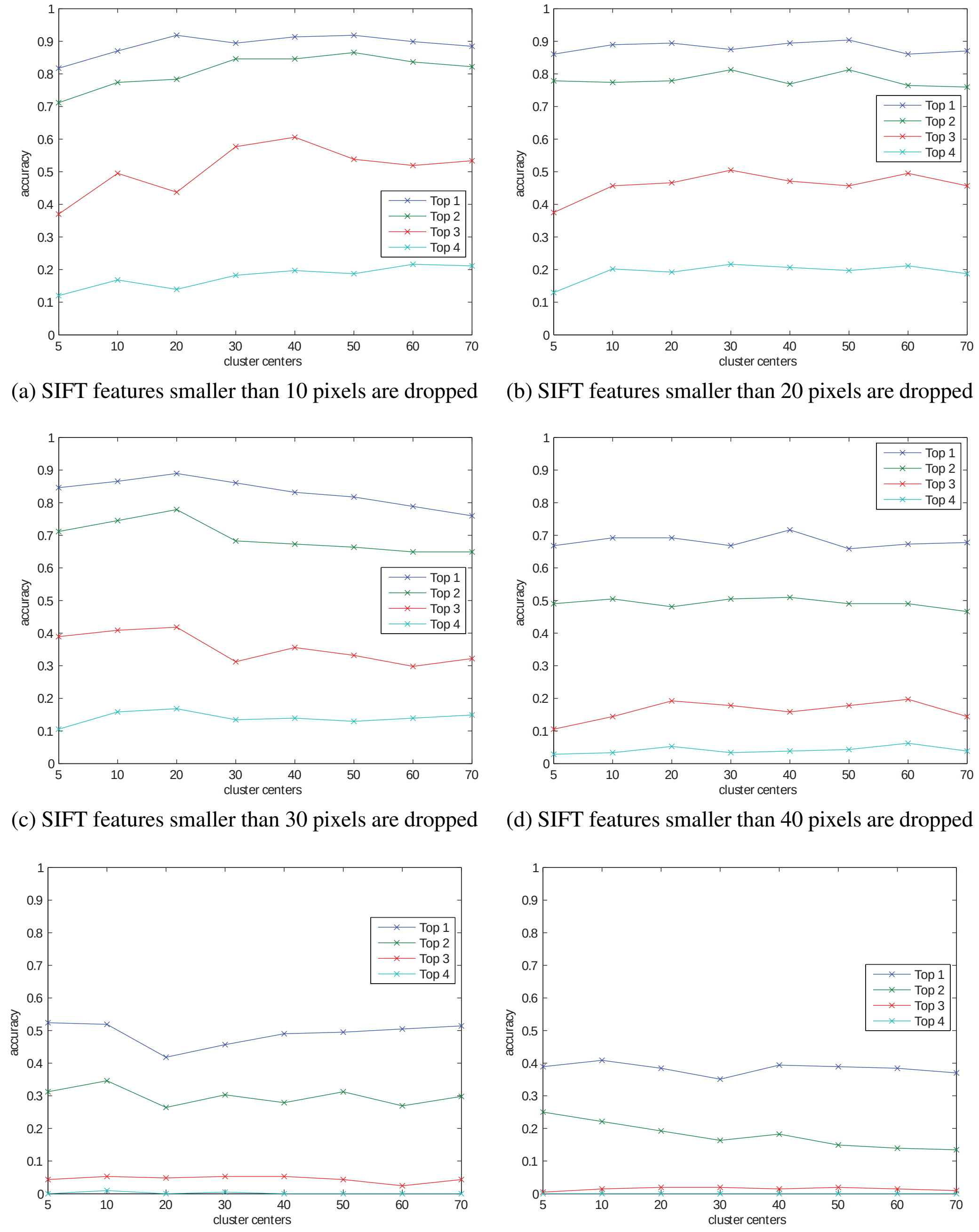
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure B.1: Parameter evaluation for Fisher Vector approach. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with $Top\ 1$, $Top\ 2$, $Top\ 3$ and $Top\ 4$. Rotation invariant SIFT features were used and no PCA was applied to the features.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

