Πανεπιστήμιο Αιγαίου

Τμήμα Μηχανικών Πληροφοριακών & Επικοινωνιακών Συστημάτων

Αναγνώριση Γραφέα

Παρασκευάς Διαμαντάτος

Επιβλέπων: **Αναπληρώτρια Καθηγήτρια: Εργίνα Καβαλλιεράτου**

Καρλόβασσι – Σάμος - Ελλάδα 2021

Διατριβή

για την απόκτηση Διδακτορικού Διπλώματος του

Εργαστήριού Τεχνητής Νοημοσύνης & Συστημάτων Στήριξης Αποφάσεων του

Τμήματος Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων

UNIVERSITY OF THE AEGEAN

Department of Information and Communication Systems Engineering

WRITER IDENTIFICATION

Ph.D. Dissertation

Paraskevas Diamantatos

Supervisor: **Associate Professor: Ergina Kavallieratou**

Karlovassi – Samos – Greece 2021

Submitted in Total Fulfilment

of the Requirements for

The degree of Doctor of Philosophy (PhD)

Declaration of Authorship

I, Paraskevas Diamantatos, declare that this Dissertation entitled “Writer Identification” and all the work presented here is my own and has been generated as the result of my original research. I confirm that:

- This work was done wholly while in candidature for a research degree at this University.
- It is always clearly attributed where I have consulted the published work of others.
- The source is always given, where I have quoted from the work of others. Except for such quotations, this thesis is entirely my work.
- I have acknowledged all the sources of help.
- The thesis is based on work done by myself jointly with others. I have made it clear precisely what was done by others and what I have contributed myself.

Signed:

Date:

Advising Committee

Associate Professor

Ergina Kavallieratou

Professor

Efstathios Stamatatos

Professor

Manolis Maragoudakis

Examining Committee

Associate Professor : Kavallieratou Ergina

Professor : Efstathios Stamatatos

Professor : Manolis Maragoudakis

Professor : Stefanos Gritzalis

Professor : Athanassios Skodras

Professor : Ioannis Pratikakis

Associate Professor : Theodoros Kostoulas

Abstract

The state-of-the-art writer identification systems use various features and techniques to identify the writer of the handwritten text. In this work, several directional features and combinations of directional with model-based features are presented. Specifically, several improvements of a statistical, directional feature, the edge hinge distribution, are attempted in novel contributions as the Skeleton Hinge Distribution, the Weighted Skeleton Hinge Distribution, the Quantized Skeleton Hinge Distribution, the Directional Stroke Run Length Distribution and the Edge Skeleton Hinge combination. Furthermore, the Skeleton Hinge Distribution feature with a model-based feature is explored, based on a codebook of graphemes.

Novel contributions related to the preprocessing of the document images and the extraction of valuable characteristics are presented. More specifically, two techniques are presented for Main Body Size estimation, a characteristic with application in a broad range of document image analysis fields. One measures Main Body size directly, while the other does an estimation for the baselines first. Both methods are segmentation free. A small collection of 10 printed document images and a collection of handwritten text were used for the presented experimental results.

Furthermore, a technique for text localization is presented that takes advantage of the fact that text should present some contrast in comparison with the background, to be distinguished by the human eye. A procedure of binarization is applied to create appropriate images for text detection. The connected components of the image are extracted, and some heuristic rules are applied to identify areas containing text.

For the evaluation, the Firemaker Database and the ICDAR 2017 writer identification competition dataset were used. A plethora of matching techniques were considered for Skeleton Hinge distribution, including nearest neighbour classifier, K-means, Hierarchical Cluster Tree, k-nearest neighbours and Support Vector Machines. The skeleton hinge distribution achieved an accuracy of 90,8%, while the combination of this method with the codebook of graphemes reached 96%. The Weighted Skeleton Hinge Distribution achieved an accuracy of 91.2%. The Quantized Skeleton Hinge Distribution achieved an accuracy of 92.4%. The Directional Stroke Run Length Distribution achieved an accuracy of 91.2%, and finally, the Edge Skeleton Hinge combination technique achieved an accuracy of 90,2%.

Keywords: Writer Identification, Edge-Hinge Distribution, Skeleton-Hinge Distribution, Codebook of Graphemes

Department of Information and Communication Systems Engineering

UNIVERSITY OF THE AEGEAN

Περίληψη

Τα σύγχρονα συστήματα αναγνώρισης γραφέα χρησιμοποιούν μια ποικιλία διαφορετικών χαρακτηριστικών και τεχνικών για να προσδιορίσουν τον συγγραφέα του χειρόγραφου κειμένου. Σε αυτή την διατριβή παρουσιάζονται διάφορα κατευθυντικά χαρακτηριστικά καθώς και συνδυασμοί κατευθυντικών χαρακτηριστικών με χαρακτηριστικά που βασίζονται σε μοντέλα. Συγκεκριμένα, επιχειρούνται αρκετές βελτιώσεις ενός στατιστικού, κατευθυντικού χαρακτηριστικού, του edge hinge distribution. Τα νέα χαρακτηριστικά που παρουσιάζονται είναι το Skeleton Hinge Distribution, το Weighted Skeleton Hinge Distribution, το Quantized Skeleton Hinge Distribution, το Directional Stroke Run Length Distribution και το Edge Skeleton Hinge Combination . Επιπλέον, διερευνάται ο συνδυασμός του Skeleton Hinge Distribution με ένα χαρακτηριστικό που βασίζετε σε μοντέλα.

Νέες συνεισφορές που σχετίζονται με την προεπεξεργασία των εικόνων εγγράφων αλλα και την εξαγωγή πολύτιμων χαρακτηριστικών του κειμένου. Ειδικότερα, παρουσιάζονται δύο τεχνικές για την εκτίμηση μεγέθους κύριου σώματος (Main Body Size Estimation), το οποίο είναι ένα χαρακτηριστικό του κειμένου με εφαρμογή σε ένα ευρύ φάσμα πεδίων ανάλυσης εικόνων εγγράφου. Η πρώτη μέθοδος μετρά άμεσα το μέγεθος του κύριου σώματος, ενώ η δεύτερη υπολογίζει πρώτα τις βασικές γραμμές (baseline). Και οι δύο προτινόμενοι μεθόδοι δεν απαιτούν τμηματοποίηση (segmentation) . Τα πειραματικά αποτελέσματα παρουσιάζονται σε μια συλλογή χειρόγραφων εγγράφων καθώς και σε μια μικρή συλλογή 10 εικόνων απο πληκτρολογημένα εγγράφα προκειμένου να προκύψουν πιο αντικειμενικά αποτελέσματα. Επιπλέον, παρουσιάζετε μια τεχνική για τον εντοπισμό κειμένου που εκμεταλλεύεται το γεγονός ότι το κείμενο πρέπει να παρουσιάζει κάποια αντίθεση σε σχέση με το υπόβαθρο (background), προκειμένου να διακρίνεται από το ανθρώπινο μάτι. Χρησιμοποιείται μια διαδικασία binarization για τη δημιουργία κατάλληλων εικόνων εγγράφου για την ανίχνευση κειμένου. Στην συνέχεια τα συνδεδεμένα στοιχεία (connected components) της εικόνας εξάγονται και εφαρμόζονται ορισμένοι ευρετικοί κανόνες για τον εντοπισμό περιοχών που περιέχουν κείμενο.

Για την αξιολόγηση της παρούσας εργασίας, η συλλογή χειρόγραφων firmaker DB χρησιμοποιήθηκε. Η συγκεκριμένη συλλογή περιλαμβάνει 4 σελίδες χειρόγραφού κειμένου από 250 διαφορετικούς συγγραφείς. Χρησιμοποιήθηκε μια πληθώρα τεχνικών αντιστοίχισης για το Skeleton Hinge Distribution, συμπεριλαμβανομένου του πλησιέστερου γείτονα, k-means, ιεραρχικών συστάδων (hierarchical cluster trees) , knn και support vector machines. Το χαρακτηριστικό Skeleton Hinge Distribution κατάφερε να ανιχνεύσει τον συγγραφέα χειρόγραφου κειμένου με ακρίβεια 90,8%, το Weighted Skeleton Hinge Distribution με ακρίβεια 91,2%, το Quantized Skeleton Hinge Distribution με ακρίβεια 92,4%, το Directional Stroke Run Length Distribution με ακρίβεια 91,2% και το Edge Skeleton Hinge Combination με ακρίβεια 90,2%.

Department of Information and Communication Systems Engineering

UNIVERSITY OF THE AEGEAN

Department of Information and Communication Systems Engineering

UNIVERSITY OF THE AEGEAN

«Make everything as simple as
possible, but not simpler.»
Albert Einstein

Acknowledgements

I want to thank my PhD Supervisor Ergina Kavallieratou for her support and guidance. The advising and examining committee for their time, effort and support. Professor Stefanos Gritzalis for his support and understanding during the period of my PhD.

I am incredibly grateful to all my colleagues and friends for the nice times we had together during the period of my PhD.

I would also like to thank my girlfriend Klelia. She encouraged me in difficult times and never stopped believing in me.

Most importantly, I want to thank my Family for their support in my entire student's life. Without their help, this PhD Dissertation would not be possible.

List of Publications

- Diamantatos, P., Kavallieratou, E., & Gritzalis, S. (2021). Directional Hinge Features for Writer Identification: The importance of the Skeleton and the effects of character size and pixel intensity. SN Computer Science – Under Review
- Diamantatos, P., Kavallieratou, E., & Gritzalis, S. (2016). Skeleton Hinge Distribution for Writer Identification. International Journal on Artificial Intelligence Tools, 25(03), 1650015.
- Diamantatos, P., Stefanos, G., & Ergina, K. (2014, September). Writer identification using a statistical and model based approach. In *2014 14th International Conference on Frontiers in Handwriting Recognition* (pp. 589-594). IEEE.
- Diamantatos, P., Kavallieratou, E., & Gomez-Gil, P. (2014, September). Binarization: a Tool for Text Localization. In *2014 14th International Conference on Frontiers in Handwriting Recognition* (pp. 649-654). IEEE.
- Diamantatos, P., Verras, V., & Kavallieratou, E. (2013, August). Detecting main body size in document images. In *2013 12th International Conference on Document Analysis and Recognition* (pp. 1160-1164). IEEE.
- Diamantatos, P., & Kavallieratou, E. (2014, September). Android based electronic travel aid system for blind people. In *IFIP International Conference on Artificial Intelligence Applications and Innovations* (pp. 585-592). Springer, Berlin, Heidelberg.

Contents

List of Figures

List of Tables

List of Equations

Chapter 1

1.1Introduction

While our future is digital, our past is analogue. Writing is one of the most important innovations in human history. Our cultural heritage, art, sciences, mythology, religious scripts, poems, certificates, and our entire history can be found in the various historical document collections written over the ages. While these collections are owned by various libraries and private collections worldwide, as hard copies, many historical documents have already become digital in the last decades. The digitization of these documents is far from over since more and more collections are digitized every day.

Each person's handwriting is unique, and therefore it can be used as a biometric characteristic [30]. More specifically, handwriting is considered a behavioural biometric characteristic since it is directly related to how each person grew up. For example, schooling, personal preferences, languages learned, and other characteristics make each writer's handwriting unique. Moreover, handwriting can be affected by other factors like the writing implements used, writing speed, writing surface, and available writing area, resulting in handwritten documents with text characters that may vary in size.

In recent years, most research in person identification mostly targets their biometrics [1, 2, 3]. Two types of biometrics exist, physiological and behavioural. Physiological biometrics identification applications are based on measuring the physical property of the human body. A variety of applications offer person identification through their physiological biometrics like their eye iris, fingerprints, retinal blood vessels, hand geometry, DNA even face identification from an image. The results yield that person identification using physiological biometrics can be considered a solved problem. Behavioural biometrics, on the other hand, uses individual traits of a person's behaviour for identification. Some behavioural biometrics applications include voice identification, signature identification, gait, keystroke dynamics and also handwriting.

Contrary to signature identification [4], which requires a predefined, sort sequence of characters or strokes, writer identification can be achieved by a writer's handwritten text, not only by a predefined one. Moreover, most signature identification systems use online information, meaning that the user signs in a specific area with an electronic pen or some other electronic form of writing aid, which monitors the movements the user does and his time. Unfortunately, writer identification systems cannot use the same information for practical reasons. Most of the samples are written in the paper, meaning the writer's time to write a text is unknown. Furthermore, the direction he travelled when writing is also unknown.

Writer identification is the task of identifying the writer of an unknown handwritten document image by matching it against a database of handwritten documents with a known writer. Features are extracted from the handwritten document image, and either a statistical analysis of these features is entailed, and then their distances are measured, or the features are used to construct models, which are later compared, to achieve identification.

In forensic practice, the identification of a writer is a problem that often arises in a court of justice to identify the writer of a handwritten document [5], a will, for example. It also has applications in the health sector where a prescription writer must be verified [6]. While in forensics [7], writer verification is most common and usually is performed by human experts, writer identification can also be beneficial. For example, in cases of threats, or ransom letters a graphologist, tries to verify the writer's identity with his handwriting texts. This happens when there is a suspect for the case, and his handwriting texts are taken as evidence. The above procedure can be automated if a writer identification system is applied to an extensive data set and output a list of top-ranked writers. Then the results can be either verified by a writer verification system or a human expert.

Writer identification and writer verification are some terms that usually get confused [8]. Writer identification systems attempt to match handwritings of unknown writer against a dataset of handwritings with a known writer. These systems can identify a writer of the handwritten text based on other handwritten text samples from the same writer. Moreover, writer identification systems perform one too many searches in an extensive database with handwriting samples of a known writer and return either one or a list of candidate writers. Writer identification can also be applied to the area of optical character recognition by exploiting the writer's style and adapting the recognition system to the type of the writer [9].

On the other hand, on writer verification systems, the goal is to do a one to one identification. A decision must be made if two specific handwritten text samples belong to the same writer. In this method, usually, the distance between the two samples is measured, and if it is below a specific threshold, then the two samples are from the same writer.

Writer identification and writer verification fall into two broad categories: text-dependent and text-independent [10]. Text-dependent methods share many similarities with signature verification techniques since they compare a predefined set of characters or words of known semantic meaning with the ones in the handwritten sample in question. Text-dependent methods require human intervention to segment characters or words correctly. On the other hand, text-independent methods use statistical features extracted from the samples without any human intervention. In this work, the main focus will be given in text-independent techniques.

Writer identification techniques can be divided into three broad categories, statistical techniques that use textural [8, 11] or structural based features [12] and model-based approaches that extract features automatically from raw data without explicit programming.

Statistical techniques usually entail a statistical analysis of features extracted from the directionality and curvature or structure of patterns in handwritten document images. In textural features [11], the handwritten document is treated as an image and not as handwriting, and usually, the analysis of the foreground texture is entailed to extract features. In structural features, the extracted features are mainly based on characteristics of the writing that even a human reader can distinguish, such as the text's main body size, the height of upper and lower Baselines, character width, and text slant.

Model-based techniques can extract features automatically by using various Artificial-Intelligence techniques like Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), Extend

Learning Model (ELM), other deep machine learning models, or allograph approaches [8] that construct models using Self Organizing Feature Maps (SOFM).

This work addresses the problem of offline, text-independent writer identification using scanned handwritten document images. The methods presented here are statistical techniques that capture textural features.

Our methods are statistically evaluated using the Firemaker data set [13] and the ICDAR, 2017 writer identification competition dataset [14].

1.2Motivation and Objectives

The most prominent application of writer identification is in forensics and as evidence in court trials [15]. Recently writer identification techniques [16] were utilized in palaeography to prove that The Great Isaiah Scroll (1QIsaa), one of the original seven Dead Sea Scrolls discovered in Qumran in 1947, was written by two distinct writers carefully mirroring another scribe's writing style. Future advancements in this field may allow us to use it in applications such as OCR, identifying the writer of anonymous historical documents, and authentication systems.

Although relatively recent works on the writer identification field utilize artificial intelligence techniques [17, 18, 14, 19, 20, 21, 22], a choice was made to use traditional methods only for several reasons.

More specifically, the datasets available for the writer identification task do not contain enough samples and are pretty limited in size. In [18], This issue is identified in the open research issues section as the cause for the scarcity of CNN based writer identification systems in the literature and as a problem that highly affects the performance of deep learning models. Some researchers [22] try to overcome those limitations by using another Artificial Intelligence technique to generate thousands of samples per writer. Other researchers [20, 21] either utilise the annotations available on Datasets to split entire pages into words or proceed with word segmentation techniques and then try to achieve writer identification on the word level.

Moreover, with the recent European Union fit for the Digital Age [23], the EU Commission proposes new rules and actions for excellence and trust in Artificial Intelligence. According to these new rules, biometric identification systems are considered high risk and subject to strict requirements. One of those requirements is using high-quality datasets to minimize risks and discriminatory outcomes.

Furthermore, the dilemmas in applying artificial intelligence methods in digital palaeography, as presented in [24], also reflect our difficulties in using artificial intelligence techniques for critical writer identification applications. In our view, a system that could be used in forensics and as evidence in court trials must be easily explainable, understood, and most importantly, trusted.

Directional methods [25, 26, 27, 28] fulfil the above criteria since they can be easily explained, understood and ultimately trusted. Moreover, they are computationally efficient and fast and could

be even run on mobile devices if such a need arises. Their requirements for training data is minimal since only one page of handwritten text is sufficient, in most cases, to capture the necessary characteristics, or the feature vector, of a writer.

In our work on Skeleton Hinge Distribution [29], the skeleton information was used to make the feature extraction faster. However, by considering only the Skeleton, a big part of the available pixel information is discarded. This choice has motivated us to investigate if any additional information that could help identify the writer lies in discarded information.

Furthermore, our work on Detecting the Main Body size [30] made apparent tha Main Body size fluctuations could be observed even in a single text written by one writer. This observation motivated us to utilize this information and explore its contribution towards identifying the writer.

Finally, on [21], they observed that neural networks trained on grey-scale images performed better than neural networks trained on binarized and contour images indicating that texture information is an essential factor for writer identification. This observation motivated us to investigate if the same can be observed in Directional methods.

In a nutshell, the objectives of this thesis are as follows

Objective 1: We aim to make advancements in the preprocessing of handwritten document images that will allow us to reduce the noise and extract valuable characteristics of the text.

Objective 2: We aim to make advancements in the directional feature extraction techniques and propose some new feature extraction methods.

Objective 3: We aim to experiment with various matching techniques for our features to understand better how the different matching techniques affect identification accuracy.

Objective 4: We aim to evaluate the choices made in our work regarding the skeletonization process and find out if there is a loss of information and the effect it has on performance.

Objective 5: We aim to experiment with pixel intensity and character size fluctuations to understand better how they affect identification accuracy.

1.3Main assumptions

The written text consists of several pen strokes applied with some force on a medium like paper. While those ink strokes represent a single line to a human observer, when that text is digitized, the same ink stroke is defined into several pixel lines. To make things worse, differences in pen ball size and the angle or surface of writing may produce significant variations in the number of pixel lines produced during the digitization process and differences in the character sizes.

The main assumption done in this work is that all stroke widths, i.e., line thickness, should be the same size. Practically this means that an attempt is made to condense all the writing information in 1-pixel width strokes.

Furthermore, an assumption is made that even if all the available information is used in 1-pixel width fragments, the accuracy should not significantly deviate from the Skeleton Hinge technique accuracy.

Similarly, an assumption is made that the main body size variance affects the identification accuracy since Hinge angles are related to the size of characters. Imagine, for example, the character "o" with two different main body sizes. This can be represented by capital "O" and a small "o". The directional angle of writing on the small "o" is smaller than the directional angle of writing on the capital "O".

Moreover, an assumption is made that noise produced by the writer could affect identification accuracy. This kind of noise either consists of the writers attempt to erase with ink what he has written by mistake or smaller ink stains that resemble salt and pepper noise.

Finally, an assumption is made that the pressure applied to the medium by the writer can also affect accuracy. Since this work deals with offline writer identification, a further assumption is that the pressure can be represented by pixel intensity on the grey-scale document image.

1.4Contributions

In this work, several directional features and combinations of directional features with model-based features are presented. Specifically, several improvements of a statistical directional feature, the edge hinge distribution, are attempted in novel contributions as the Skeleton Hinge Distribution, the Weighted Skeleton Hinge Distribution, the Quantized Skeleton Hinge Distribution, the Directional Stroke Run Length Distribution and the Edge Skeleton Hinge combination. Furthermore, the combination of the Skeleton Hinge Distribution feature with a model-based feature is explored based on a codebook of graphemes.

Novel contributions related to the preprocessing of the document images and the extraction of valuable characteristics are presented. More specifically, two techniques are presented for Main Body Size estimation, a characteristic with application in a broad range of document image analysis fields. One measures the Main Body size directly, while the other first estimates the baselines. Both methods are segmentation free. In order to give more objective results, experimental results are presented over a small collection of 10 printed documents and a collection of handwritten text.

Furthermore, a technique for text localisation is presented, which takes advantage of the fact that text should present some contrast in comparison with the background to be distinguished by the human eye. A procedure of binarisation is applied to create appropriate images for text detection.

The connected components of the image are extracted, and some heuristic rules are applied to identify areas containing text. This is further used as a preprocessing step to clean the document image from the noise that is not part of the text.

1.5 Overview

In chapter 2, state of the art in writer identification is presented. Furthermore, an overview of the anatomy of the handwritten document image is given. Finally, significant terms and techniques related to document image analysis, in general, will be presented.

In chapter 3, two novel approaches for estimating the Main Body size and a technique for noise-cleaning through text localisation will be presented.