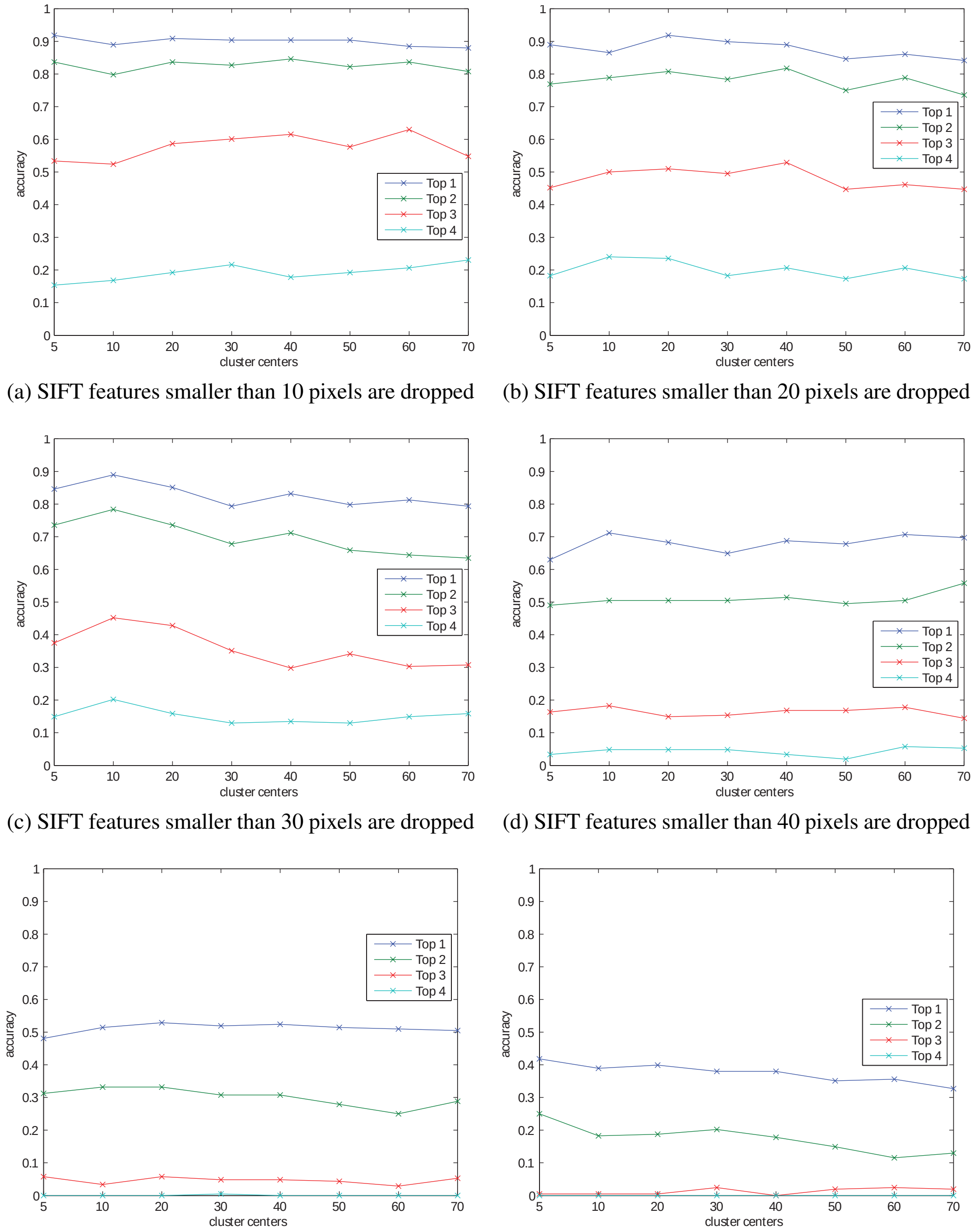
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure B.2: Parameter evaluation for Fisher Vector approach. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with Top 1, Top 2, Top 3 and Top 4. The first 32 components of the PCA of the rotation invariant SIFT features were used.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

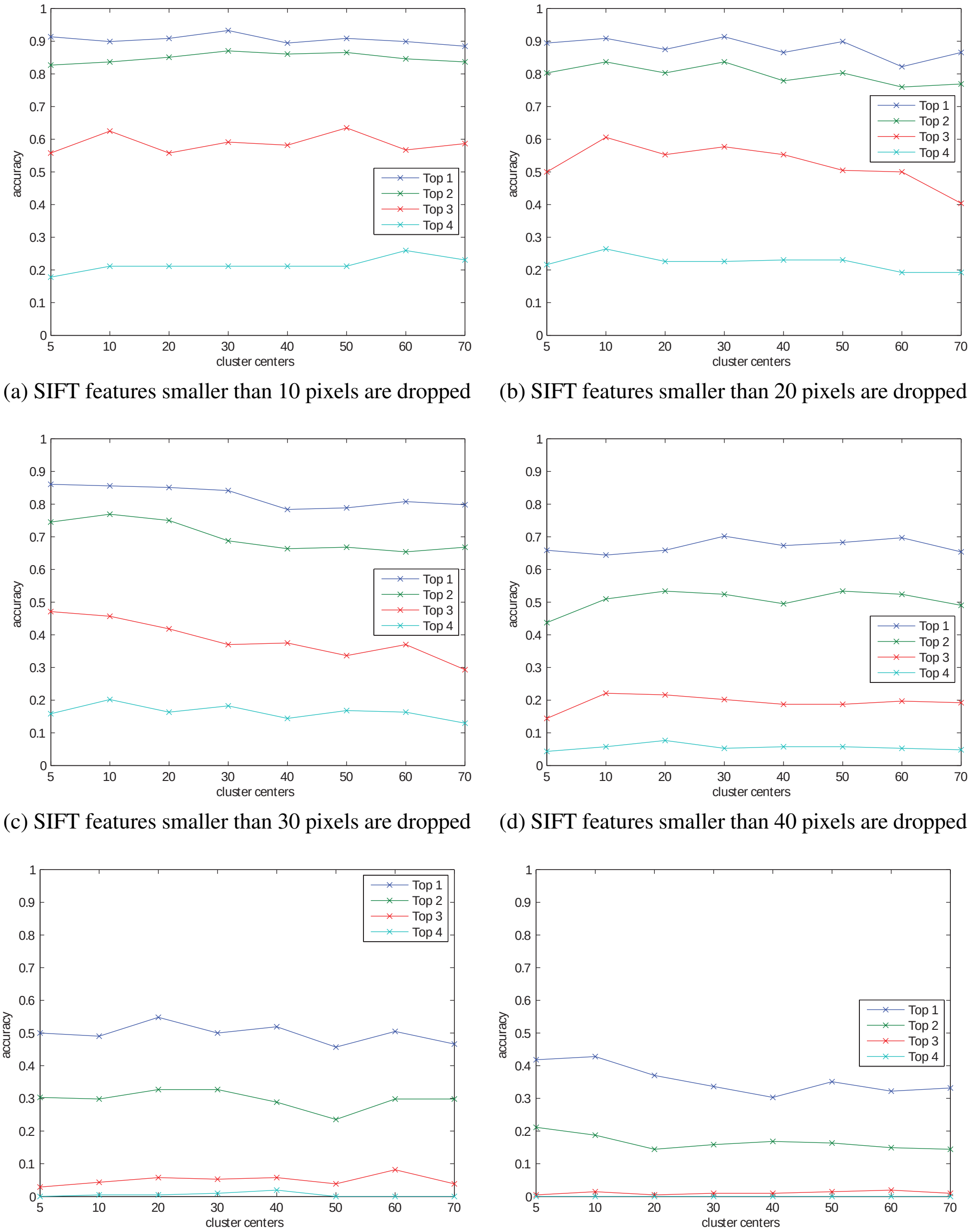
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure B.3: Parameter evaluation for Fisher Vector approach. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with Top 1, Top 2, Top 3 and Top 4. The first 64 components of the PCA of the rotation invariant SIFT features were used.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

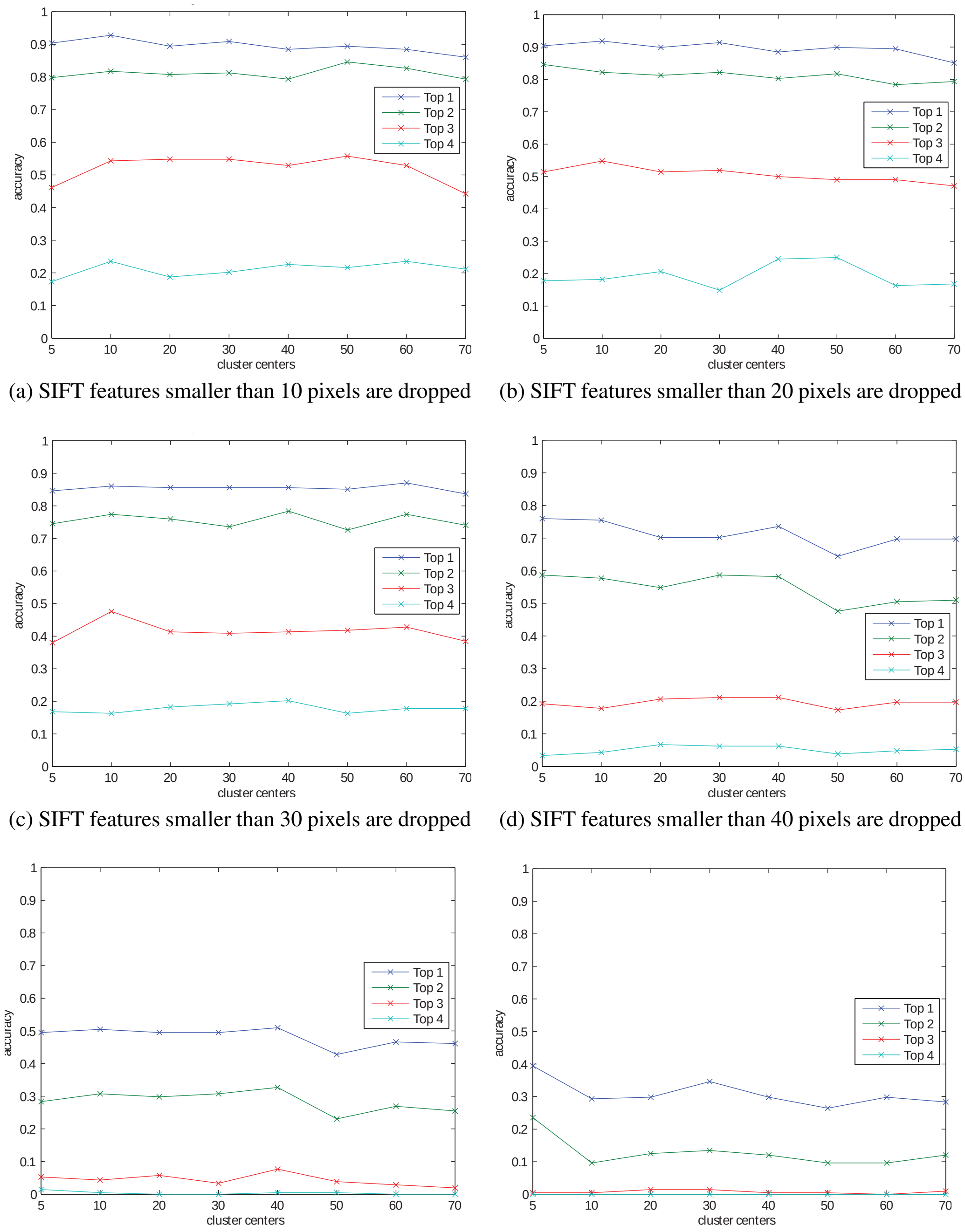
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure B.4: Parameter evaluation for Fisher Vector approach. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with Top 1, Top 2, Top 3 and Top 4. The first 96 components of the PCA of the rotation invariant SIFT features were used.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