In chapter 4, statistical and model-based features used for writer identification are presented. Individually, Edge Direction Distribution, Edge Hinge Distribution and Edge Hinge Combinations, along with our contributions, the Skeleton Hinge Distribution, the Weighted Skeleton Hinge Distribution, the Quantized Skeleton Hinge Distribution, the Directional Stroke Run Length Distribution and the Edge Skeleton Hinge combination are presented. Furthermore, a Model-Based feature that only considers closed areas of the characters is presented.

In chapter 5, an interpretation of the feature vector characteristics produced by the directional methods mentioned above is presented. Furthermore, a detailed explanation of how matching is achieved and what happens on false identifications is provided.

In chapter 6, the data set used to evaluate this work is presented along with experimental results from our feature extraction techniques. More specifically, experimental results on Skeleton Hinge Distribution using the Nearest Neighbour classifier, K-means, Hierarchical Cluster Tree, K-Nearest Neighbours and Support Vector Machines are presented. Moreover, results on Quantized Skeleton Hinge Distribution, Weighted Skelton Hinge Distribution, Run Length Directional Hinge and Edge Skeleton Hinge Combinations, and Codebook of Graphemes combined with Skeleton Hinge Distribution are presented.

Finally, in Chapter 7, our conclusion is drawn.

Chapter 2

2.1State of the art

In this section, a review of recent papers published on the topics of writer identification and some from writer verification are presented. Writer verification was chosen because some of the feature extraction techniques developed for writer verification can also be used in writer identification. Several approaches exist in the literature for writer identification. First, works that entail a classical method, i.e. statistical or model-based strategies, are reviewed—followed by the most recent works based on artificial intelligence and deep learning.

Said et al. [11] proposed a text-independent approach for writer identification that derives writer-specific texture features using multi-channel Gabor filtering and Gray-Scale Co-occurrence Matrices. This method requires uniform blocks of the text created by word deskewing, predefined thresholds of the distance between text lines, words and text padding. Two small sets of 20 writers, with a large number of 25 samples of handwriting text per writer, are used in the evaluation. The Nearest-centroid classification using weighted Euclidean distance and Gabor features achieved an accuracy of 96%. One of the main issues of this approach is the large number of sample pages required per writer.

Zois and Anastassopoulos [31] proposed a method for writer identification using a single word. They apply image thresholding and curve thinning, resample the horizontal projection profiles, and then use morphological operators to obtain 20 dimensional feature vectors classified using a Bayesian classifier. Experiments performed on a single word, the word ”characteristic”, written 45 times by each writer, both in English and in Greek. The dataset consisted of 50 different writers. The reported accuracy of this method is 95%.

Srihari et al. [32] on a writer verification approach, proposed a considerable number of features divided into two categories. Macro-features, which operate at document, paragraph and word level. Also, Micro-features, which operate at word and character level. The macro-features are based on grey-level entropy and threshold, number of ink pixels, number of interior and exterior contours, number of 4-direction slope components, average height and slant, paragraph aspect ratio and indentation, word length and upper and lower zone ratio. The Micro-features utilize gradient, structural, and concavity attributes. The proposed system considers two handwritten document images and outputs a decision made if the two input images are from the same writer or a different one. Experimental results were performed on a dataset containing 1000 writers who copied a fixed text of 156 words (the CEDAR letter). This writer verification method achieved on same writer accuracy of 94.6 % while different writer accuracy was 97.6 %.

Bensefia et al. [33] use graphemes generated by a handwriting segmentation method to encode the individual characteristics of handwriting. These Graphemes are then clustered to define a feature space common for the document set. Grapheme clustering is used to define a feature space common for all documents in the dataset. The reported experiment results achieved an accuracy of 90 % on a dataset consisting of 88 writers (PSI) and 68 % on a dataset of 150 writers (IAM).

Schomaker et al. [34] compute fragments of connected-component contour classified to identify the writer. A codebook of graphemes is generated by training a Kohonem SOFM on a large number of grapheme contours. Later graphemes are extracted from each document and matched with the graphemes in the codebook. A histogram of graphemes for every document is generated. Experimental results achieved an accuracy of 95 % on ten writers and 83 % on 215 writers. When combined with Edge Directional features 97% accuracy is achieved.

Laurens van der Maaten et al. [26] improved edge hinge directional features by using a combination of window sizes while combining these features with a codebook of graphemes achieved 97% identification accuracy. The edge hinge combinations methodology proposed achieved 81% identification accuracy on the Firemaker dataset, which consists of 250 writers.

Schlapbach and Bunke [35] used HMM to identify and verify writers. Single writer recognisers are specialised by training using only handwriting originating from the chosen writer. More specifically, the output log-likelihood scores of the HMMs were used to identify the writer on handwritten text lines of varying content. This method achieved 96% identification accuracy and 2.5% error in verification accuracy as reported on a subset of the IAM database containing 100 writers, 5 handwritten pages per writer.

Pervouchine and Leedham [36] proposed a writer identification scheme based on high frequent characters. The high frequent characters ('f', 'd', 'y', 'th') are identified and used to determine the writer. Characteristics like height, width, height to width ratio, height of ascenders and descenders, stroke angle, slant angle etc. Experimental results achieved an identification accuracy of 58 %.

Bar-Yosef et al. [37] proposed a method for writer identification applied to historical Hebrew calligraphy documents based on topological features. While his approach seems similar to Pervouchine and Leedham approach, they use three high frequent Hebrew characters only to identify the writer. Connected components for tracing background, the convex hull of the characters, the ratio between background and convex hull, concavity, compactness, etc. are some of the features used in this approach. The reported experimental results achieved an accuracy of 100 % on 34 writers.

Li et al. [38] proposed a method for text-independent online handwriting writer identification. They used the feature vector of hierarchical structure in shape primitives and the dynamic and static feature for writer identification for English and Chinese documents. Experimental results achieved an identification rate of 91.5% with datasets in Chinese text and 93.6% in English text. It is an exciting methodology with the drawback that it cannot be applied to offline writer identification because the direction of the stroke of the writer is unknown.

He et al. [39] developed a technique for off-line, text-independent writer identification of Chinese handwriting documents. He applied the Gabor filter to extract features from the text and incorporated a Hidden Markov Tree in the wavelet domain. Experimental results achieved an accuracy of 36.4%, on a dataset of 500 writers with two handwritten text documents each.

Yan et al. [40] utilise spectral feature using Fast Fourier Transformation to identify the writer of Chinese text. The identification rate achieved in this method is 64 % on 100 writers.

Bulacu et al. [41] developed a text-independent writer identification method for Arabic text. They use textural and allographic features to define a probability distribution function and apply the nearest neighbourhood classifier using them as a distance measure.

Al-Dmour et al. [42] identify writers in Arabic using different feature extraction methods such as hybrid spectral-statistical measures (SSMs), multiple-channel (Gabor) filters, and the grey-level co-occurrence matrix (GLCM) were verified to find the best subset of features. They experimented with various classifiers to rank the extracted features.

Wu et al. [43] proposed a method that is based on scale-invariant feature transform (SIFT) in three stages of training, enrollment, and identification. An isotropic LoG filter is utilized to segment the image to word regions from where SIFT descriptors are extracted into a codebook. Scale and orientations are used to construct an orientation histogram. A distance metric is used for matching. Experimental results on Firemaker dataset achieved 92.4% accuracy.

Nicolaou et al. [44] developed a generic text-as-texture classification scheme where Sparse Radial Sampling Local Binary Patterns are constructed in histograms for different radius. In their Barcelona variation used in ICDAR 2017 writer identification competition [14] they used 12 radii to create histograms normalized with a PCA transform.

Mohammed et al. [45] proposed the Local Naïve Bayes Nearest-Neighbour (Local NBNN) classifier. In this method, the SIFT algorithm is used to detect and describe critical points with a constrain that considers the particularity of handwriting patterns and prevents irrelevant points to be matched. Normalisation is also proposed to cope with unbalanced data. In their Hamburg variation used in the ICDAR 2017 writer identification competition [14], they used NBNN instead of the local NBNN without normalization.

Newell et al. [46] proposed the oriented Basic Image Feature Columns (oBIF Columns) that entail a mixture of allograph and texture-based method that encodes a writer's deviation from the mean encoding of a population of writers with the Delta Encoding.

Abdeljalil et al. [47] developed a method that uses oriented Basic Image Features (oBIFs) that labels locations in the document images into seven symmetry classes for several orientations. Column histograms are constructed, and a distance metric is used for matching. In their Tebessa I and II variations used in ICDAR 2017 writer identification competition [14] they used city block distance.

Faddaoui and Hamrouni [48]applied 16 Gabor filters for handwriting texture analysis and Gazzah, and Ben Amara [49] applied spatial-temporal textural analysis. Al-Maʻadeed et al. [50] identify Arabic writers using only 16 words. They utilise edge-based directional features and three edge-direction distributions of different sizes.

Chahi et al. [51] suggested using a Block Wise Local Binary Count (BW-LBC) operator, which represented the writer by a set of histograms calculated from all the connected components in the text and is based on the occurrence distribution of pixels in small blocks. Nearest-neighbour classification using the Hamming distance was utilized for matching.

Chahi et al. [52] proposed Cross multi-scale Locally encoded Gradient Patterns (CLGP). This new feature extraction technique that represents better salient local writing structure operates at connected component sub-images of the writing sample. Then CLGP histogram feature vectors computed from all these observation regions in all writing samples, and the Nearest Neighbor Classifier is used for matching. Accuracy results reported in this work for the Firemaker database achieved an accuracy of 97.60%, showing that traditional methods are still useful for writer identification.

Fiel and Sablating [17] suggested a convolutional neural network (CNN) method for writer identification. A CNN-based feature vector was generated for each writer and compared with the precalculated feature vectors stored in the database using nearest neighbour classification.

Xing and Qiao [53] proposed DeepWriter, a deep multi-stream CNN that takes local handwritten patches as input and is trained with softmax classification loss.

Tang and Wu [22] suggested using a CNN and joint Bayesian at two stages, feature extraction and writer identification. In the first stage, and because much data is needed to train an effective CNN, an augmentation technique is devised that generates thousands of handwriting images for each writer. These generated images are then used to train the CNN model while the joint Bayesian method is utilized for writer identification.

Khan et al. [54] suggested an offline text-independent writer identification system, which combined SIFT (Scale Invariant Feature Transform) and RootSIFT descriptors in a set of Gaussian mixture models (GMM). They reported an accuracy of 97.98% on the Firemaker dataset.

He and Schomaker [19] suggested using an end-to-end multi-task neural network with one or several adaptive convolutional layers with two types of information. Explicit information includes data like lexical content or word length and implicit attributes such as the author's identity. Their method performs writer identification on word level by resizing all word images to 120×40×1.

He and Schomaker [20], in a later work, proposed FragNet. This deep neural network entails two pathways: a feature pyramid that is first used to extract feature maps and then a fragment pathway based on fragments extracted from the input image and the feature maps from the feature pyramid. In this work, word images were used to achieve writer identification on the word level. Their method achieved 57.5% accuracy on word images from the Firemaker dataset.

Finally, He and Schomaker [21] suggested the global-context residual recurrent neural network (GR-RNN) method. This work utilizes an end-to-end neural network that jointly integrates global-context information and a sequence of local fragment-based features. A global average pooling step is used at the tail of the neural network to acquire the global-context information, while a low-level deep feature map is used to extract the local fragment-based features. A recurrent neural network (RNN) is used to model the spatial relationship between the sequence of fragments and to strengthen the discriminative ability of the local fragment features. They reported an accuracy of 98.8 % on the Firemaker Dataset that is the best performance reported on the literature for the Firemaker Dataset.

2.1.1 State of the Art on Directional Features

Directional features have been researched extensively in the literature and can be divided into two main approaches, the statistical approach, where a statistical analysis of the features is entailed, and the model-based approach, where a codebook is generated from the features.

Bulacu et al. [55] proposed the Edge Directional Distribution (EDD) and the Edge-Hinge Distribution (EHD) features. While Edge directional distribution considers the direction of a single edge fragment, the Edge-Hinge distribution considers the directions of two edge fragments emerging from a central pixel of a sliding window. A probability distribution of the directions detected is generated for every writer in the train data set and then for every writer in the test data set. The generated distributions from the test data set are matched against the generated distributions of the train data set using the nearest neighbourhood algorithm. Experimental results reported an accuracy of 35% for the Edge directional distribution and 63% for the Edge-Hinge directional distribution on the Firemaker DB [13].

Laurens van der Maaten et al. [26] suggested an improved Edge Hinge Directional feature, the Edge Hinge Combinations (EHC), by combining various sliding window sizes in a single feature. Experimental results achieved an identification accuracy of 81%, on the Firemaker DB.

Brink et al. [56] suggested the Quill feature, a probability distribution of the local relation between ink direction (φ) and ink width (w). Furthermore, a variation of this feature was also suggested, the Quill-hinge which records the ink width in conjunction with the two directions ($\varphi 1$) and ($\varphi 2$). While the Quill feature achieved 71% accuracy, the Quill-hinge achieved 86% accuracy, both on the Firemaker DB.

He and Schomaker [28] proposed two directional features. The CoHinge feature is defined as the joint distribution of the Hinge kernel on two different pixels of writing contours and the QuadHinge feature defined as the joint distribution of angles, along with the curvature information of contour fragments. The CoHinge feature was used in the ICDAR 2017 writer identification competition [14] in the method Groningen, achieving an accuracy of 76.1%.

Finally, He et al. [57] proposed a model-based approach for junction detection by using the stroke length distribution in every direction around a reference point inside the ink of texts. A codebook-based representation of the junctions detected is constructed and used for writer identification achieving an accuracy 80.6% on the Firemaker DB.

2.2 Handwritten Document Image Anatomy

Many characteristics can describe handwritten documents. The characters and words have unusual shapes and sizes. Their layouts and the skew of the text are not uniform and depend on the writer. The text lines do not follow a straight line but tend to have a curvature. Text lines may interfere with each other or might be physically connected with other lines. The interline spacing is not uniform. For an example of handwritten documents with the above problems, see Fig. 1 and 2.

While all the above are considered a problem for tasks like the automatic reading of these documents, they are regarded as features for the task of writer identification since they can describe the specific writer of the document.

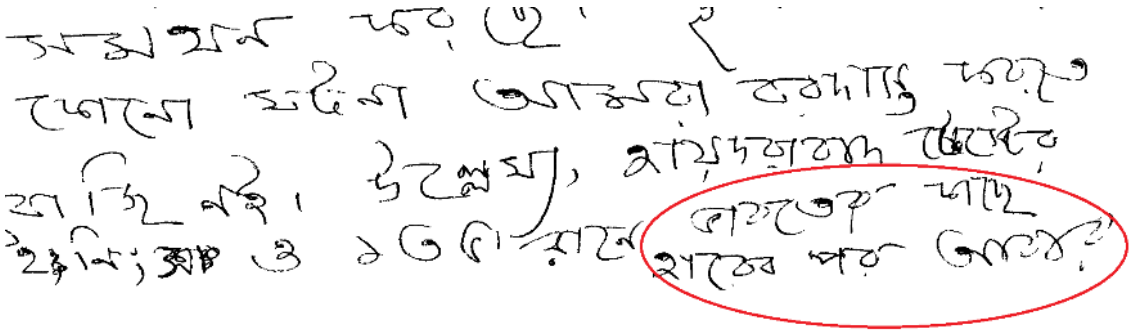

Figure 1. Example of a part of a handwritten historical document image from the ICDAR 2013 Handwriting Segmentation Contest [10] benchmark dataset with interfering lines (ellipse), non-uniform skew, non-uniform interline spacing and text lines with curvature.

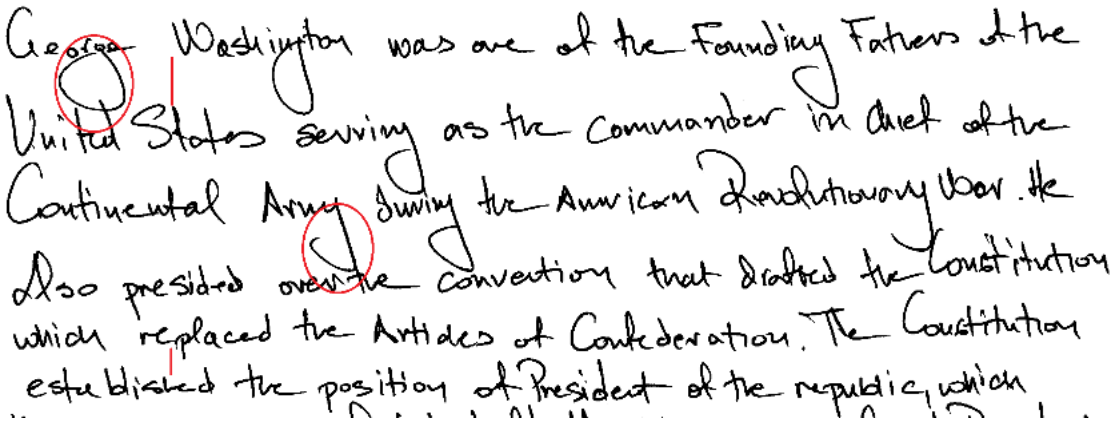

Figure 2. Example of a part of a handwritten historical document image from the ICDAR 2013 Handwriting Segmentation Contest [10] benchmark dataset with connected lines (circles), non-uniform skew and non-uniform interline spacing (line).

Handwritten documents do not have a standard form of writing or any uniform layout. The text line structure is the most dominant structure of these documents. A handwritten document image can be viewed as a text area that consists of text lines. Every text line also consists of one or more words, while every word can be seen as a set of characters in order. Characters, in their turn,

consist of black pixels. This work assumes that document images have a white background black foreground (text).

While humans can easily distinguish the text lines, the mechanisms of the inherent ability are a fantastic feature of the human brain that still is an unsolved problem for computer algorithms. Even when the handwritten document image is seen by a great distance, while the characters and the words are still blurry, the human brain can still distinguish the distinct lines that form the text. For example, see Fig. 3 and Fig. 4, two images from the ICDAR 2013 Handwriting Segmentation Contest are scaled to 5% of the original image size. While it is still hard for a human to read the exact text, it is quite easy to try to segment the different text lines.

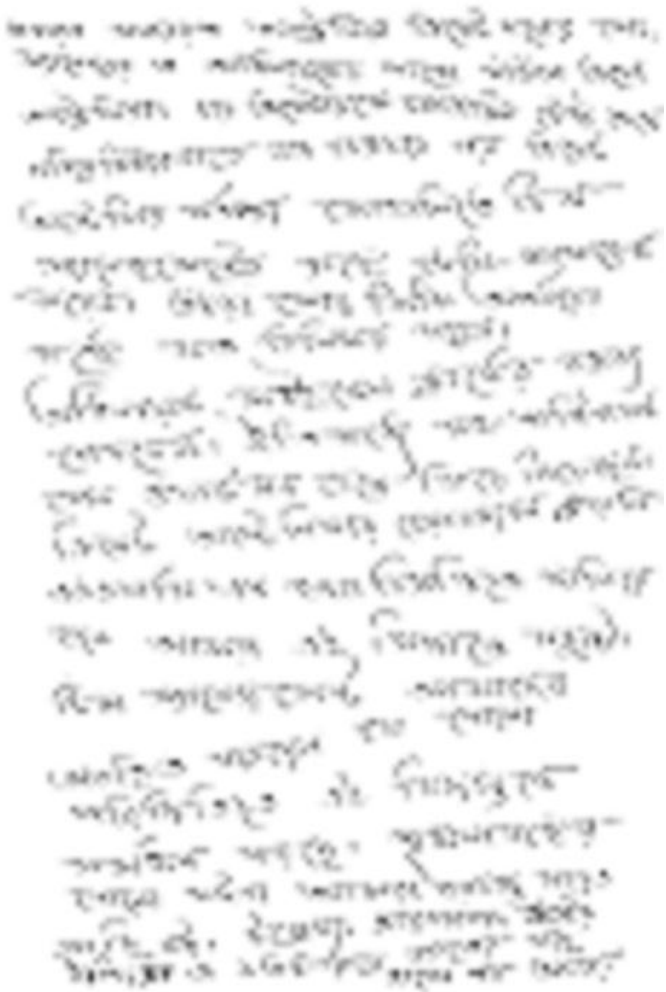

Figure 3. Example of handwritten historical document images from the ICDAR 2013 Handwriting Segmentation Contest [10] benchmark dataset, scaled to 5% of the original size. Text lines can be distinguished from a human even on this scale.

Figure 4. Example of handwritten historical document images from the ICDAR 2013 Handwriting Segmentation Contest [10] benchmark dataset, scaled to 5% of the original size. Text lines can be distinguished from a human even on this scale.

2.2.1 Terms Definition

In this section, the different terms associated with the physical structure of handwritten documents are presented. For graphical representations, please see Fig. 5 and Fig. 6

- A stroke is considered the movement of a writing instrument (pen) on a writing surface (paper)
- The baseline of text is the fictional line which follows the lower part of characters
- The median line of text is the fictional line that follows the upper part of the character
- The upper line of text is the fictional line which follows the upper parts of ascenders
- The lower line of text is the fictional line which follows the lower parts of descenders
- The main body of the text is the size between the baseline and the median line

- The ascenders are the parts of lowercase characters that lie above the median line.
- The descenders are the parts of lowercase characters that lie below the baseline.
- A Component is considered a single character or several connected characters that form a word in this work. Component, in a more general term, is regarded as the connected pixels with similar intensity values.
- Overlapping components are the ascenders or descenders that are in the region of the line above or below.
- The term Touching components means the ascenders or descenders that are physically connected with a part of the text line above or below.