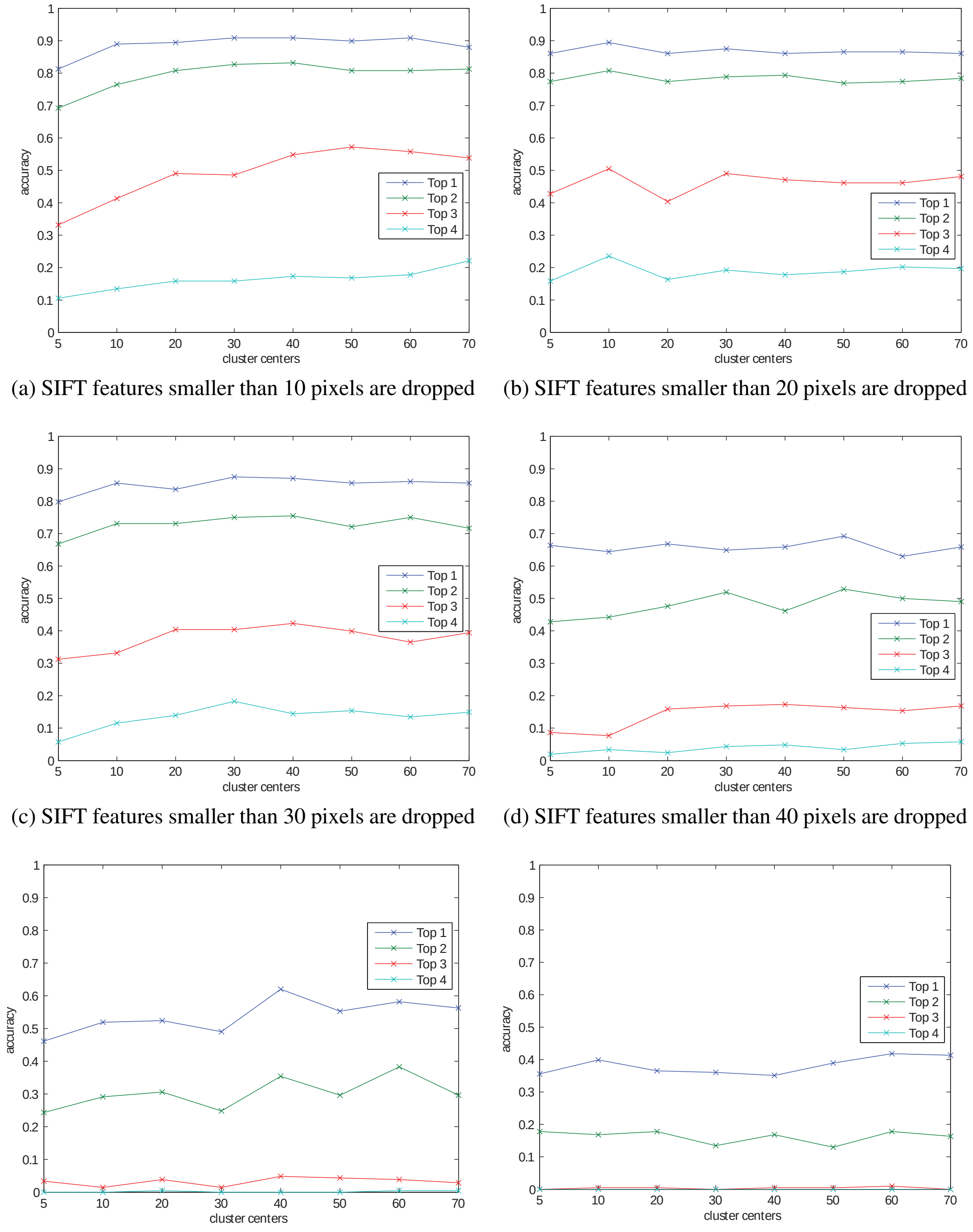
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure B.5: Parameter evaluation for Fisher Vector approach. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with Top 1, Top 2, Top 3 and Top 4. Rotation variant SIFT features were used and no PCA was applied to the features.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

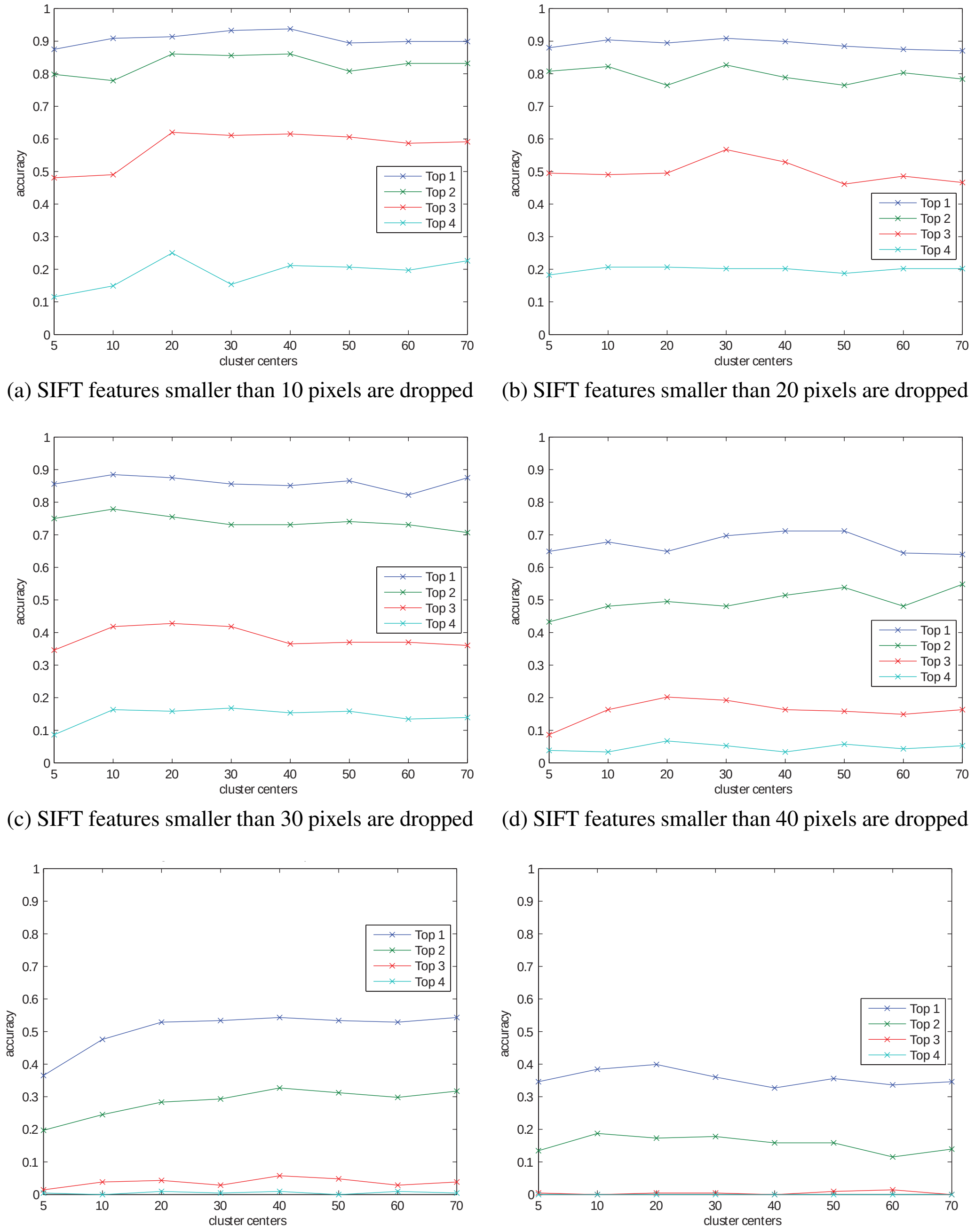
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure B.6: Parameter evaluation for Fisher Vector approach. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with $Top\ 1$, $Top\ 2$, $Top\ 3$ and $Top\ 4$. The first 32 components of the PCA of the rotation variant SIFT features were used.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

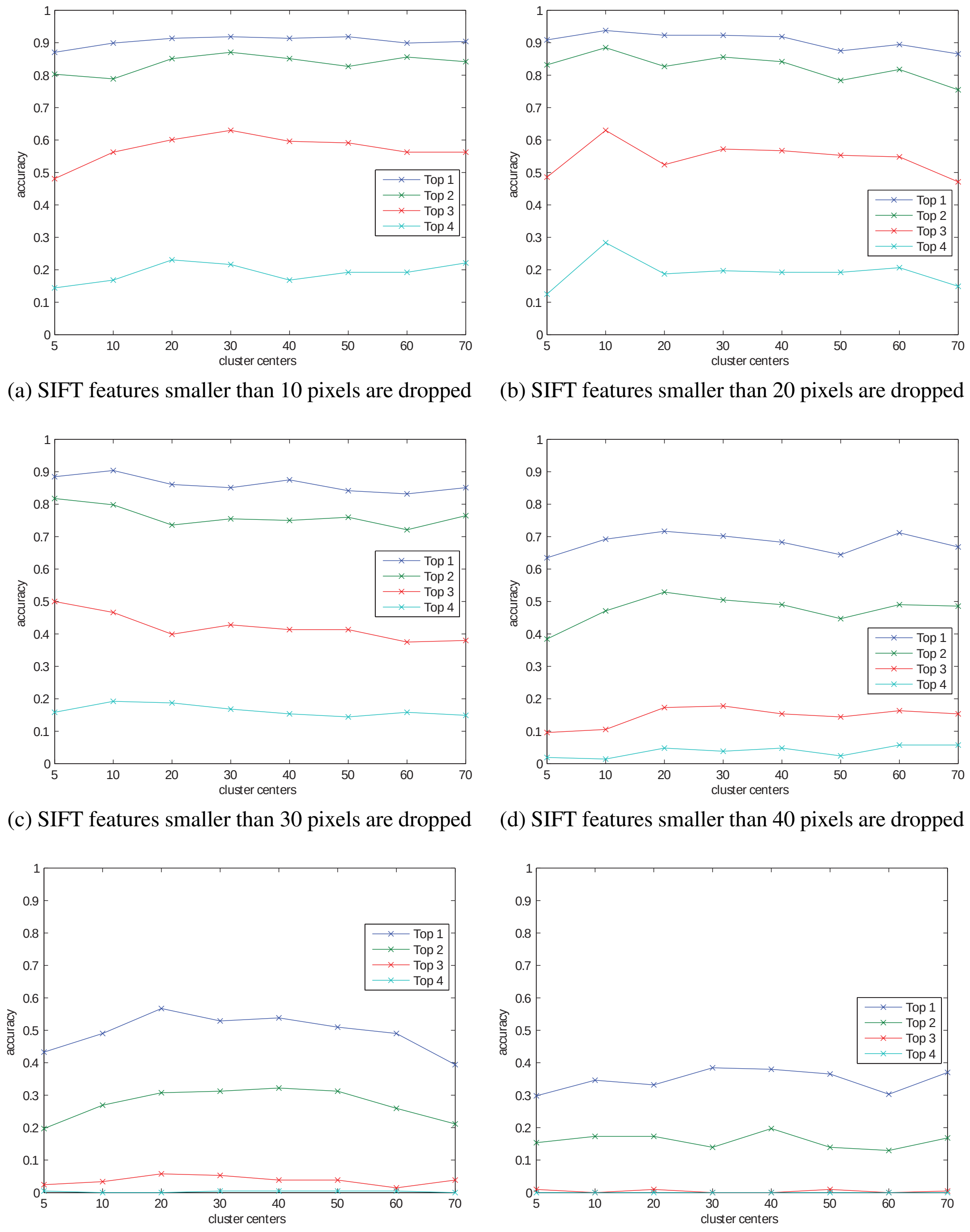
(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure B.7: Parameter evaluation for Fisher Vector approach. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with Top 1, Top 2, Top 3 and Top 4. The first 64 components of the PCA of the rotation variant SIFT features were used.



(a) SIFT features smaller than 10 pixels are dropped

(b) SIFT features smaller than 20 pixels are dropped

(c) SIFT features smaller than 30 pixels are dropped

(d) SIFT features smaller than 40 pixels are dropped

(e) SIFT features smaller than 50 pixels are dropped

(f) SIFT features smaller than 60 pixels are dropped

Figure B.8: Parameter evaluation for Fisher Vector approach. To generate the vocabulary the TripGraph dataset has been used and for tests the cropped ICDAR 2011 dataset was used. The accuracy values are the hard evaluation with Top 1, Top 2, Top 3 and Top 4. The first 96 components of the PCA of the rotation variant SIFT features were used.

In can be seen in the results, when filtering out features with a size above 30 pixels, the performance of this methods drops independently from the PCA setting and from the number of Gaussians used. Too much information which is stored in the smaller SIFT features is lost for a successful writer identification. It can also be seen that when taking too few Gaussians (5) the performance is lower compared to the performance with more Gaussians. The performance starts to decrease when using 70 Gaussians, but only slightly. When using 70 Gaussians without applying a PCA, the dimension of feature vector used is already 8960, compared to 640 when using 5 Gaussians. Thus, comparing the feature vectors for an identification is computationally more expensive. The highest mean average is achieved when using the rotation variant descriptor, take the first 96 PCA components, 10 Gaussians, and set the minimal size of the SIFT features to 20 pixels. For the rotation invariant descriptors, the same feature set has the highest performance. Additionally, the parameter set with a filter size of 10, the first 64 PCA components, 40 Gaussians are also taken for further evaluation since they have the third highest mean performance and the values differ from the highest.