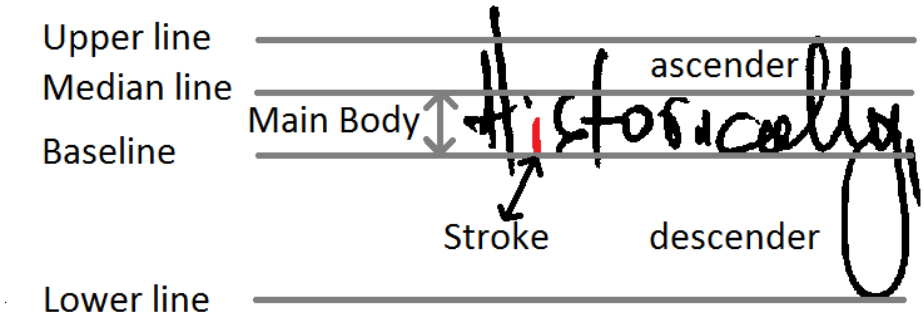

Figure 5. Upper line, Median line, Baseline, Lower line, a single stroke, ascenders and descenders.

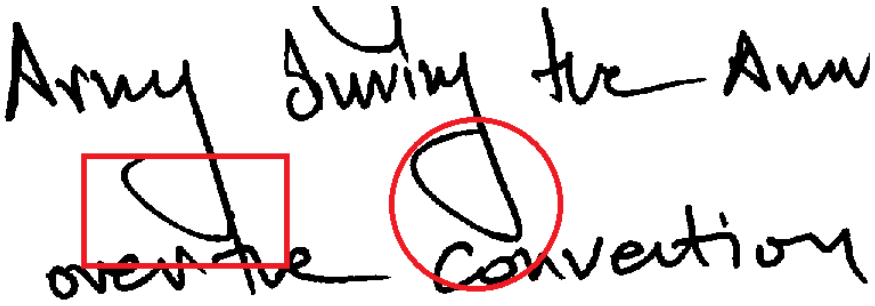

Figure 6. Square: touching components. Circle: overlapping components

2.3 Document Image Analysis

In this section, an attempt is made to give all the necessary definitions for the methods described in the next chapter, to be more easily understandable. A basic introduction of what is a digital image is given. Furthermore, some interesting image analysis techniques like image binarisation, edge detection, Gabor filters, skeletonisation, connected components, contour

tracing, main body, run lengths, Fourier transformation and text localisation will be briefly presented.

2.3.1 Digital Image

When a document image is scanned, it is transformed, through a process of digitisation, to a digital image. This digital image is, in fact, a numeric representation of a two-dimensional matrix if the image is digitised to contain only the grayscale representation. Alternatively, it can be represented to a three-dimensional matrix if the image is digitised to include all the available colour information. For the digitisation process, the image is first sampled on a discrete grid, and then each sample, or pixel, is quantised using a finite number of bits. A computer processes the digitised image. For example, in Fig.7 a fragment of a document image is presented, with the word "The".

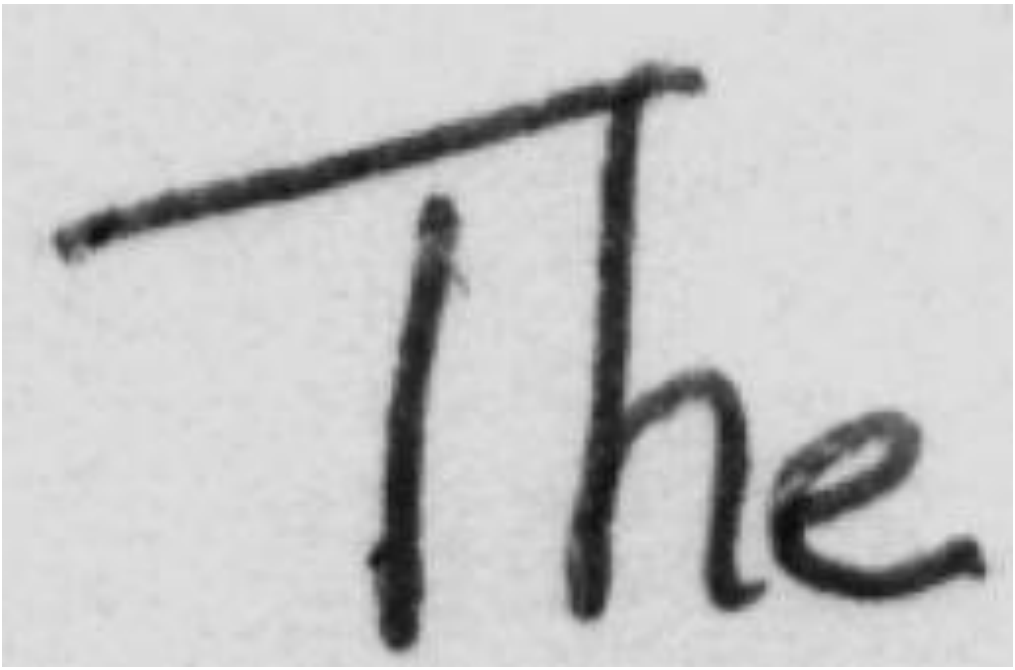

Figure 7. Fragment of a digital image with the word "The"

In the process of scanning or digitisation in general, the image is viewed as small elements, called pixels. A matrix of pixel intensity is stored that can later represent the scanned image. For example, if a zoom-in, on a digital image is attempted, at some point, the distinction between the different pixels the image consists, can be observed. For example, see Fig 8.

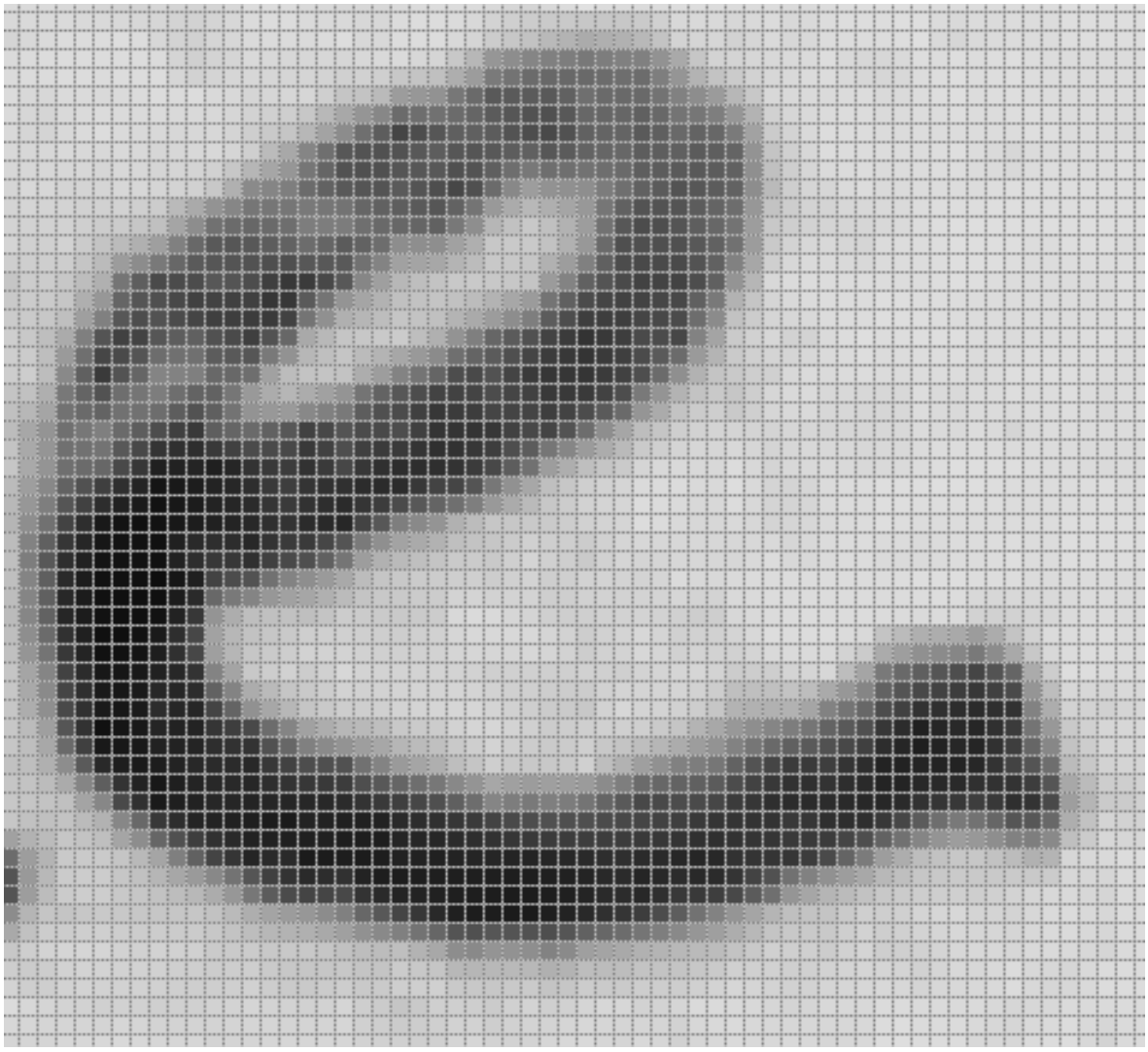

Figure 8. Fragment of a digital image with the character "e", zoomed to the pixel level

This pixel intensity matrix is, in fact, a numerical matrix. Each pixel intensity value stored in this matrix represents how bright a pixel will appear on a screen. The higher the intensity value of a pixel is, the whiter it will look. Grayscale images use 8-bit integers to store pixel values. This means that a pixel can have a value between 0 and 255. In Fig. 9 the character "e" is represented in a pixel intensity matrix. It is the same character "e" as viewed in Fig. 7 and Fig 8, only scaled at 30 %.

220	221	221	222	222	221	221	218	219	220	221	221
218	219	219	221	221	221	221	220	218	220	220	222
216	221	222	221	223	220	219	219	221	222	220	222
216	215	220	222	221	224	224	221	220	221	222	219
219	222	223	177	141	132	170	218	218	217	218	219
201	140	96	83	68	97	98	137	220	220	219	220
118	74	75	92	136	115	83	95	211	219	219	220
102	102	128	189	210	105	71	111	217	214	219	222
67	171	212	189	125	60	60	159	224	221	221	220
194	192	130	78	48	46	110	203	215	219	220	219
105	79	48	50	67	127	204	211	214	213	216	218
46	37	55	102	186	218	221	218	212	219	221	222
61	126	190	210	209	210	216	215	216	220	220	222
191	207	207	216	209	209	210	213	223	225	220	212
218	216	213	212	208	208	216	208	218	182	99	71
145	181	208	220	211	214	195	152	115	72	28	24
34	51	87	133	140	117	81	65	38	30	37	52
22	18	21	26	30	38	32	33	63	110	176	191
149	116	69	44	44	66	96	140	195	211	213	215
208	208	211	198	195	207	210	216	211	214	212	216
214	211	207	214	214	216	213	218	216	215	222	217
216	216	219	215	214	214	218	220	219	219	221	219
211	215	218	218	209	220	215	216	215	218	212	215
217	218	219	216	216	219	216	215	214	219	217	214

Figure 9. the character "e" represented in a pixel intensity matrix

2.3.2 Binarisation

An image can be binarised using a threshold [58], to consist only of values 1 or 0, white or black. For example, with a threshold of 150 in fig 3, the values that are smaller than 150 are set to 0, and values greater or equal than 150, are set to 1. This grayscale image will be transformed into a binary one with the above process, although this is just one method for binarizing an image using a global threshold. More complicated methods exist that use adaptive [58] thresholding and other techniques. In Fig. 10, an example of the image in Fig. 9, binarised using a global threshold, is presented.

1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	0	0	0	0	0	1	1
1	1	0	0	0	0	1	0	0	0	1
1	0	0	0	1	1	1	0	0	0	1
0	0	0	1	1	1	0	0	0	1	1
0	0	1	1	1	0	0	0	0	1	1
0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1	1	0
0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	1	1
1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1

Figure 10.The character "e" represented in a binary pixel intensity matrix

Binarisation is a ubiquitous pre-processing task of image processing, which reduces the size of the image and allows fast and easy calculations as well as further processing of an image. It is considered a mandatory task in many computer vision systems [59] and several works have used binarisation as an aid for text detection [60, 61, 62, 63].

2.3.3 Edge Detection

Edge detection [64] is the process of identifying the specific points in a digital image where the image brightness suddenly changes or has discontinuities. Those points are typically organised into a set of curved line segments named edges. With Edge detection, the boundaries of objects in an image are indicated, making the methods of Edge detection a fundamental tool in image processing, machine vision and computer vision. The output of an edge detection algorithm is a binary image, with only the edge pixels having a value of 1. For an example of applying an edge detection algorithm, and more specifically Sobel edge detection, see Fig 11.

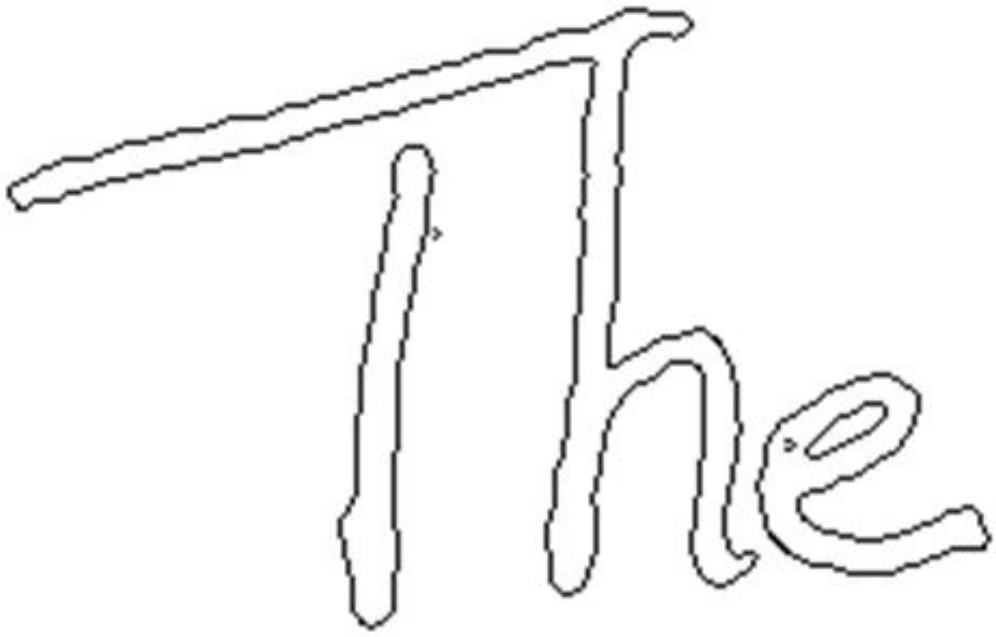

Figure 11. The output of edge detection on the image of Fig. 7.

2.3.4 Gabor Filter

Gabor filters [65] are orientation-sensitive filters used for texture analysis. They typically traverse an image in multiple directions. A Gabor filter, set at a direction, will give a strong output for locations of the target images that have structures in this given direction. For example, suppose the target image is made of a periodic grating in a diagonal direction. In that case, a Gabor filter set at a direction will give a strong output only if its direction matches one of the gratings. Gabor filters have also been used to localise and extract edges since edges are composed of higher frequency components, whereas other image regions are relatively smooth. Frequency and orientation representations of Gabor filters are quite similar to those of the human visual perception, and they are appropriate for texture identification and representation. Gabor filters have many practical applications. They are mostly used in character recognition and fingerprint enhancement and also writer identification systems. For example, see Fig. 12.

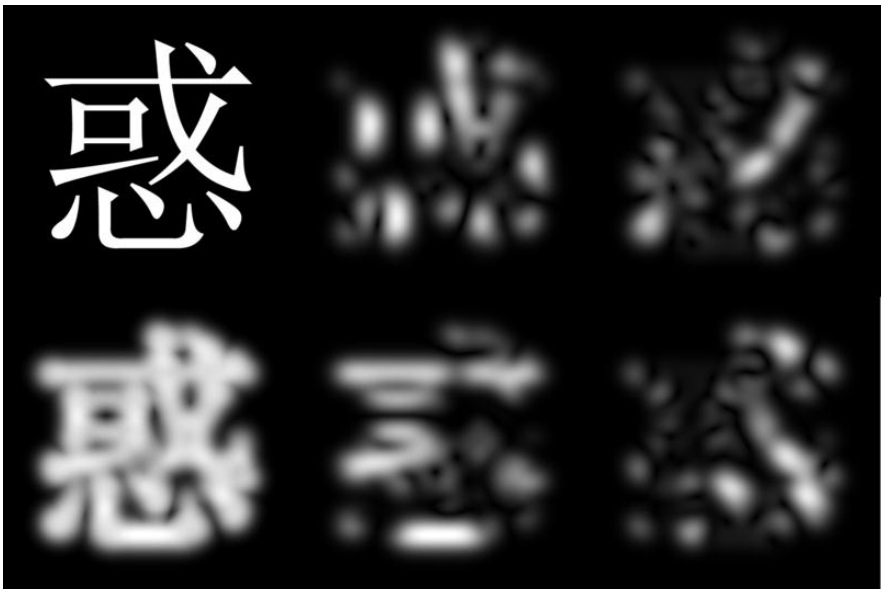

Figure 12. Top left: Original Chinese character. Top middle: Orientation = 0 degree. Top right: Orientation = 45 degree. Bottom middle: Orientation = 90 degree. Bottom right: Orientation = 135 degree. Bottom left: Superposition of all four orientations.

2.3.5 Skeletonisation

The topological skeleton [66] of a shape is a thin version of that shape, with pixels on, in the middle of its boundaries. The skeleton usually highlights the geometrical and topological properties of a shape. These properties include its length, direction, width, topology and connectivity. Along with the information of the distance of its points to the shape boundary, the skeleton of shape can also serve as a representation of that shape since it contains all the information necessary to reconstruct that shape. In this work, skeletonisation is referred to the process of thinning the characters, so only their skeletons are left.

Skeletons have been utilised in various fields, like image analysis, computer vision, and digital image processing, including applications for fingerprint recognition, optical character recognition, binary image compression, and pattern recognition. An example of the skeleton of the word “The” is given in Fig. 13.

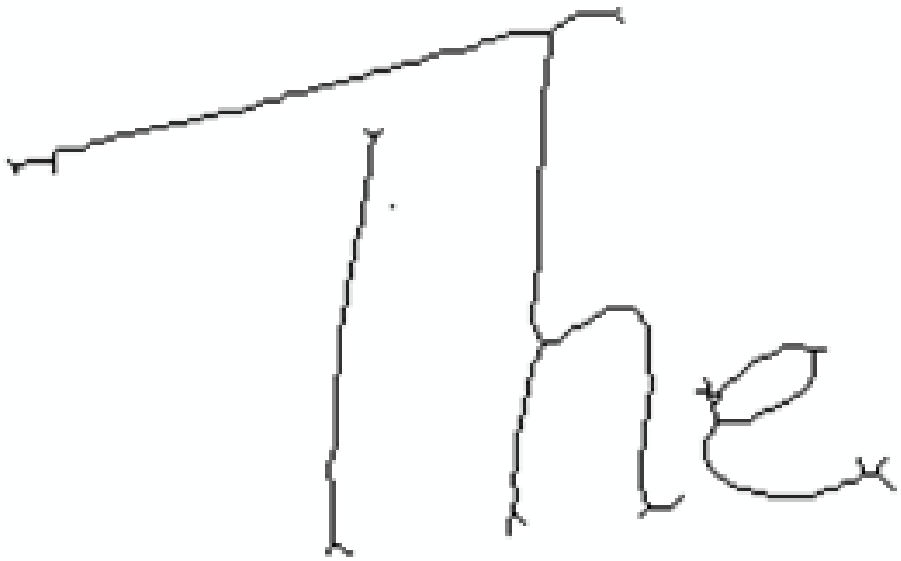

Figure 13. The output of skeletonisation on the image of Fig. 7.

2.3.6 Connected Components

Connected components are groups of pixels that share similar pixel intensity values and are connected [67]. Connected component algorithms work by traversing an image, pixel-by-pixel (from top to bottom and left to right) to identify connected pixel regions, i.e. regions of adjacent pixels that share the same set of intensity values. An algorithm can be set to check for a 4-connectivity connected component or an 8-connectivity connected component Fig. 14. 4-connectivity algorithms check the upper, the bottom, the left, and the right neighbour pixel for the same intensity. 8-connectivity checks the entire neighbourhood of the central pixel for pixels with the same intensity.

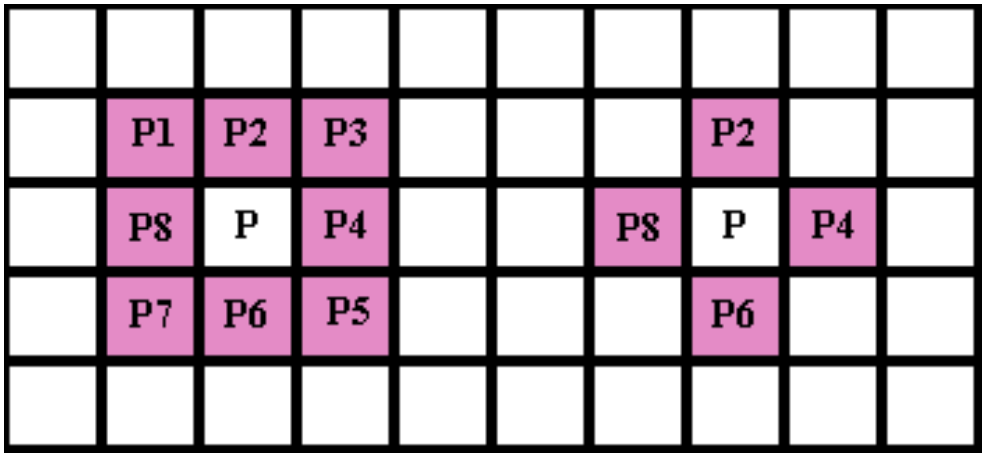

Figure 14. left: 8-connectivity. Also called a Moore neighbourhood. right: 4-connectivity

Only connected pixels with value 1 will be considered to be on the same group for a binary image. While on grayscale images, a range of intensity values is considered. Each group of connected components is labelled, either with an id or with a different colour.

Connected components technique is particularly useful in document image analysis because each character can be categorised as a connected component with a given label.

2.3.7 Contour Tracing

Contour tracing [68] output might look almost the same as edge detection algorithms; however, edge detection algorithms try to find points that are extrema of the image gradient in the direction of the gradient, with the edge pixels, pointing out a significant difference between neighbouring pixels. Contour tracing tries to find the contour, i.e. the boundaries, of an object. Contours need to be closed curves to map precisely the boundaries of any given object, while edge detection does not require closed curve edges. Usually, objects are first identified through a connected component tracing, and then the contour of every object is extracted. A complete contour includes both the exterior contour and the interior contour. Interior contours are harder to detect because they reside in character closed areas. For example, a complete contour of character A is given in Fig 15.

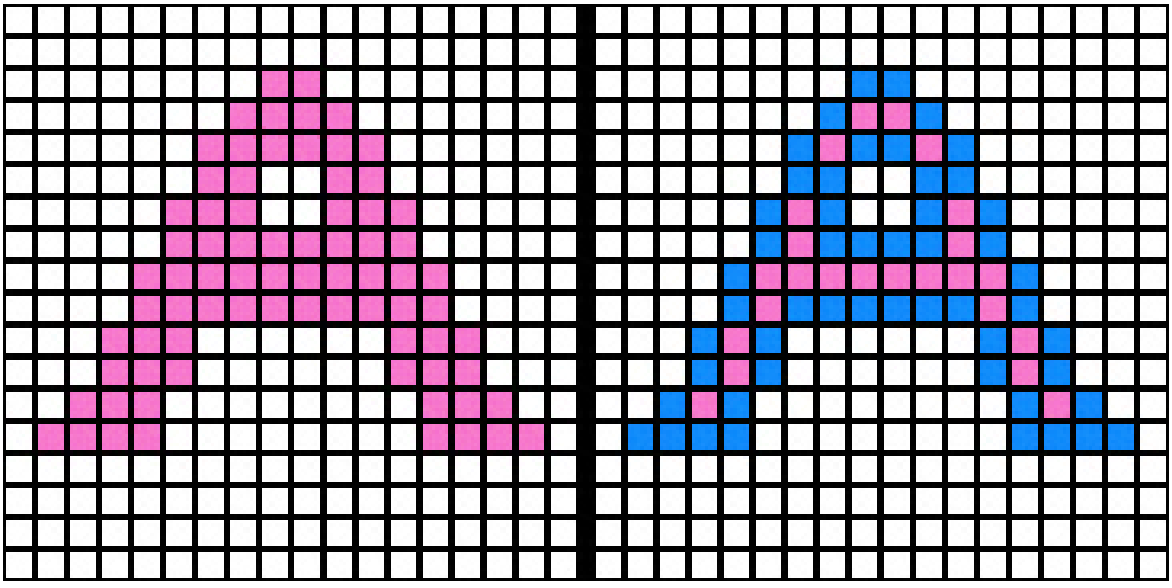

Figure 15. left: The character "A". right: The complete contour of character "A"

2.3.8 Main Body

Main body [30] or core region size is a characteristic that is used quite often in most document image processing systems. By this term, it is considered the central part of the text, excluding ascenders and descenders (Fig.16). Most of the times, it is referred to words.

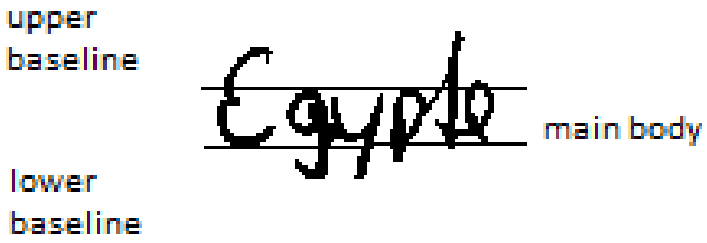

Figure 16. Word main body and baselines.

Main Body is a characteristic used in a large number of systems that use image processing for a variety of tasks in document images. It has been used in systems for OCR [69, 70], segmentation [71, 72], slant removal [73], dewarping [74, 75], word matching [76], indexing [77], normalization [78], word spotting [79], etc. All the above systems utilise the main body information and use it as a threshold or character size information, as it is directly related to the size of the characters, the document image resolution and the text orientation.

The Main Body size can also be utilised in order to get a rough estimation of the character width. Especially in [80], they mention: By mean width of the character, we consider the width of characters such as a, b, c, d, e, f etc., excluding the characters i,l,j,m,w that are either too narrow (i,l), or too wide (m,w). Although the character width differs between characters and writers, a rough estimation of the mean width could be made by accepting that excluding the ascenders and descenders the characters with mean width (as defined above), present width equal to their height.

Considering all the above, we see that Main Body is crucial in document image processing systems. Thus, many techniques have been developed for calculating the Main Body.

2.3.9 Run Length Encoding

Run Length Encoding counts runs of data with the same value that occur in consecutive pixels. It is used mostly in binary images for various tasks, from data compression to skew detection and line segmentation. In binary images, two distinct types of run lengths exist. Black run Lengths, where consecutive pixels that are off (0) are counted and White Run Lengths where consecutive pixels that are on are counted. For example, the pixel sequence in Fig. 17 has a black run-length encoding of 2, 3, 7, 2, 3

0 0 1 0 0 0 1 0 0 0 0 0 0 0 1 0 0 1 0 0 0

Figure 17. An example of a pixel sequence with a black run length with values 2, 3, 7, 2, 3

2.3.10 Fourier Transformation

Fourier Transformation is a valuable image processing tool that is used to decompose an image into its sine and cosine components [81]. The output of a Fourier transformation is a complex number valued output image that represents the input image in the frequency domain. The input image is considered the spatial domain equivalent, and by spatial, it is meant the normal image space. In the Frequency domain image, each point represents a particular frequency contained in the spatial domain image.

The complex number valued output image produced by the Fourier Transform can be displayed with two images, the magnitude image and the phase image. The magnitude image contains most of the information of the geometric structure of the input image, and thus, in image processing, only the magnitude of the Fourier Transform is displayed. However, in some tasks like re-transforming the Fourier image into the correct spatial domain image after some processing in the frequency domain, both the magnitude and phase of the Fourier image are required.

Furthermore, the Fourier domain image consists of a much higher range than the image in the spatial domain. Thus, to be sufficiently accurate, its values are usually calculated and stored in float values. The Fourier Transform can be found in various applications, such as image filtering, image analysis, image reconstruction and image compression. An example of the magnitudes of specific handwriting letters can be seen in Fig 18.

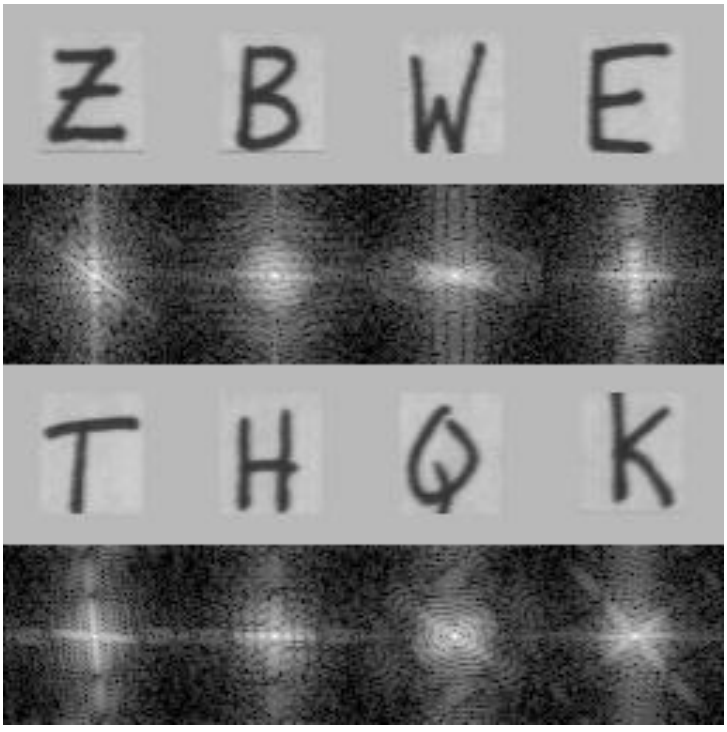

Figure 18. Handwritten characters along with their Magnitude in the Fourier domain

2.3.11 Text Localization

Text localisation is an old computer vision problem, which started to be studied in the ’90s [82]. Nowadays, efficient solutions to this problem are more useful than ever in fields like robotics, smart cars, smartphones, etc. This is the reason why many techniques have been proposed using connected components [83], edge detection [84], sliding windows [85], hybrid techniques [86], as well as other techniques [87]. Moreover, in the last years, four competitions have been organised on robust reading [88, 89, 90, 91], which has motivated active research in this area.

Chapter 3

3. Main Body and Text Localization contributions

In order to validate or dismiss two of our assumptions, and more specifically, the third and fourth assumption, some preprocessing techniques were developed to facilitate us with this validation.

Firstly, for the third assumption, regarding the effect of Main Body variance in the accuracy of writer identification, two techniques appropriate to detect the Main Body size were developed [30]. From the two Main Body size estimation techniques, only one was used for writer identification due to its speed and the lower estimation error. Directional Features are susceptible to the Main Body size variance since the angle measured will be narrower in smaller characters and wider in bigger characters. Our approach to validate or dismiss this assumption consists of measuring the global Main Body size of a document and local Main Body size on the word level and use their ratio as the weight for our feature extraction technique. To the best of our knowledge, there is no other paper in the literature specific in Main Body size detection other than [30].

For the fourth assumption, regarding the effect of noise produced by the writer in the accuracy of writer identification, a variation of a technique used for text localization [92] that is appropriate to localise only pure text using some rules and dismiss all the noise produced by the writer. This kind of noise usually consists of the writers attempt to erase with ink what he has written by mistake or smaller ink stains that could be regarded as salt and pepper noise. For example, see Fig. 19.

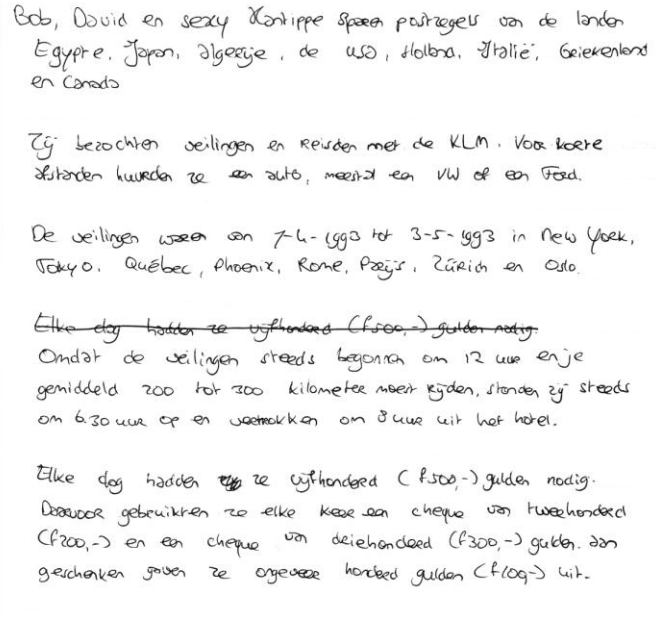

Figure 19. Writers attempt to erase with ink what he has written by mistake

3.1Main Body Size Estimation

3.1.1 First Technique

Our first technique, shown briefly in Fig.20 and analysed in this section, estimates the average main body of words in a scanned document. Although it has some similarities with [73], it is not that complex; it does not require line segmentation nor image binarisation. The technique is applied to grey level images, although the experimental results prove that if the image is binarised and cleaned from extra noise and then converted to grey level, the results are improved.

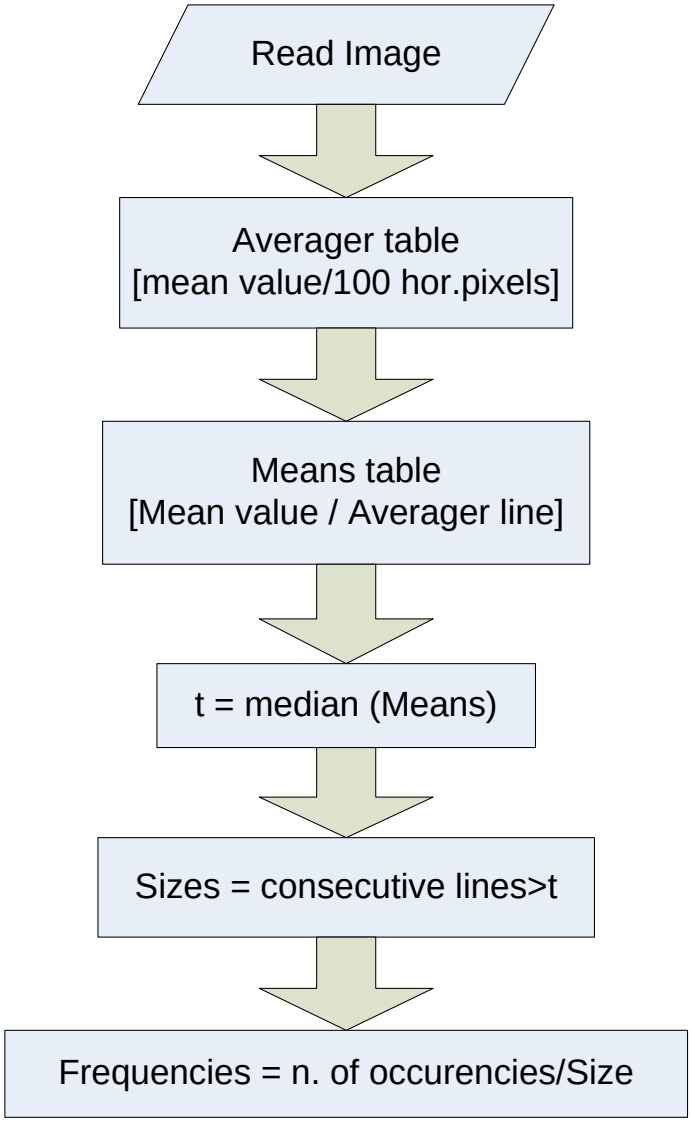

Figure 20. The proposed Main Body Size Extraction methodology.

First, the average pixel value is calculated for every N pixels that exist in each pixel line of the image. N can be any value. It is only essential for the skew angle of the page that it can handle.

The smaller the N, the biggest the skew angle it handles. However, since this work does not emphasise that, for the results presented here, N=100 was chosen. The results are saved in the Averager table with size Hx[W/100], where H is the height and W the width of the image.

Next, the table Means is created of size Hx1, where its elements are the average values of the corresponding lines of the Averager matrix. Then the threshold t is set as the median value of matrix Means. By this threshold, we set to zero the values of Means that are smaller than t, while we count the consecutive lines with a value bigger than t. The set of the different amount of the consecutive lines is the Sizes of the various main bodies in the image. Then the occurrences for each size are also counted and saved in Frequencies. As main body size, the maximum in Frequencies is considered.

Figure 21. Schematic presentation of the technique through example

In Fig. 21 the technique is presented through an example. This technique does not require binarisation. Moreover, it can give more information if different main body sizes are present on the same page.

3.1.2 Second Technique

This technique, developed by Vassilios Veras, a co-author on [30], estimates the baselines of the text in a document page. Initially, the document is binarised. Then, the Connected Components (CCs) of the document are detected for the 8-pixel neighbourhood. All CCs bigger than 30000 pixels and smaller than 10 pixels are removed, that is a very big area e.g. scan noise or figures and tiny noisy areas or accents, respectively.

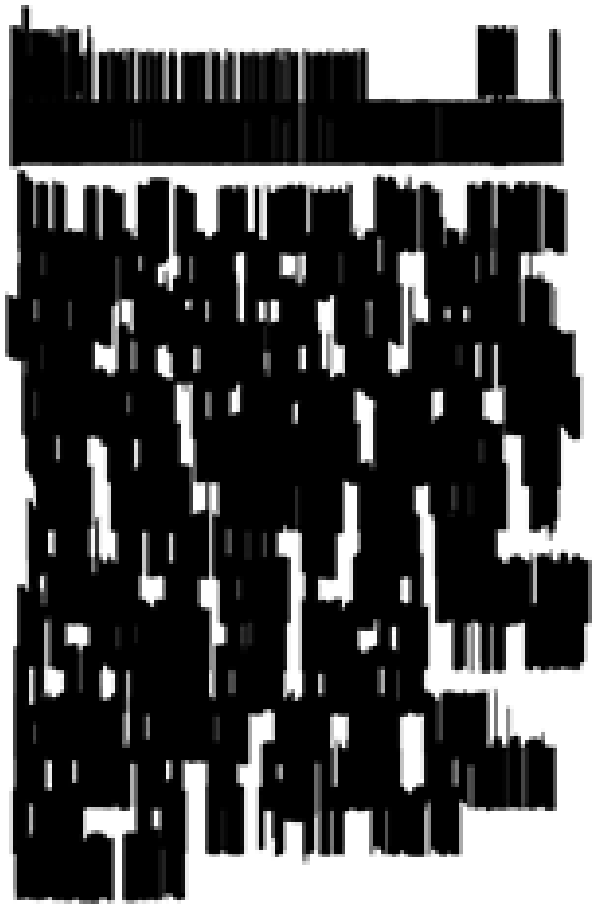

Figure 22. Vertical dilate.

Then vertical dilate is applied with an aim to identify the horizontal borders of the text area (left-right) and if the text consists of text columns, this is necessary since, in the case of the text columns, each column is treated separately. After the vertical dilate the columns of text form a big connected

area (Fig.22). Consequently, CCs are again detected and now only those bigger than 10000 pixels are kept. A vertical histogram is taken, and those pixel columns with black pixels more than 75% of the average are marked as text and the others as background (Fig.23).

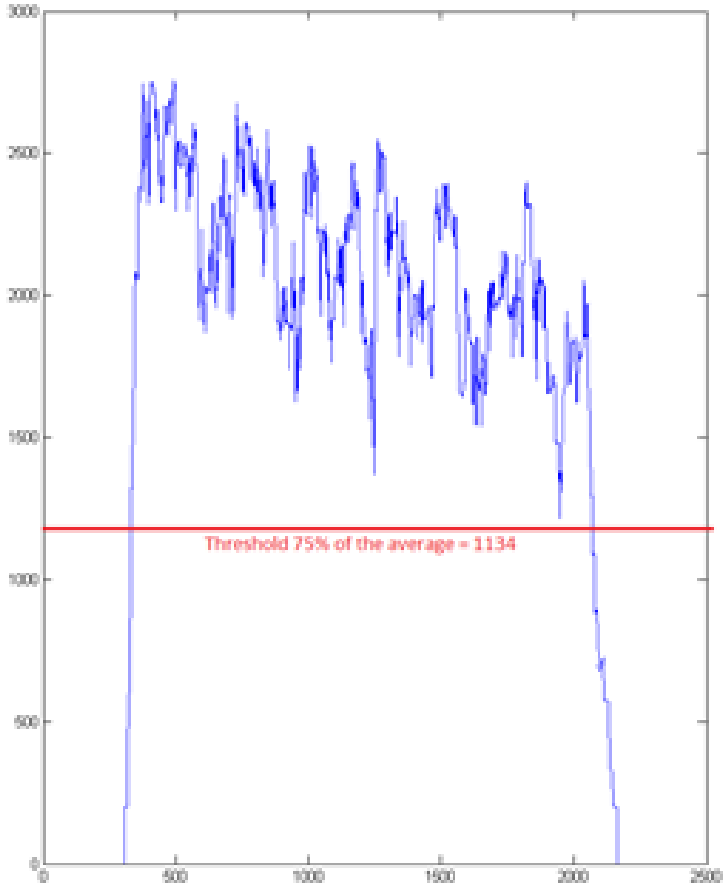

Figure 23. Vertical text localisation.

Then the document is scanned from left to right, and the total number of text columns is identified. For each of the text columns, a similar procedure is followed this time with a horizontal dilate (Fig.25). The text lines are detected with their respective start and end indexes in the document. To detect the main body of the text the pixel row must contain 170% of the average pixel rows (Fig.26), this ensures that the beginning and the end of the main body will be detected, without including the ascenders and the descenders. Finally, the average baselines are calculated and returned as showed on the original document (Fig. 27). The technique is presented in Fig.24.