Bibliography

[1] V. Atanasiu, L. Likforman-Sulem, and N. Vincent. Writer Retrieval - Exploration of a Novel Biometric Scenario Using Perceptual Features Derived from Script Orientation. In *2011 11th International Conference on Document Analysis and Recognition (ICDAR)*, pages 628 – 632, 2011.

[2] I. Bar-Yosef, N. Hagbi, K. Kedem, and I. Dinstein. Line Segmentation for Degraded Handwritten Historical Documents. In *2009 10th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1161–1165, 2009.

[3] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded Up Robust Features. In *2006 9th European Conference on Computer Vision (ECCV)*, volume 3951 of *Lecture Notes in Computer Science*, pages 404–417. Springer Berlin Heidelberg, 2006.

[4] A. Bensefia, T. Paquet, and L. Heutte. A writer identification and verification system. *Pattern Recognition Letters*, 26(13):2080–2092, 2005.

[5] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[6] A.A. Brink, R.M.J. Niels, R.A. van Batenburg, C.E. van den Heuvel, and L.R.B. Schomaker. Towards robust writer verification by correcting unnatural slant. *Pattern Recognition Letters*, 32(3):449 – 457, 2011.

[7] M. Brown and D. Lowe. Invariant Features from Interest Point Groups. In *Proceedings of the British Machine Vision Conference*, pages 23.1–23.10. BMVA Press, 2002.

[8] M. Bulacu and L. Schomaker. Text-Independent Writer Identification and Verification Using Textural and Allographic Features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):701–717, apr 2007.

[9] M. Bulacu, L. Schomaker, and L. Vuurpijl. Writer Identification Using Edge-Based Directional Features. In *2003 7th International Conference on Document Analysis and Recognition (ICDAR)*, pages 937–941, August 2003.

[10] W.M. Campbell, D.E. Sturim, and D.A. Reynolds. Support vector machines using GMM supervectors for speaker verification. *IEEE Signal Processing Letters*, 13(5):308–311, May 2006.

[11] V. Christlein, D. Bernecker, F. Honig, and E. Angelopoulou. Writer identification and verification using GMM supervectors. In *2014 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 998–1005, March 2014.

[12] D. C. Ciresan, U. Meier, and J. Schmidhuber. Multi-column Deep Neural Networks for Image Classification. *CoRR*, abs/1202.2745, 2012.

[13] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with Bags of Keypoints. In *In Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22, 2004.

[14] J. Deng, O. Russakovsky, J. Krause, M. S. Bernstein, A. Berg, and L. Fei-Fei. Scalable Multi-label Annotation. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, pages 3099–3102, New York, NY, USA, 2014. ACM.

[15] B.V. Dhandra and M.B. Vijayalaxmi. Text and script independent writer identification. In *2014 International Conference on Contemporary Computing and Informatics (IC3I)*, pages 586–590, Nov 2014.

[16] M. Diem, F. Kleber, and R. Sablatnig. Text Line Detection for Heterogeneous Documents. In *2013 12th International Conference on Document Analysis and Recognition (ICDAR)*, pages 743–747, 2013.

[17] M. Diem and R. Sablatnig. Recognizing Characters of Ancient Manuscripts. In *Proceedings of IS&T SPIE Conference on Computer Image Analysis in the Study of Art*, volume 7531, 2010.

[18] Markus Diem, Florian Kleber, Stefan Fiel, and Robert Sablatnig. Semi-Automated Document Image clustering and Retrieval. In Bertrand Coüasnon and Eric K. Ringger, editors, *Proceedings of Document Recognition and Retrieval XXI*, pages 90210M–1 – 90210M–10. SPIE, 2014.

[19] C. Djeddi, I. Siddiqi, L. Souici-Meslati, and A. Ennaji. Multi-script Writer Identification Optimized with Retrieval Mechanism. In *2012 International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 509–514, Sept 2012.

[20] C. Djeddi, L. Souici-Meslati, and A. Ennaji. Writer Recognition on Arabic Handwritten Documents. In *Image and Signal Processing*, volume 7340 of *Lecture Notes in Computer Science*, pages 493–501. Springer Berlin Heidelberg, 2012.

[21] L. Du, X. You, H. Xu, Z. Gao, and Y. Tang. Wavelet Domain Local Binary Pattern Features For Writer Identification. In *20th International Conference on Pattern Recognition (ICPR)*, pages 3691 –3694, Aug 2010.

[22] J. Farquhar, S. Szedmak, H. Meng, and J Shawe-Taylor. Improving "bag-of-keypoints" image categorisation: Generative Models and PDF-Kernels. Technical report, University of Southampton, 2005.

[23] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of Adjacent Contour Segments for Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(1):36–51, Jan 2008.

[24] S. Fiel and R. Sablatnig. Writer Retrieval and Writer Identification Using Local Features. In *2012 10th IAPR International Workshop on Document Analysis Systems (DAS)*, pages 145 –149, March 2012.

[25] S. Fiel and R. Sablatnig. Writer Identification and Writer Retrieval Using the Fisher Vector on Visual Vocabularies. In *2013 12th International Conference on Document Analysis and Recognition (ICDAR)*, pages 545–549, 2013.

[26] S. Fiel and R. Sablatnig. Writer Identification and Retrieval using a Convolutional Neural Network. In *16th International Conference on Computer Analysis of Images and Patterns (CAIP)*, pages 26–37, 2015.

[27] K. Franke, L. Schomaker, C. Veenhuis, L. Vuurpijl, M. van Erp, and I. Guyon. WANDA: A common ground for forensic handwriting examination and writer identification. *ENFHEX News*, pages 23–47, 2003.

[28] K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, 1980.

[29] C.M. Greening, V.K. Sagar, and C.G. Leedham. Handwriting identification using global and local features for forensic purposes. In *European Convention on Security and Detection, 1995*, pages 272–278, May 1995.

[30] P.S. Hiremath, S. Shivashankar, J.D. Pujari, and R.K. Kartik. Writer identification in a handwritten document image using texture features. In *International Conference on Signal and Image Processing (ICSIP)*, pages 139 –142, Dec 2010.