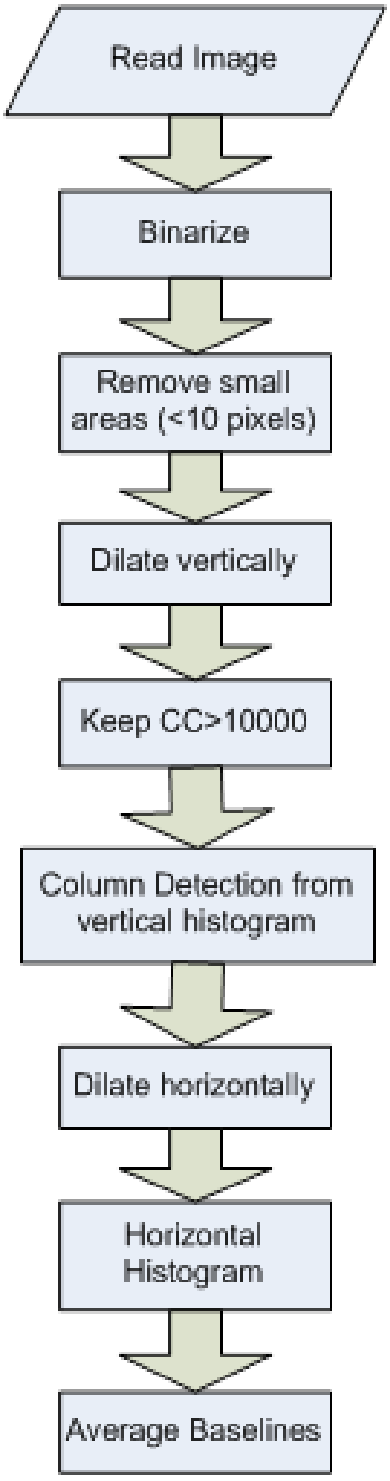

Figure 24. The second technique

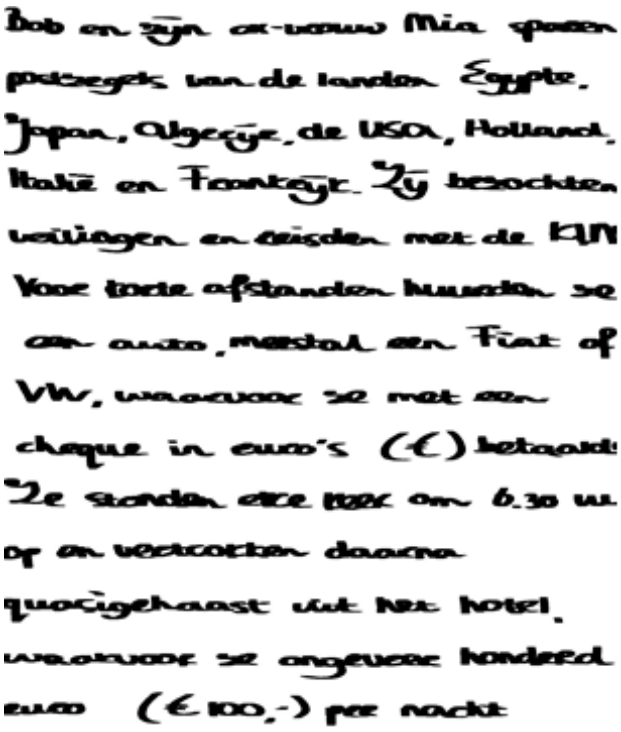

Figure 25. Horizontal dilate.

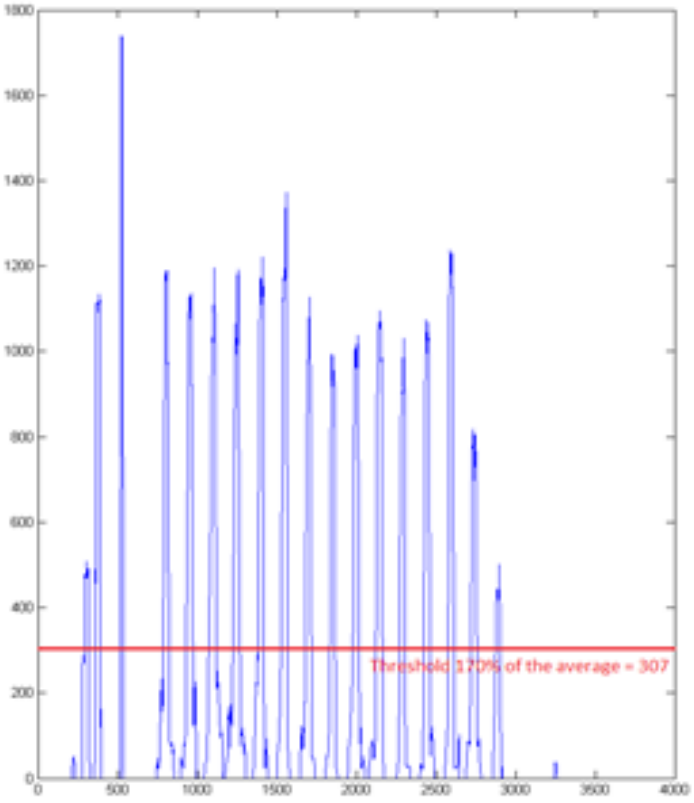

Figure 26. Horizontal text localisation.

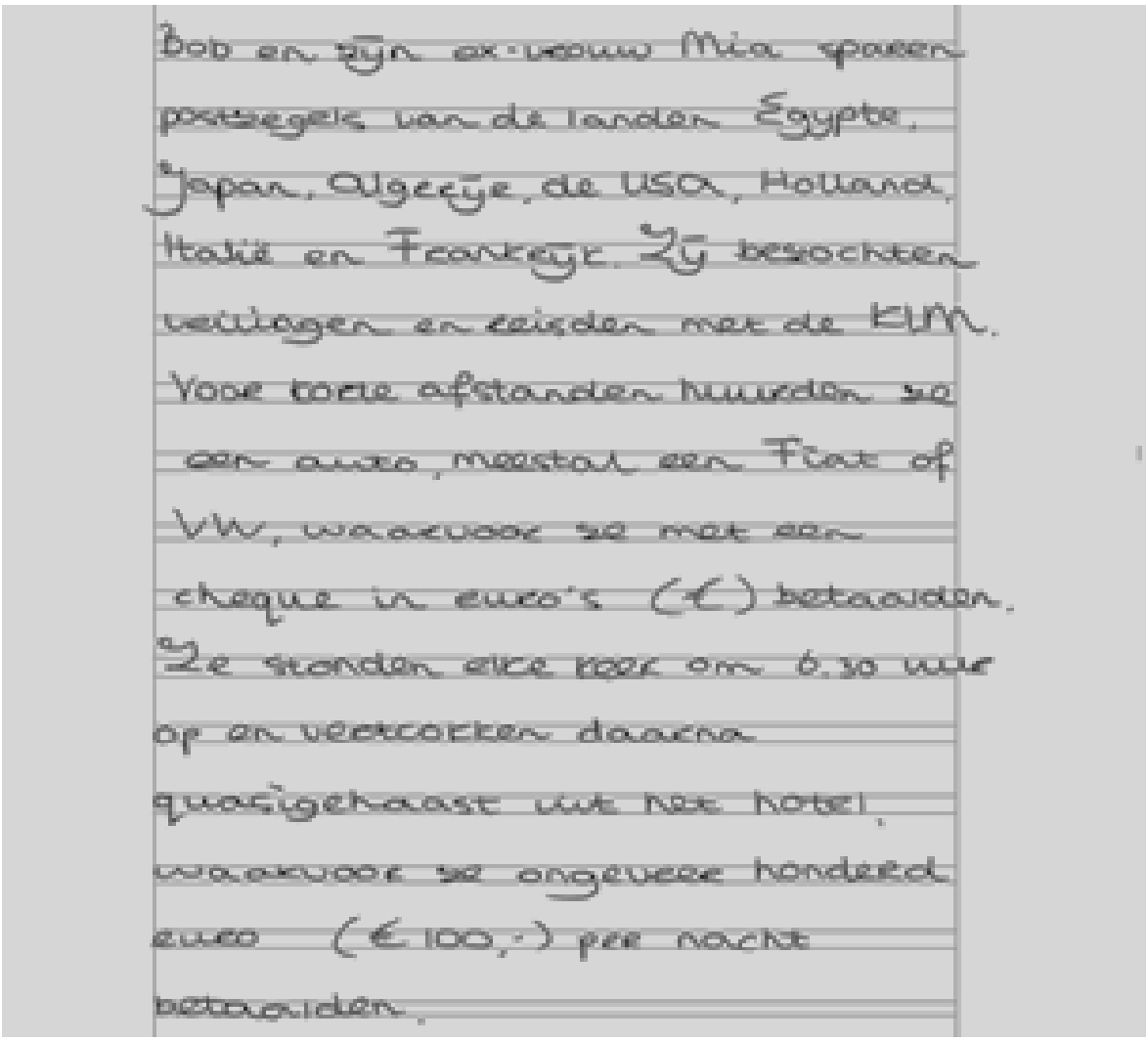

Figure 27. Result with upper and lower baselines visible

3.1.3 Experimental Results

Evaluating a technique that estimates the text main body size is not easy, primarily when we refer to the handwritten text. Here, the TrigraphSlant data set [93] that contains images of handwriting produced under conditions of natural and forced slant, were used. It includes 190 images from 47 persons. We used 30 images of natural writing by different writers.

In order to create ground truth data, the height of 10 'o' of each image was measured, and the mean value was considered. It took us by surprise that even on the same document image, written by the same person, differences of more than 10 pixels were found.

In table 1, the mean estimated main body size for five writers (D00X) is shown, while you can see the results for 4 document images of the same writer (D00X-1, D00X-2, D00X-3, D00X-4). It is obvious how the size changes, even for the same writer. In our experiments, only the first document image (D00X-1) of each writer for X between 1 and 30 was used.

Since, as it was explained, it is difficult to have exact results, in table 2, the average error deviation between the estimated values and the ones detected by the two techniques is given. Moreover, in order to give more objective results, in the same table, it is given the average error deviation between the real values and the ones detected by the two techniques over a collection of 10 printed images that includes font sizes between 8 and 24 pts.

Table 1. Examples of Main Body size estimation

Document Image Code	***Estimated Mean Main Body Size (pixels)***
D001-1-An	34,8
D001-2-Bn	35,2
D001-3-Bl	31,6
D001-4-Br	31,6
D002-1-An	33,6
D002-2-Bn	27,4
D002-3-Bl	37,2
D002-4-Br	27,2
D003-1-An	34,2
D003-2-Bn	33,4
D003-3-Br	30,8
D003-4-Bl	33,4
D004-1-An	29,2
D004-2-Bn	29,2
D004-3-Br	32,8
D004-4-Bl	34,2
D005-1-An	34,6
D005-2-Bn	34,2
D005-3-Br	28
D005-4-Bl	33,4

Table 2. Experimental Results

Technique	*Average error deviation (pixels) on Trigraph*	*Average error deviation (pixels) on printed DB*
first	2.17	0.67
second	4.96	1.05

3.2 Text Localization

3.2.1 System Overview

The proposed method, developed by Ergina Kavalieratou and Pilar Gomez-Gil, co-authors on [92] takes advantage of two facts: a) a text should contrast with its background in order to be readable; b) a text follows some regularity in any language. Figure 28 shows the main steps of this method. First, an RGB image is transformed to a grayscale image using the formula [94]:

Equation 1. Grayscale image transformation from RGB

$$gray = 0.2989 * R + 0.5870 * G + 0.1140 * B$$

Where R, G, B correspond to the colour of the pixel, respectively. This image is binarised for using various thresholds, which is defined as:

Equation 2. Thresholds for image Binarization

$$threshold = minimumI + k * STEP$$

Where minimumI corresponds to the minimum intensity of the grayscale image and STEP is a small value. In the experimental results reported here, STEP values go from 1 to 7. The k parameter considers values from 1 up to (maximumI - minimumI)/STEP, in order to cover with various thresholds all the range between the minimum (minimumI) and the maximum (maximumI) of the grayscale image. A binary image and its reversed one are built, using as threshold all the multiples of a specific STEP. The reverse image is also used since it cannot be known in advance if the foreground is lighter or darker than the background.

Using all possible black and white images, all possible contrasts should be included for a small step. After this, each image is examined in detail for the existence of several constraints that the parts of images corresponding to texts are expected to accomplish:

1. Similar colours between the text parts or dissimilarity less than 10% are expected.
2. Within a text region, the dissimilarity in colour should remain less than 10%.
3. Areas with either size less than 5 pixels are not considered.

4. The parts of the text, usually characters, are expected to have a similar width, with a maximum deviation of 10%. A difference up to 50% is allowed in height, in order to include words with uppercase and lowercase letters.
5. Neighbour text parts are expected to have similar areas with a deviation of maximum 30%, due to the difference between characters and uppercase/lowercase letters.
6. The parts of the same text square are expected to have between them a horizontal distance of maximum 3xMB and a vertical distance of maximum one MB, where MB is a rough approach of the mean character size.

As it is mentioned in [80]: By mean width of character, we consider the width of characters such as a, b, c, d etc, excluding the characters i, l, j, m, w that are either too narrow (i, j, l), or too wide (m, w). ... Although the character width differs between characters and writers, a rough estimation of the mean width could be made by accepting that excluding the ascenders and descenders the characters with mean width (as defined above), present width equal to their height. In our system, MB is estimated as the CCs height; rules 4, 5 and 6 derive from the above definition.

CCs of every binary image is extracted, and several properties are calculated:

- The centroids of the CCs.
- The minimum, maximum and mean intensity of the corresponding area in the grayscale image.
- The area of each CC.
- The bounding boxes of each CC. along with the coordinates of the upper left corner and their width and height.
- The main body, as the height of the CC.

Then, for each pair of CCs and using these properties, the rules described above are applied, respectively:

1. The mean intensity of the two CCs is expected not to differ more than 10%.
2. The minimum and maximum intensity of each CC is expected not to differ more than 10%.
3. The CCs with an area less than 10 pixels or Bounding Box width or height less than 5 pixels are eliminated.
4. Both bounding box heights and widths are required not to differ more than 10%.
5. The areas of the CCs are expected not to differ more than 30%
6. The x coordinates of CCs centroids are expected to be situated within 3MB and the y coordinates within one MB.

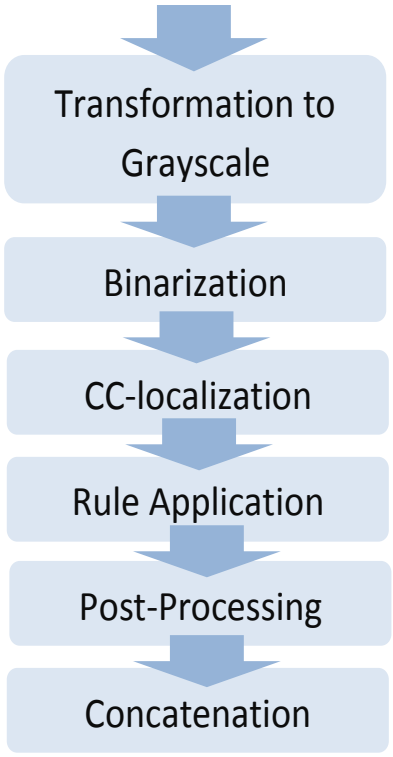

Figure 28. The tasks of the proposed system

A pair of CCs that pass successfully the above constraints is considered to be part of the same group. The procedure is repeated for the reverse image. As a result, from each image, several areas are extracted as possible text areas. Considering that natural scenes can include all tones of grayscale from 0 to 255 (not always), some dozens of binary images are considered, and the extracted areas could be up to hundreds. However, since they are black and white images, the processing is fast.

Thus, the extracted areas are unified if the following rule holds:

IF the mean value of the y-coordinates of an area is included in the y-coordinates of another area AND (their areas are either overlapped, OR they are not located more than one MB apart, horizontally.)

At this point, our system presented a recall >70%, which is very high compared to the results of other systems of the competition but very low precision. This is the reason that a post-processing stage was included in the system. In this post-processing stage, every single part of the image is analyzed in order to confirm that it includes text. Thus, it includes the following procedures:

- First, the image is cleaned in all sides, above, below, left and right by pixel lines that include entirely white or entirely black pixels.
- Then, if there are horizontal pixels lines that include only white or only black pixels, the image is split into horizontal text lines, limited by these pixel lines.

- Finally, it is checked if there are entire columns with only white or black pixels that it is expected to separate characters.

Once the post-processing is done, a concatenation procedure is applied, in order to unify the overlapped parts of the image.

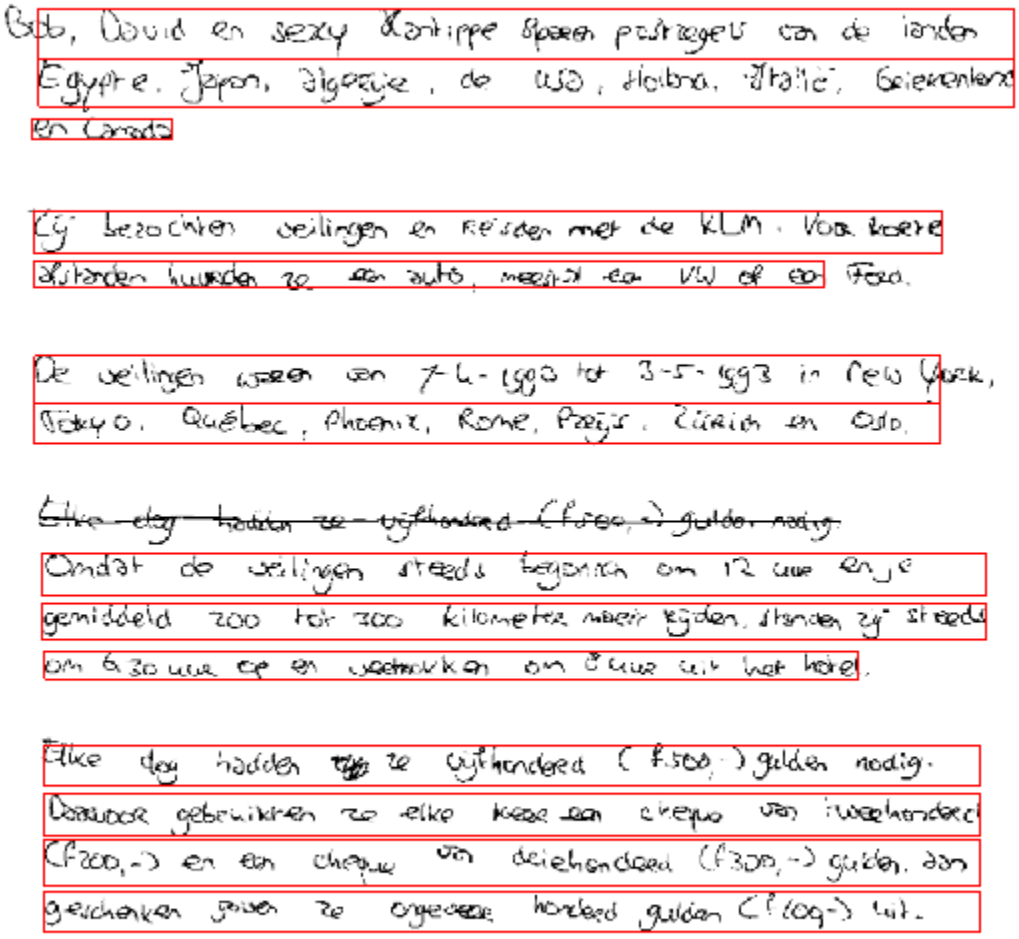

Figure 29. Text localisation result example of the document from Figure 19.

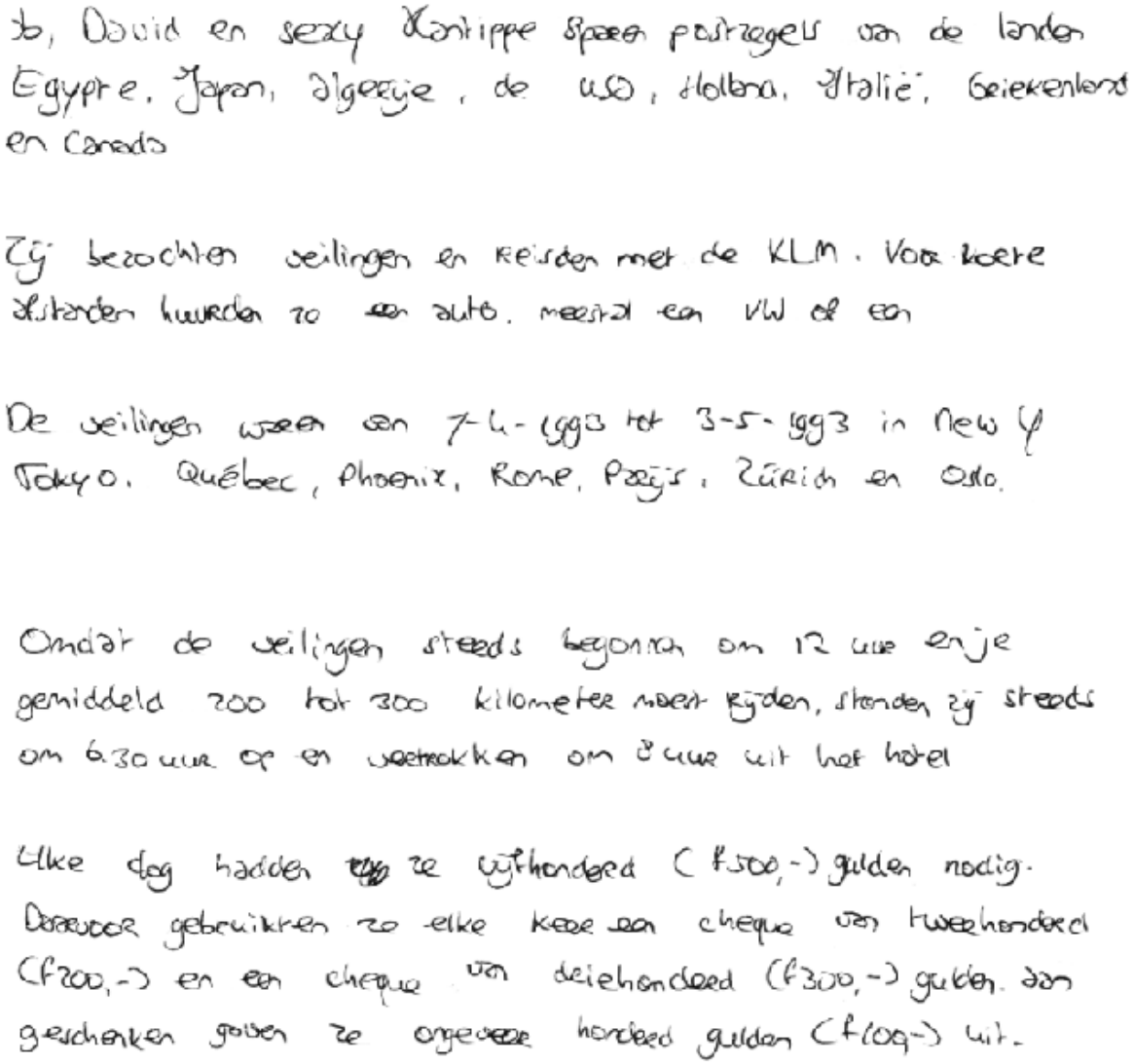

Figure 30. Final Image after text localisation, on the document from Figure 19.

3.2.2 Experimental Results

For the evaluation of the proposed procedure, the dataset of ICDAR 2011 Robust Reading Competition Challenge 2: Reading Text in Scene Images [95] was used. The final dataset consisted of 485 images containing text in a variety of colours and fonts on many different backgrounds and orientations. A comparison of our results with other techniques is shown in Table 3. Furthermore experimental results for the writer identification task, after filtering the document images with this

technique and using the Skeleton Hinge distribution for feature extraction can be found in chapter 5.

Table 3. Comparative results with, the dataset of ICDAR 2011 Robust Reading Competition Challenge 2: Reading Text in Scene Images [96].

Method	**Recall**	**Precision**	**Harmonic Mean**
technic 1	62.47	82.98	71.28
technic 2	58.09	67.22	62.32
technic 3	57.68	66.97	61.98
technic 4	52.54	68.93	59.63
technic 5	53.52	63.52	58.09
technic 6	50.07	62.97	55.78
technic 7	44.57	59.67	51.03
technic 8	38.32	35.01	36.59
technic 9	25.96	50.05	34.19
Proposed Technique	**77.08**	**57.15**	**65.63**

Chapter 4

4. Writer Identification

State of the art writer identification systems uses various techniques that use different features types and classifier approaches to identify the writer of handwritten text. Features can have many types, statistical features where distribution is calculated, structural features where specific rules that relate with the text structure are applied, model-based features where the characters are treated like allographs or graphemes, textural features where the text is treated as texture and many others. Furthermore, features can be extracted from different levels of the text, like the macro level that includes features from the entire document, paragraphs, lines or words and the micro-level that includes features from characters, parts of characters (graphemes) or pixels. On the other hand, classification approaches can be categorised into five types [96]: minimum distance classifiers, statistical classifiers, neural networks, fuzzy classifiers and syntactic classifiers.

In the scope of this work, a focus is given on statistical-textural directional features on the micro level that are extracted using a Probability Distribution Function (PDF) and minimum distance classifiers. Further experiments were performed using a combination of statistical-textural features with model-based features. Finally, statistical classifiers and neural networks were also considered.

On the statistical approaches, the focus will be given on edge direction features [55], and their advancements, edge hinge distribution and edge hinge combinations [26]. Edge hinge distribution is reported to outperform all other statistical features while edge hinge combinations, is an improvement of the previous method. An attempt is made [97], to improve even further the edge hinge combinations methodology using image skeleton, thus referring to this methodology as skeleton hinge distribution. Further improvements are also attempted on the skeleton hinge distribution on a weighted variation using the main body size at the pixel level. Furthermore, an attempt was made to utilise the pixel intensity information on a quantised version of the skeleton hinge approach. A novel approach on directional features is presented using Directional Stroke Run Length Hinge Distribution. Finally, a combination of the Edge Hinge Combinations Distribution with the Skeleton Hinge Distribution is presented.

While directional features distributions have good results, they are directly related to the writer's slant. The slant is a characteristic that can be easily forged. To secure this method, and further improve the results a combination of skeleton hinge distribution with a model based one is presented. The model-based technique that was used in this thesis involves the use of pre-defined models of small strokes of handwriting called graphemes.

4.1 Statistical Features

4.1.1 Edge-direction distribution

This feature extraction starts with edge detection. Edge detection generates a binary image in which only the edge pixels are kept. Next, each edge pixel is considered the centre of a square neighbourhood. All the pixels are checked, using logical AND operators, to all directions, emerged

from the central pixel and end on the periphery of the neighbourhood, looking for the presence of another edge fragment (i.e. connected sequences of pixels). In Fig. 31, an example with a 4-pixel length edge fragment quantised in 12 directions is presented. A histogram is created, using the count of all the verified instances, and then it is normalised to a probability distribution p(φ). This distribution gives the possibility of finding in the image an edge-based fragment oriented at the angle φ to the horizontal. Moreover, the most dominant direction in p(φ) corresponds to the slant of the handwritten text.

Some important practical details that relate to the implementation of edge-direction distribution [55] should be mentioned. In order to avoid repetition, the algorithm only checks the upper two quadrants in the neighbourhood. This is done because without online information, it is hard to determine which way the writer "travelled" along the found oriented edge fragment. In the experiments contacted in [55], they only considered 3,4 and 5 pixel-long fragments that were quantised in n= 8,12 and 16 directions, respectively. It is also worth mentioning that the edge detection method used does not generate 1-pixel wide edges, but instead edges that have a wide of 1 to 3 pixels. This practical detail introduced smoothing into the histogram computation, something that they found advantageous in the experiments contacted. For more details about algorithm options and results, see [55].

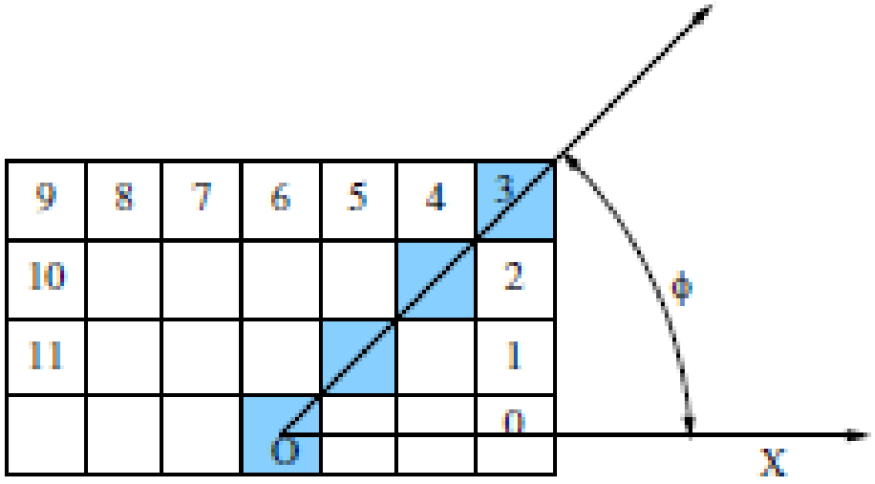

Figure 31. Extraction of edge-direction distribution.

4.1.2 Edge-Hinge distribution

As reported by Bulacu et al. [55] the edge hinge distribution is a statistical feature, which outperforms all the other statistical approaches. The central idea in the edge hinge distribution is

to consider not one but two edge fragments in the neighbourhood, emerging from the central pixel, and subsequently compute the joint probability distribution of the orientations of the two fragments. This feature concerns the direction changes of a writing stroke in handwritten text. The edge-hinge distribution is extracted using a window that scans a binary handwriting image that contains only the edge information. When the central pixel of the window is "on", the two edge fragments emerging from this central pixel are considered only when $\varphi 1 < \varphi 2$. In Fig 32, an example with 4-pixel length edge fragment quantised in 24 directions. The directions are measured and stored in pairs. A joint probability distribution $p(\phi 1, \phi 2)$ is obtained over a large sample of pairs.

Furthermore, some practical details that relate to the implementation of the algorithm used [55] for this feature is worth mentioning. In this implementation, the edge detection algorithm used does not produce 1-pixel wide edges, but instead, it produces 1-3-pixel wide edges. In the edge-direction distribution, that only one edge fragment is checked. This does not consist of a limitation. In our case, two edge fragments must be checked. This consist of a limitation; thus, an extra constraint is implemented. The ends of the edge fragments are required to be separated by at least one "non-edge" pixel. In the experiments conducted for this feature in [55], like the edge-direction distribution, only 3,4 and 5 pixel-long fragments are considered that are quantised in $2n = 16,24$ and 32 directions, respectively. Furthermore, two more constraints are implemented in the algorithm that worth mentioning. The first is that the $\varphi 1$ angle must be lower than the $\varphi 2$ angle. The second one is that the cases when the ending pixels have a common side are eliminated. For more details about algorithm options and results, see [55]

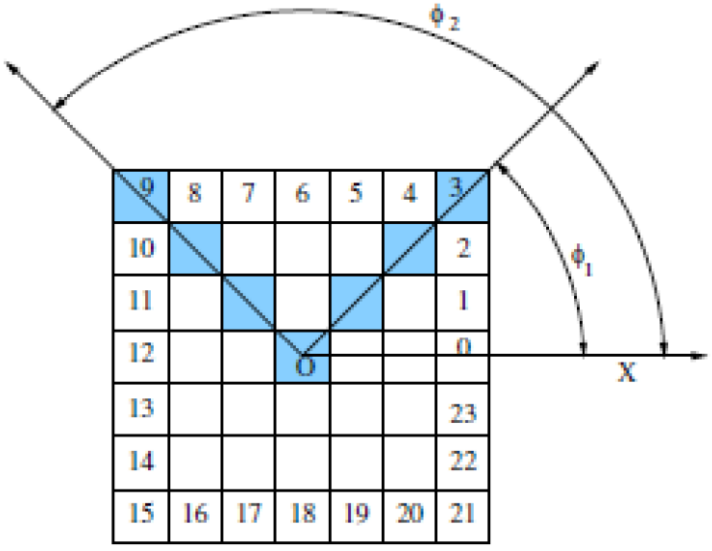

Figure 32. Edge Hinge Distribution Extraction

4.1.3 Edge-Hinge combinations

The edge-hinge combinations, proposed by Van der Maaten et al. [26], improved the edge hinge distribution by considering multiple pixel length edge fragments (i.e. window sizes) instead of just one. Experimenting with combinations of edge hinge distributions and using various fragment lengths, they improved the results of writer identification by up to 12% compared with the edge-hinge distribution. The algorithm of this implementation is available at [98].

4.1.4 Skeleton-Hinge distribution

The main problem with the current implementations is that the edges are usually close to each other, filling the feature matrix with unnecessary data. To overcome this problem, a similar technique as Edge-Hinge Combinations was used but with the skeleton information of the image instead of the edge information. Henceforth, this technique will be referred to as skeleton hinge distribution.

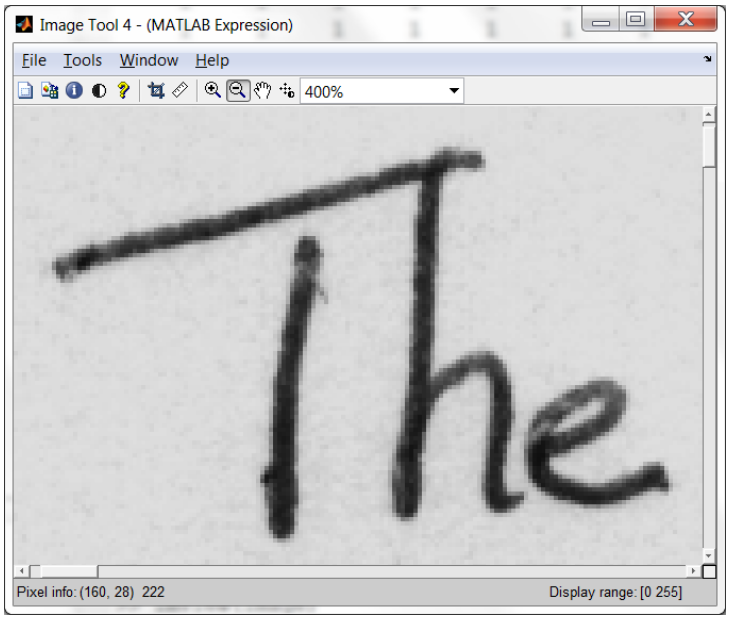

Figure 33. Handwritten digitised text

Usually, when something is written on paper (Fig.33), its thickness is considered to be a single line. When the image is digitised, the same trace of ink is translated into several pixel lines. By considering the edge hinge distribution, on an edge image (Fig. 34), much unnecessary information, like the bottom or the side curves of the letters, is included in the feature vector.

Furthermore, differences in line thickness from a variety of different pens may produce significant variations in the extracted features in both edge hinge distribution and edge hinge combinations.

The main suggestion in this work is that all stroke widths, i.e. line thickness, should be the same size. This is achieved by skeletonising the characters to a single-pixel width line.

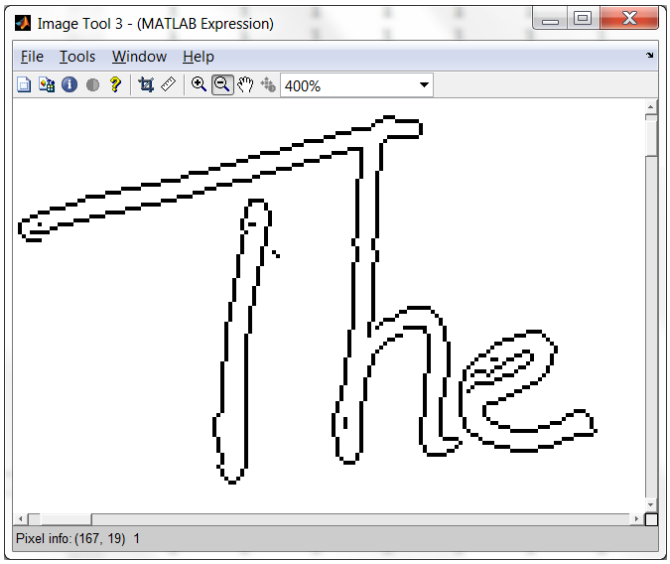

Figure 34. Edge image of handwritten text

On the skeleton hinge distribution, only the skeleton of the letters is considered (Fig.35), a simple structure that considers the basic required information to match the features to already known ones.

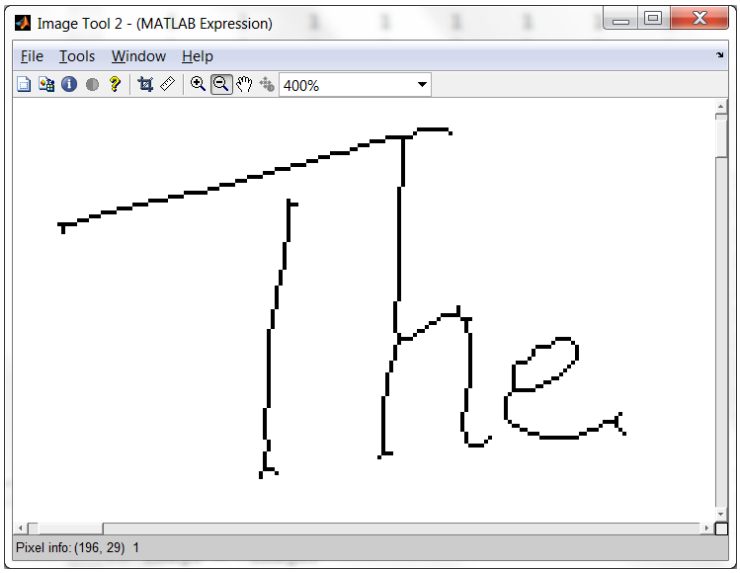

Figure 35. Skeleton image of handwritten text

The Skeleton Hinge distribution [97] belongs to a family of similar techniques like the edge hinge distribution and edge hinge combinations. The main idea is to locate two hinge line fragments emerging from a central pixel on a sliding window (Fig. 36), and store their directions in pairs.

While on edge hinge distribution and edge hinge combinations, the edge information is used to locate hinge fragments, on skeleton hinge distribution, the skeleton information is used.

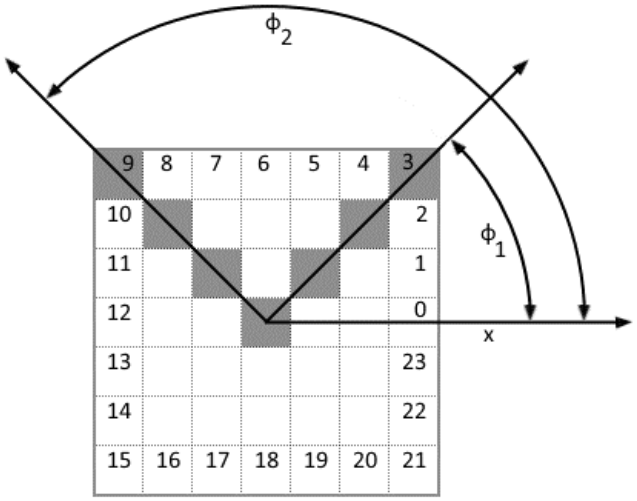

Figure 36. 4 pixels long Hinge line fragments, emerging from a central pixel, on a 7x7 window.

Skeleton Hinge distribution starts with the image skeleton extraction using a generic skeletonisation approach [99]. Then a sliding window technique that uses several window sizes, each quantised in a different number of directions, checks for skeleton line fragments, which emerge from the central window pixel. Finally, their directions are measured and stored in pairs. Only skeleton line fragments with φ1<φ2 are counted and stored in pairs in a histogram. A joint probability distribution p(ϕ1, ϕ2) is obtained over a large sample of pairs. The probability distributions, acquired by the various sliding window sizes, are combined and considered for matching. For an instance of the Skeleton, Hinge Distribution extraction, see Fig. 37.

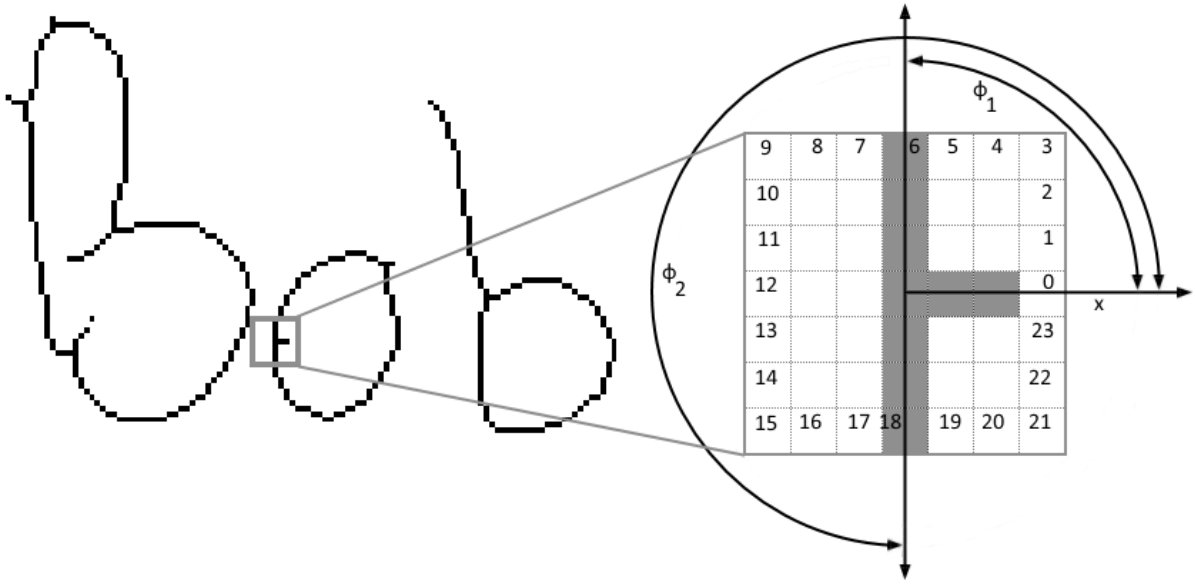

Figure 37. An instance of Skeleton Hinge distribution extraction with 4 pixels-long edge fragments on the part of the word "Bob".

The main ideas of edge hinge distribution and edge hinge combinations are present in the proposed technique. On the other hand, by applying this methodology to a skeleton image, a significant improvement on the results of the writer identification task is observed (See chapter 5 for results).

It is important to mention that the resulting feature matrix includes more compact information than the Edge Hinge Distribution feature matrix, and it is easier to compare two resulting matrices of test and train samples. Please check a successful application of the proposed system in figures 38,39,40, where some text samples are provided over their results. On the upper part of the figure, fragmented samples of the text are provided. On the left, a fragment of the text used as a training sample, and on the right, a fragment of the text, used as a test sample. Both samples in each picture are from the same writer. Next, the surface of the Skeleton Hinge Distribution is presented. The left one corresponds to the training sample, while the right one to the test sample. On the lower part of the figure, the edge hinge combinations surface is presented. Again, the left one corresponds to the training sample and the right to the test sample.