[31] D. H. Hubel and T. N. Wiesel. Receptive Fields, Binocular Interaction, and Functional Architecture in the Cat's Visual Cortex. *Journal of Physiology (London)*, 160:106–154, 1962.

[32] David H. Hubel and Torsten N. Wiesel. Receptive Fields and Functional Architecture in two Nonstriate Visual Areas (18 and 19) of the Cat. *Journal of Neurophysiology*, 28(2):229–289, 1965.

[33] T. Jaakkola and D. Haussler. Exploiting Generative Models in Discriminative Classifiers. In *In Advances in Neural Information Processing Systems 11*, pages 487–493. MIT Press, 1998.

[34] C. Jacobs, P.Y. Simard, P. Viola, and J. Rinker. Text recognition of low-resolution document images. In *2005 8th International Conference on Document Analysis and Recognition (ICDAR)*, pages 695–699 Vol. 2, Aug 2005.

[35] R. Jain and D. Doermann. Offline Writer Identification Using K-Adjacent Segments. In *2011 11th International Conference on Document Analysis and Recognition (ICDAR)*, pages 769 –773, sept. 2011.

[36] R. Jain and D. Doermann. Writer Identification Using an Alphabet of Contour Gradient Descriptors. In *2013 12th International Conference on Document Analysis and Recognition (ICDAR)*, pages 550–554, Aug 2013.

[37] R. Jain and D. Doermann. Combining Local Features for Offline Writer Identification. In *2014 14th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 583–588, Sept 2014.

[38] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. *CoRR*, abs/1408.5093, 2014.

[39] Y. Jianchao, Y. Kai, G. Yihong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009*, pages 1794–1801, June 2009.

[40] T. Joachims. Text categorization with Support Vector Machines: Learning with many relevant features. In *Machine Learning: ECML-98*, volume 1398 of *Lecture Notes in Computer Science*, pages 137–142. Springer Berlin Heidelberg, 1998.

[41] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *Tenth IEEE International Conference on Computer Vision, 2005. ICCV 2005*, volume 1, pages 604–610, Oct 2005.

[42] E. Khalifa, S. Al-maadeed, M.A. Tahir, A. Bouridane, and A. Jamshed. Off-line writer identification using an ensemble of grapheme codebook features. *Pattern Recognition Letters*, 59:18 – 25, 2015.

[43] R. Khan, C. Barat, D. Muselet, and C. Ducottet. Spatial orientations of visual word pairs to improve Bag-of-Visual-Words model. In *Proceedings of the British Machine Vision Conference*, pages 89.1–89.11. BMVA Press, 2012.

[44] F. Kleber, S. Fiel, M. Diem, and R. Sablatnig. CVL-DataBase: An Off-Line Database for Writer Retrieval, Writer Identification and Word Spotting. In *2013 12th International Conference on Document Analysis and Recognition (ICDAR)*, pages 560–564, 2013.

[45] F. Lauer, C. Y. Suen, and G. Bloch. A trainable feature extractor for handwritten digit recognition. *Pattern Recognition*, 40(6):1816 – 1824, 2007.

[46] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, 2006.

[47] Y. Le Cun, O. Matan, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jacket, and H.S. Baird. Handwritten zip code recognition with multilayer networks. In *Proceedings of the 10th International Conference on Pattern Recognition, 1990, ICPR*, volume ii, pages 35–40, Jun 1990.

[48] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.

[49] X. in Li and X. Ding. Writer Identification of Chinese Handwriting Using Grid Microstructure Feature. In Massimo Tistarelli and Mark Nixon, editors, *Advances in Biometrics*, volume 5558 of *Lecture Notes in Computer Science*, pages 1230–1239. Springer Berlin / Heidelberg, 2009.

[50] T. Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, 21:224–270, 1994.

[51] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang. CASIA Online and Offline Chinese Handwriting Databases. In *2011 International Conference on Document Analysis and Recognition (ICDAR)*, pages 37–41, Sept 2011.

[52] S.P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–136, 1982.

[53] G. Louloudis, B. Gatos, and N. Stamatopoulos. ICFHR2012 Competition on Writer Identification, Challenge 1: Latin/Greek Documents. In *2012 International Conference on Frontiers in Handwriting Recognition*, pages 825–830, 2012.

[54] G. Louloudis, B. Gatos, N. Stamatopoulos, and A. Papandreou. ICDAR 2013 Competition on Writer Identification. In *2013 12th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1397–1401, Aug 2013.

[55] G. Louloudis, N. Stamatopoulos, and B. Gatos. ICDAR 2011 Writer Identification Contest. In *2011 11th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1475–1479, Sept 2011.

[56] D.G. Lowe. Object recognition from local scale-invariant features. In *The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999*, volume 2, pages 1150–1157 vol.2, 1999.

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

[58] X. Lu, D. Xiaoqing, P. Liangrui, and L. Xin. An Improved Method Based on Weighted Grid Micro-structure Feature for Text-Independent Writer Recognition. In *2011 11th International Conference on Document Analysis and Recognition (ICDAR)*, pages 638–642, 2011.

[59] U.-V. Marti and H. Bunke. The IAM-database: an English sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition*, 5(1):39–46, 2002.

[60] U.-V. Marti, R. Messerli, and H. Bunke. Writer Identification Using Text Line Based Features. In *2001 6th International Conference on Document Analysis and Recognition (ICDAR)*, pages 101 –105, 2001.

[61] A. Mezghani, S. Kanoun, M. Khemakhem, and H.E. Abed. A Database for Arabic Handwritten Text Image Recognition and Writer Identification. In *Proceedings of the 2012 International Conference on Frontiers in Handwriting Recognition*, ICFHR '12, pages 399–402, Washington, DC, USA, 2012. IEEE Computer Society.

[62] H. Miklas. Zur editorischen Vorbereitung des sog. Missale Sinaiticum (Sin. slav. 5/N). In H. Miklas, V. Sadovski, and S. Richter, editors, *Glagolitica - Zum Ursprung der slavischen Schriftkultur*, volume XV-XVI, pages 117–129. (OAW, Phil.-hist. Kl., Schriften der Balkan-Kommission, Philologische Abt. 41), 2000.