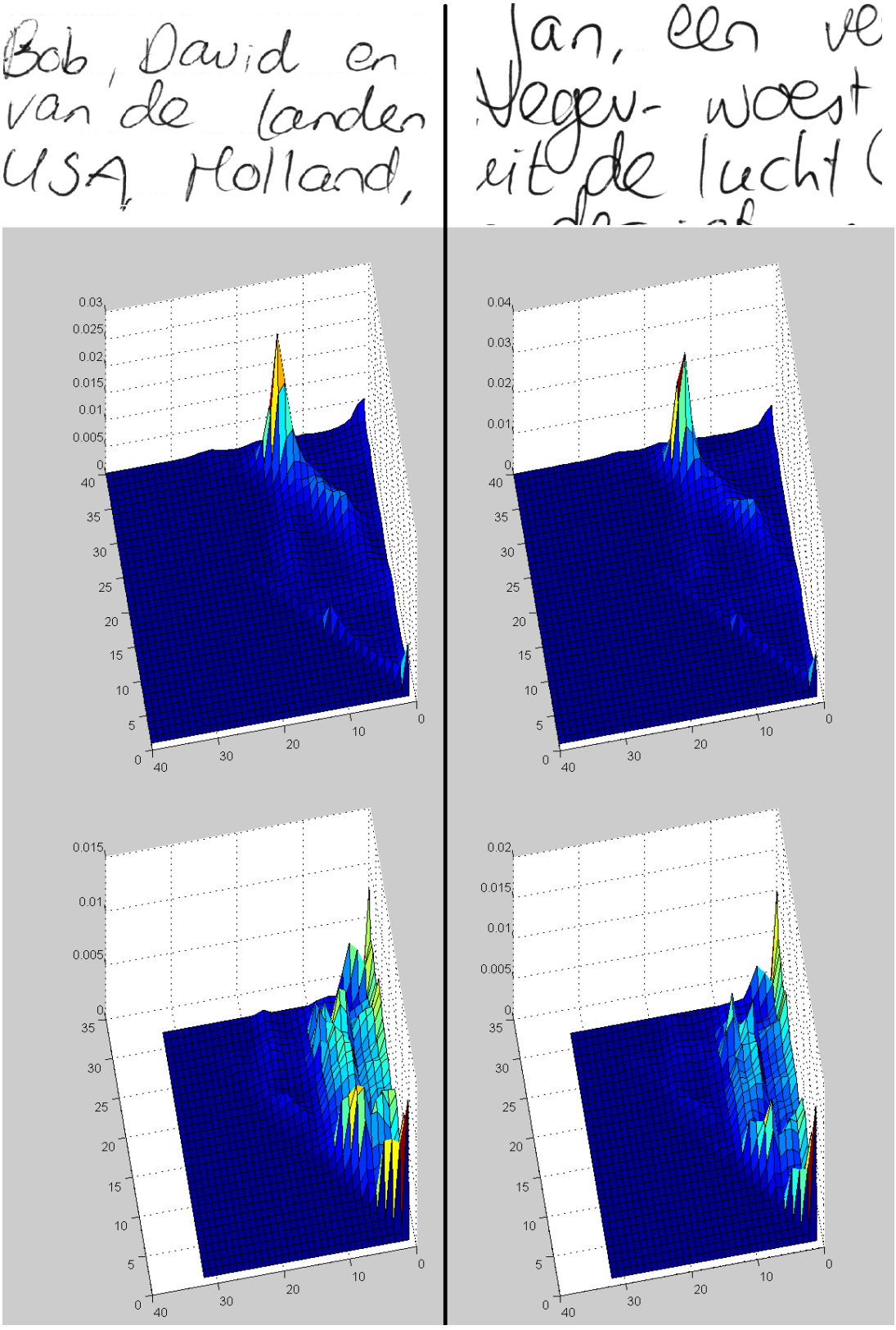

Figure 38. Text samples from the same writer along with skeleton hinge distribution feature surface (middle) and edge hinge combinations feature surface (bottom)

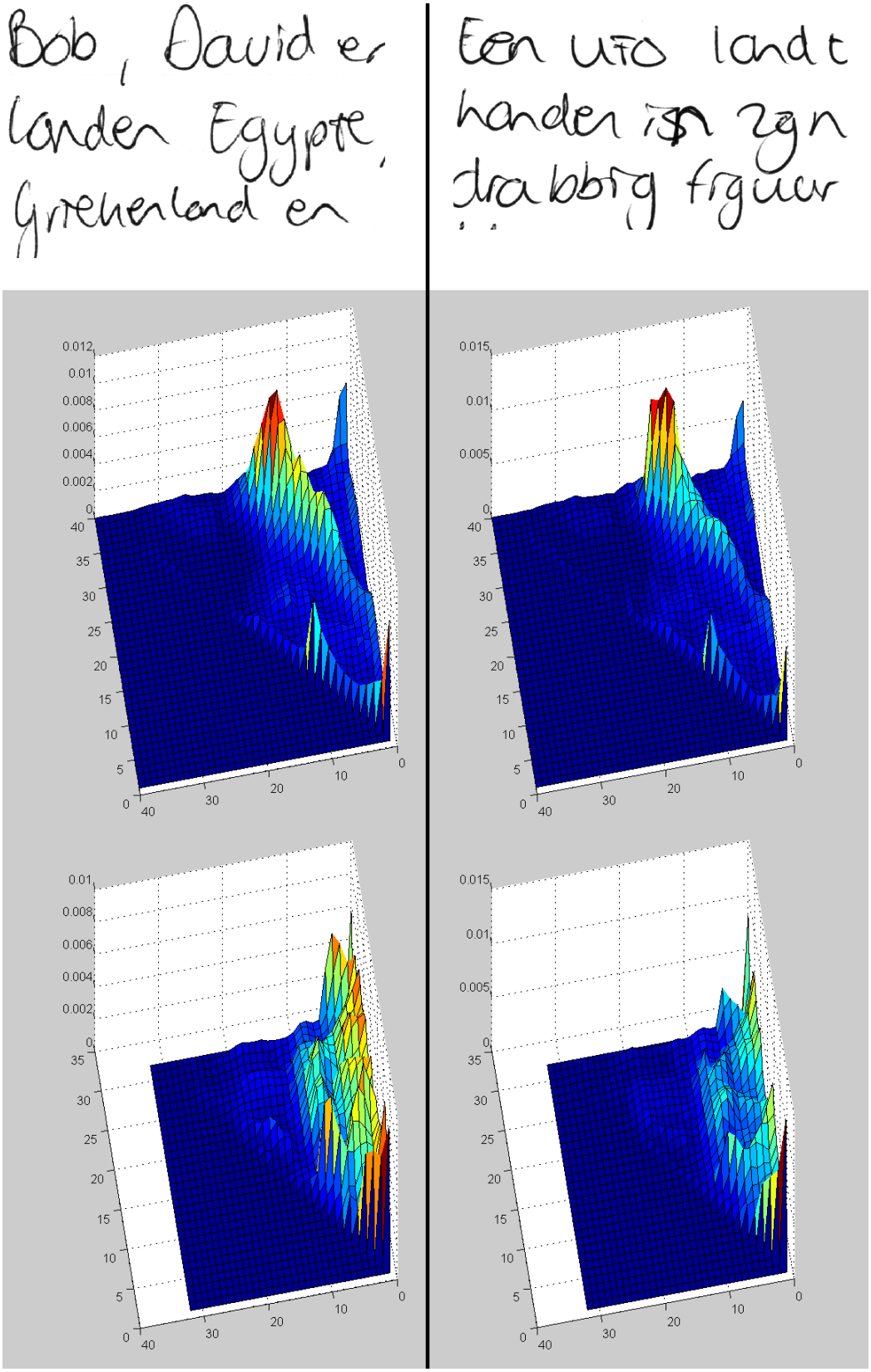

Figure 39. Text samples from the same writer along with skeleton hinge distribution feature surface (middle) and edge hinge combinations feature surface (bottom)

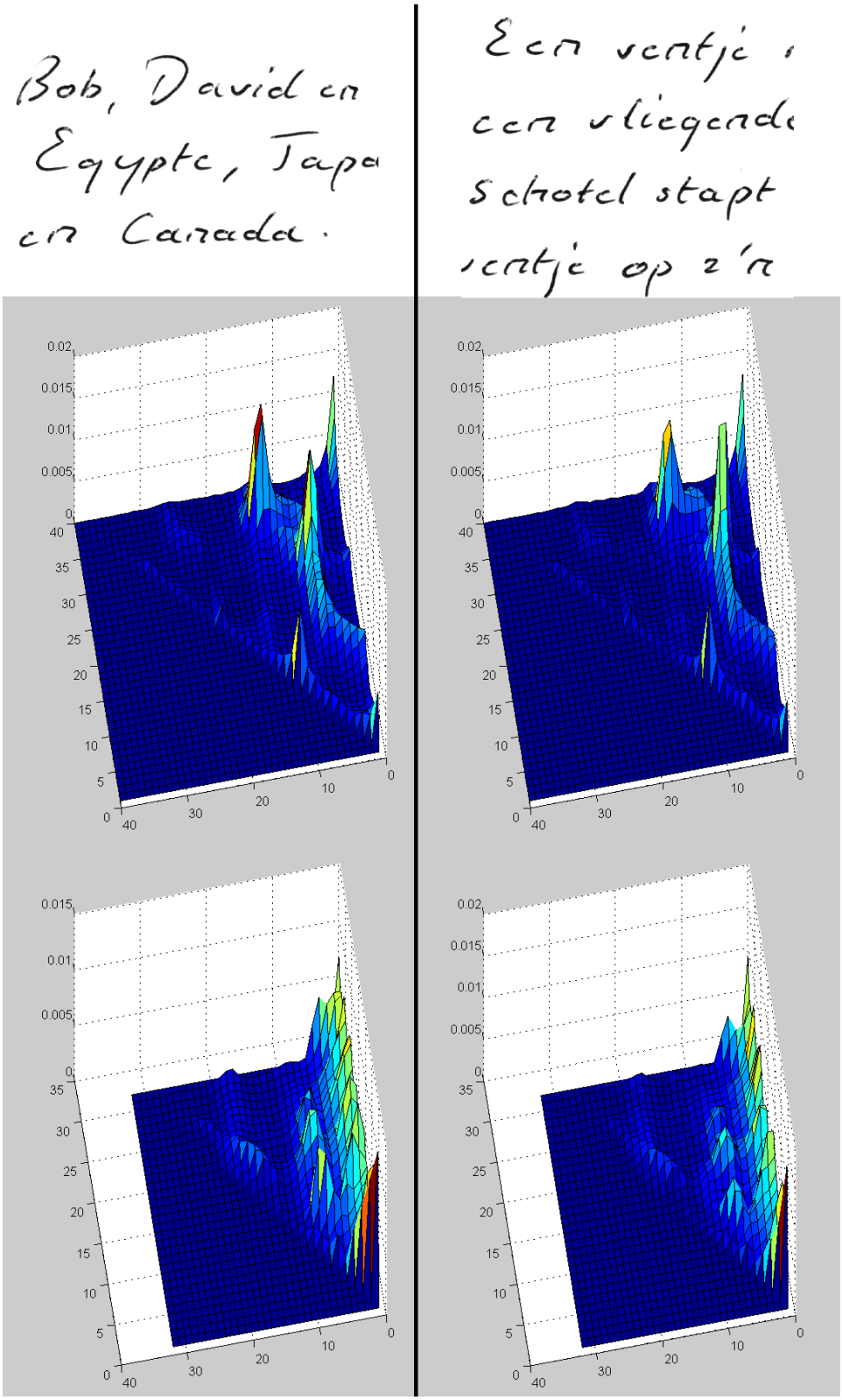

Figure 40. Text samples from the same writer along with skeleton hinge distribution feature surface (middle) and edge hinge combinations feature surface (bottom)

4.1.5 Weighted Skeleton-Hinge distribution

One of the factors affecting the performance of skeleton hinge distribution is the varying character size in a document. Imagine, for example, a capital “O” and a small “o”, the hinge angle on the small “o” is smaller than the hinge angle on the capital “O”. In the Weighted Skeleton-hinge distribution, the ratio of the varying character sizes found in a document image along with the normal character size of the document is considered Weight.

4.1.5.1 Main Body Size and Main Body Map

Usually, in document image processing systems, it is quite crucial to be able to identify the character size information easily. The Main Body or core region describes the central part of the text, between the upper baseline and lower baseline, excluding ascenders and descenders (Fig. 41), and it is usually referred to words. The purpose of this characteristic is to provide a reference for thresholds and sizes of lines, words and characters as it is directly related to the size of the characters [30].

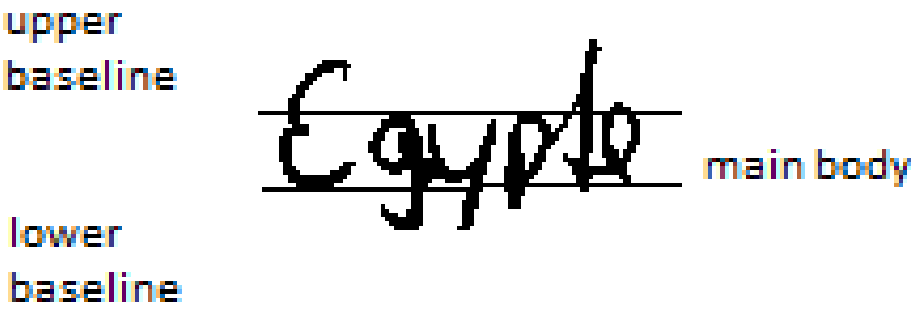

Figure 41. Word main body and baselines

In this work, the Main Body size is referred to small areas of text, usually one or more words, in a part of the text that has a length C.

The Main Body Map corresponds to an image, where, when a pixel is on (black pixel) on the document image, the intensity value on the map is the Main Body size value detected in that area of text. For an example of the Main Body Map, see Fig. 42 and 43. Pixels outside of the Main Body area, for example, above the upper baseline and below the lower baseline, are assigned with values equal to the most common Main Body size detected in the document image.

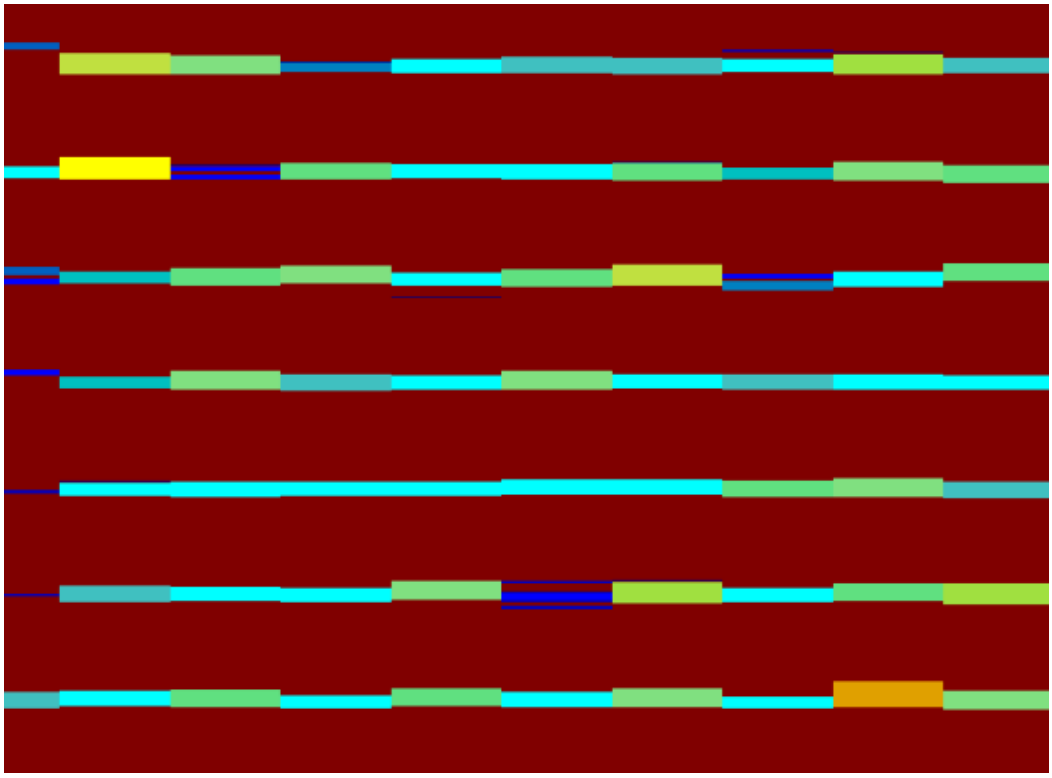

Figure 42. Main Body Map Example

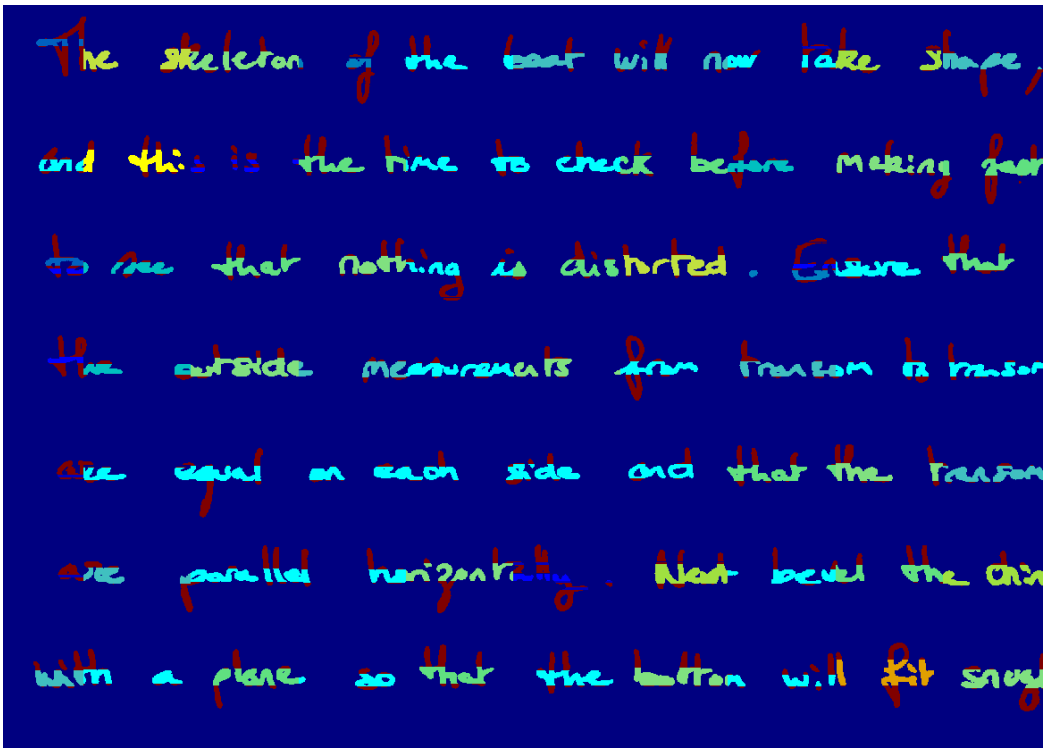

Figure 43. Main Body Map projected on the document image

4.1.5.2 Main Body Size extraction and weighted skeleton hinge

The Main Body size estimation technique starts by applying a smoothing technique that downsamples the image width while keeping the height the same. This is achieved with a horizontal smoothing procedure that uses the mean value of every C consecutive pixels. An example of the original text and the smoothed image can be seen in Figures 44 and 45. By observing the resulting image on figures 46 and 47 vertically, it is easily observable that the intensity levels follow a bimodal distribution (Fig 48). It is expected that by traversing the new smoothed image vertically, multiple Bimodal distributions will be observed. In Bimodal distributions, the external (e.g. top and bottom) modes are expected to take maximum values. For a large enough C, the external modes correspond to the word baselines (Fig. 41), allowing the estimation of the Main Body Size.

In the proposed MBS estimation methodology, the main idea presented above is used. The smoothed image is traversed vertical, and the bimodal distributions are identified by considering different thresholds. The threshold corresponds to the expected intensity value of the external modes. A range of thresholds is used. On every threshold used, the distances of the external modes detected are considered and stored in a histogram of external mode distances. The distance with the maximum value is considered the most common distance for the selected threshold. This value is stored in the second histogram of threshold distances. Finally, the distance with the max value in the histogram of threshold distances is selected as the most common MBS in the document. Furthermore, the MBS threshold can be found from the maximum frequency value of the MBS from all the histograms of external nodes.

Finally, the Main Body Map is constructed by extracting all the external modes, along with their distances, by using the MBS threshold. The intensities of the pixels that are on and between two external modes (upper and lower baselines) are set to the distance value detected for the specific modes. The intensities of the pixel outside of the Main Body area are set to the MBS value.