[63] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, Oct 2005.

[64] B. Nickolay and J. Schneider. Virtuelle Rekonstruktion "vorvernichteter" Stasi-Unterlagen. Technologische Machbarkeit und Finanzierbarkeit - Folgerungen für Wissenschaft, Kriminaltechnik und Publizistik. In *Schriftenreihe des Berliner Landesbeauftragten für die Unterlagen des Staatssicherheitsdienstes der ehemaligen DDR*, volume 21, pages 11–28. Johannes Weberling and Giselher Spitzer, 2007.

[65] E. Nowak, F. Jurie, and B. Triggs. Sampling Strategies for Bag-of-features Image Classification. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part IV*, ECCV'06, pages 490–503, Berlin, Heidelberg, 2006. Springer-Verlag.

[66] K. Ntirogiannis, B. Gatos, and I. Pratikakis. ICFHR2014 Competition on Handwritten Document Image Binarization (H-DIBCO 2014). In *2014 14th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 809–813, Sept 2014.

[67] M. Opitz, M. Diem, S. Fiel, F. Kleber, and R. Sablatnig. End-to-End Text Recognition with Local Ternary Patterns, MSER and Deep Convolutional Nets. In *Proceedings of the 11th International Workshop on Document Analysis Systems*, pages 186–190, 2014.

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

[69] G. S. Peake and T. N. Tan. Script and Language Identification from Document Images. In Adrian F. Clark, editor, *Proceedings of the British Machine Vision Conference 1997, BMVC*, pages 610–619, 1997.

[70] F. Perronnin and C. Dance. Fisher Kernels on Visual Vocabularies for Image Categorization. In *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07.*, pages 1–8, june 2007.

[71] F. Perronnin, G. Dance, C.and Csurka, and M. Bressan. Adapted Vocabularies for Generic Visual Categorization. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part IV*, volume 3954 of *Lecture Notes in Computer Science*, pages 464–475. Springer Berlin Heidelberg, 2006.

[72] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *Proceedings of the 11th European Conference on Computer vision: Part IV*, ECCV'10, pages 143–156, Berlin, Heidelberg, 2010. Springer-Verlag.

[73] A. Petter. Die Rekonstruktion zerrissener Stasi-Unterlagen. Ursachen, Bedeitung und Perspektiven einer besonderen Fachaufgabe. *Journal juristische Zeitgeschichte*, 3:61–64, 2009.

[74] R. Plamondon and G. Lorette. Designing an automatic signature verifier: Problem definition and system description. In *Computer Processing of Handwriting*, pages 3–20, 1990.

[75] M. Ranzato, Fu Jie Huang, Y.-L. Boureau, and Y. LeCun. Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition. In *CVPR '07. IEEE Conference on Computer Vision and Pattern Recognition, 2007*, pages 1–8, June 2007.

[76] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, pages 1–42, April 2015.

[77] V.K. Sagar, S.W. Chong, C.G. Leedham, and Y. Solihin. Slant manipulation and character segmentation for forensic document examination. In *Digital Signal Processing Applications TENCON '96. Proceedings*, volume 2, pages 933–938 vol.2, Nov 1996.

[78] J. Schneider and B. Nickolay. The Stasi puzzle. *Fraunhofer Magazine, Special Issue*, 1:32–33, 2008.

[79] L. Schomaker and M. Bulacu. Automatic writer identification using connected-component contours and edge-based features of uppercase Western script. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):787–798, June 2004.

[80] I.A. Siddiqi and N. Vincent. Writer Identification in Handwritten Documents. In *2007 9th International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 108–112, 2007.

[81] S. N. Srihari, S.-H. Cha, Arora H., and Lee S. Individuality of handwriting. *Journal of Forensic Sciences*, 4:856–872, 2002.

[82] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. *CoRR*, abs/1409.4842, 2014.

[83] G. X. Tan, C. Viard-Gaudin, and A. C.. Kot. Automatic writer identification framework for online handwritten documents using character prototypes. *Pattern Recognition*, 42(12):3313 – 3323, 2009.

[84] W. Tao, D.J. Wu, A. Coates, and A.Y. Ng. End-to-end text recognition with convolutional neural networks. In *2012 21st International Conference on Pattern Recognition (ICPR)*, pages 3304–3308, Nov 2012.

[85] P. Thumwarin and T. Matsuura. On-line writer recognition for Thai based on velocity of barycenter of pen-point movement. In *2004 International Conference on Image Processing, 2004. ICIP '04*, volume 2, pages 889–892, Oct 2004.

[86] L. van der Maaten and E.O. Postma. Improving Automatic Writer Identification. In *Proceedings of the 17th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC)*, pages 260–266, 2005.

[87] L. Zhu, A. B. Rao, and A Zhang. Theory of Keyblock-based Image Retrieval. *ACM Transactions on Information Systems (TOIS)*, 20(2):224–257, April 2002.

[88] Y. Zhu, J. Sun, and S. Naoi. Recognizing Natural Scene Characters by Convolutional Neural Network and Bimodal Image Enhancement. In *Camera-Based Document Analysis and Recognition*, volume 7139 of *Lecture Notes in Computer Science*, pages 69–82. Springer Berlin Heidelberg, 2012.

Stefan Fiel

Personal Data

Date of Birth	20.07.1981
Place of Birth	Feldkirch
Nationality	Austria

Education

Since 2011	Research Assistant at Computer Vision Lab, TU Wien Projects: Document Information Retrieval, Fieldvibes
2010	Graduation, Dipl.-Ing. (comparable with Master of Science), TU Wien, Thesis: Automated Identification of Tree Species from Images of Bark, Leaves or Needles
2009-2010	Study of Visual Computing, TU Wien
2009	Graduation Bachelor of Science
2000-2009	Study of Software and Information Engineering, TU Wien
1991-2000	BG & BRG Feldkirch

Wassergasse 16/14 – 1030 Wien
✉ *stefan@fiel.name*