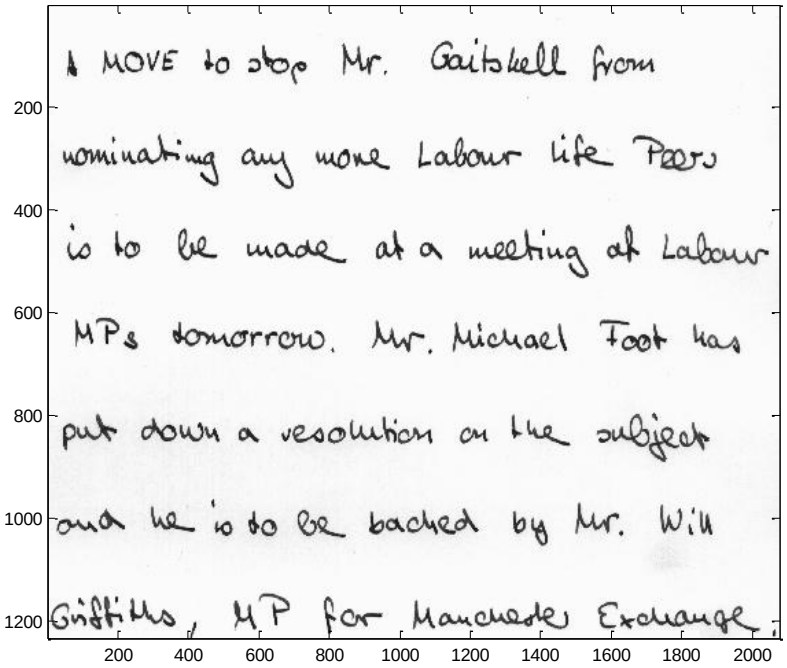

Figure 44. Handwritten document image with a resolution of 1232x2076

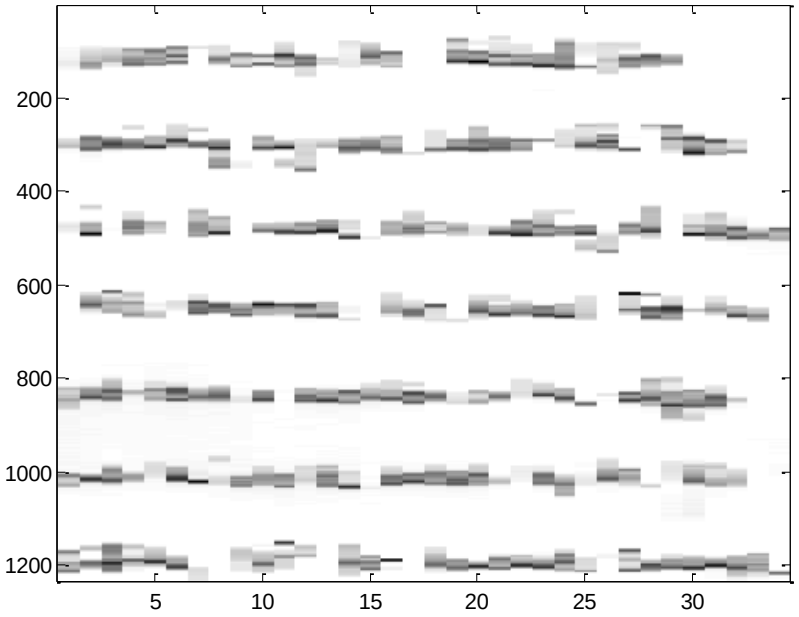

Figure 45. The smoothed version of the document image with the parameter C set to 60 pixels and resolution 1232x34

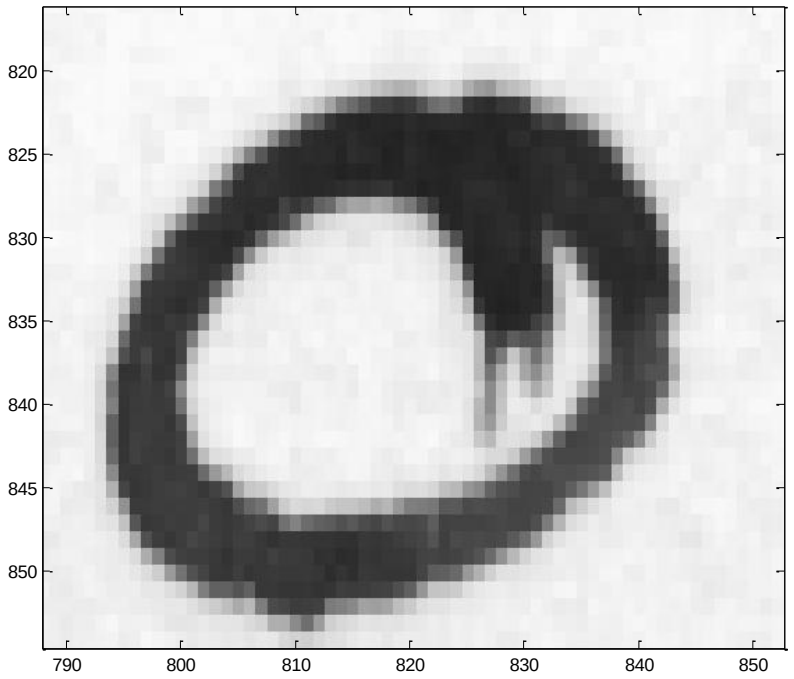

Figure 46. Example of the letter O

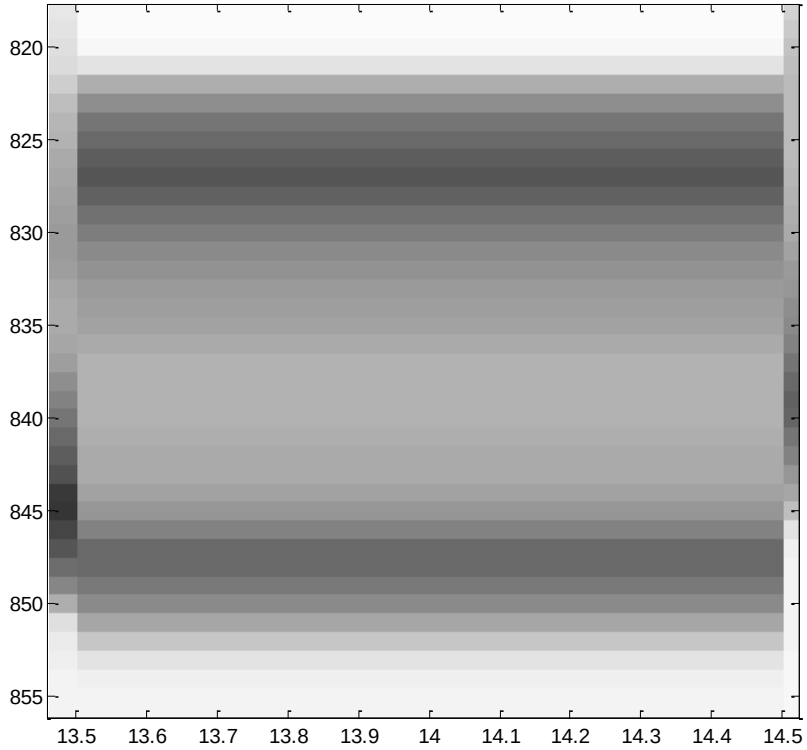

Figure 47. Example of the corresponding smoothed image of the letter O with a height of 35 pixels and width of 1.

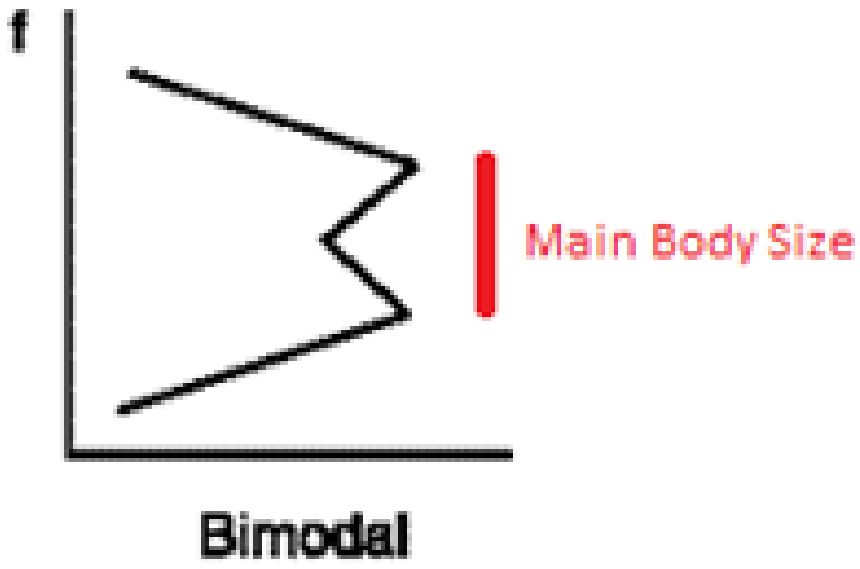

Figure 48. Bimodal distribution

While Skeleton Hinge distribution [97], treats all the skeleton fragments detected in the same way, in the proposed method, the weighted skeleton hinge distribution, the varying size of the text is considered. For example, in skeleton hinge distribution, a word with a large MBS will be treated the same as a word with a smaller MBS. In the proposed technique, the Main Body Map information is used, in order to give different weights on the skeleton hinge fragments detected, according to their central pixel value on the Main Body Map (MBMvalue) and its deviation from the MBS (MBSvalue). The weight considered is the corresponding value on the Main Body Map divided with the MBS value if the Main Body Map value is lower or equal to the MBS value.

Equation 3. Weight value if the Main Body Map value is lower or equal to the MBS value.

$$Weight(x,y) = \frac{MBMvalue(x,y)}{MBSvalue(x,y)}$$

On the other hand, if the Main Body Map value is greater than the MBS value, then the weight considered is

Equation 4. Wright value if the Main Body Map value is greater than the MBS value.

$$Weight(x,y) = 1 - \left(\left(\frac{MBMvalue(x,y)}{MBSvalue(x,y)} \right) - 1 \right)$$

4.1.6 Quantised Skeleton Hinge distribution

Pixel intensity information in Handwritten text is not uniform if we suppose the surface of writing is the same for all the writers; the pen is pressed with more power or less power during writing, depending on the angle, the character, and the written text. In Gray Scale images, we can use the information of pen pressure, denoted as pixel intensity, to augment the skeleton hinge information and try to prove that those points of pressure provide additional information for the identification of the writer.

In this method, the pixel intensity is quantised in N discrete values, and the number of quantizations is used to construct a 3-dimensional matrix with the third direction having a length of N. Quantized Skeleton Hinge distribution also starts with the image skeleton extraction using a generic skeletonisation approach [99]. Then a sliding window technique that uses several window sizes, each quantised in a different number of directions, checks for skeleton line fragments, which emerge from the central window pixel. Finally, their directions and the quantised intensity are measured and stored in triplets. Only skeleton line fragments with $\varphi 1 < \varphi 2$ are counted and stored in pairs in a histogram. A joint probability distribution $p(\phi 1, \phi 2, n)$ is obtained over a large sample of pairs. The probability distributions, acquired by the various sliding window sizes, are combined and considered for matching.

4.1.7 Directional Stroke Run Length Hinge distribution

While skeleton hinge distribution and weighted skeleton hinge distribution perform well, some information might be lost since the skeleton information is used. The same applies to the edge hinge distribution and edge hinge combinations with the edge information.

The main idea behind the Directional Stroke Run Length Hinge Distribution method is to consider all the available information in the document image by utilising run lengths. To achieve that, all the pixels that are on are considered. Next, a sliding window technique is used with various window sizes, each quantised in directions. On every central pixel that is on the black run lengths, in eight directions are considered. Only the two directions with the most significant run lengths are kept. Starting from the direction of the maximum run length, the next pixel is selected by following that direction, and the run lengths of the five directions that emerged from that pixel are considered. The three directions excluded are the opposite direction of the previously selected direction and the two neighbouring directions. This process is repeated by following the pixels found on the largest run lengths from the five directions until the window border is reached. The same technique is applied in the second-largest direction that was initially kept. Finally, the directions of the two run-length directional fragments are measured and stored in pairs—only fragments with $\varphi 1 < \varphi 2$ are counted and stored in pairs in a histogram. A joint probability distribution $p(\phi 1, \phi 2)$ is obtained over a large sample of pairs. The probability distributions, acquired by the various sliding window sizes, are combined and considered for matching. For example, visualising some of the steps of this technique, see figures 49 to 59.

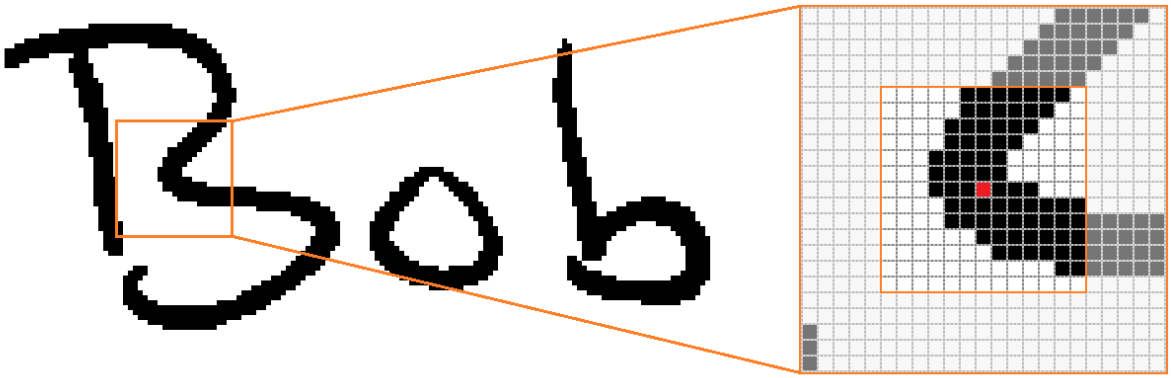

Figure 49. An instance of Stroke Run Length Directional Hinge window of 6 pixels-long fragments with the central pixel selected as starting point.

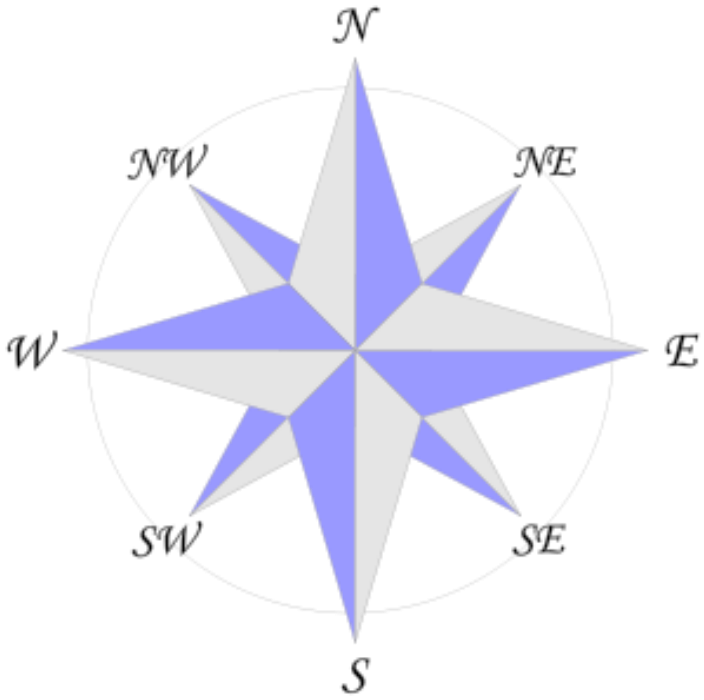

Figure 50. cardinal and intermediate directions will be used to describe run length directions

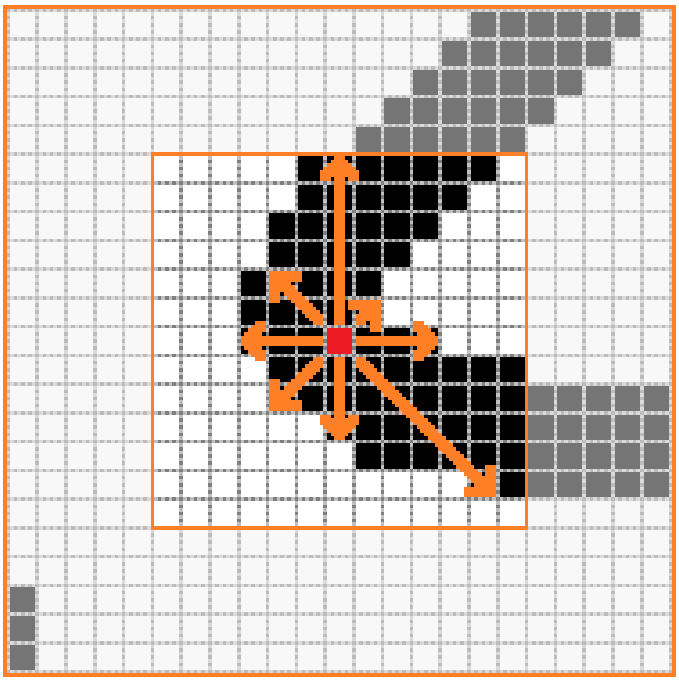

Figure 51. The selected Starting point along with the run lengths in 8 directions with the largest one being 6 pixels length with North direction and the second largest with 5 pixels length on South-East direction

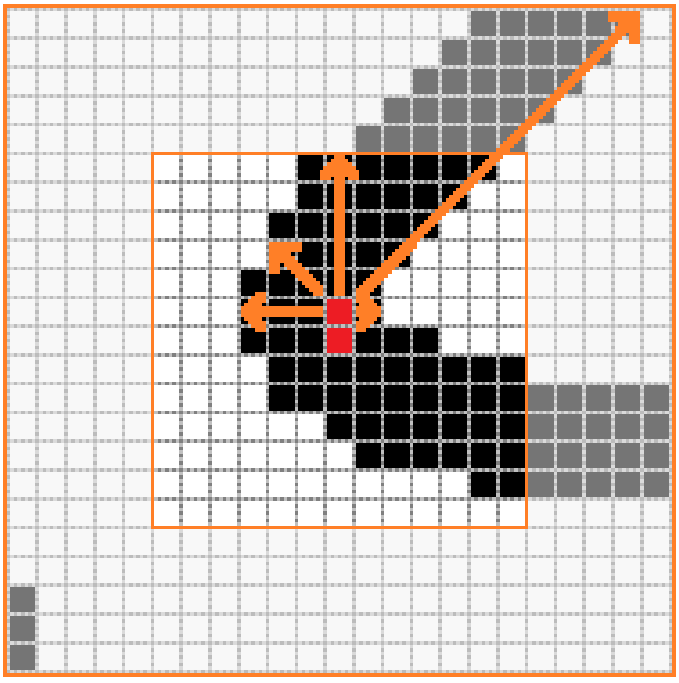

Figure 52. The five directions emerging from the second point along with the run lengths in five directions, with the largest one being in the North-East direction

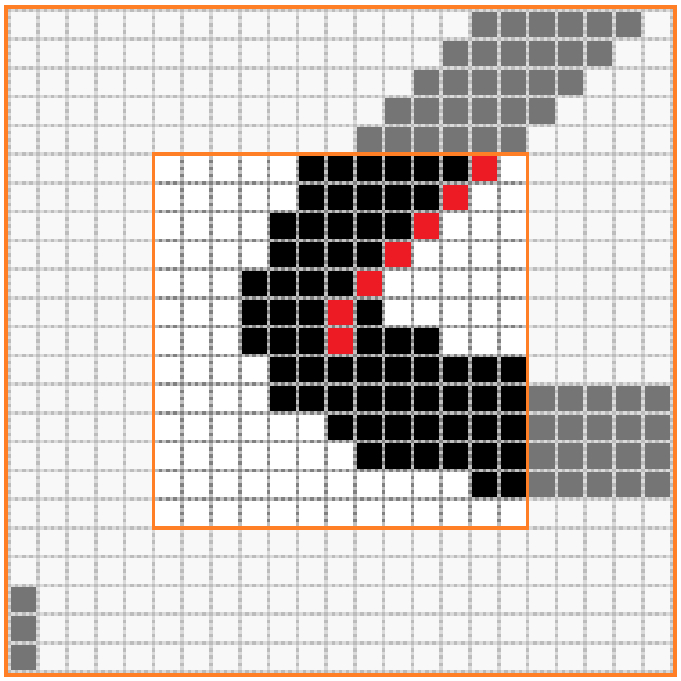

Figure 53. The Largest Run Length is followed until the border is reached

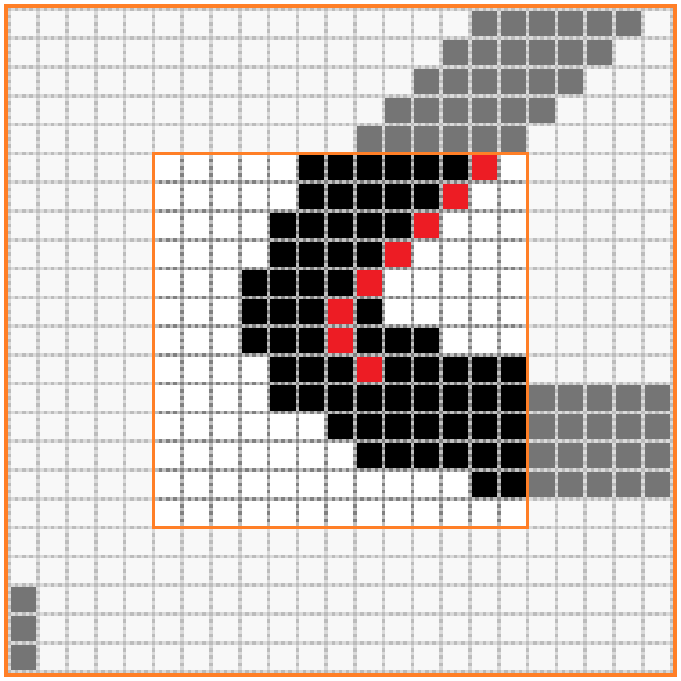

Figure 54. The next pixel is selected in the direction of the second largest direction

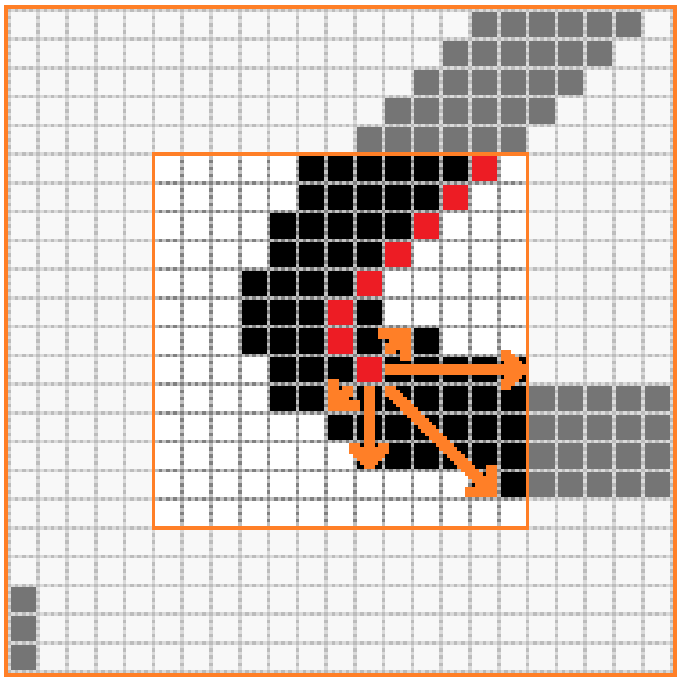

Figure 55. The five directions emerging from the first point on the second-largest direction along with the run lengths in five directions, with the largest one being in the East direction

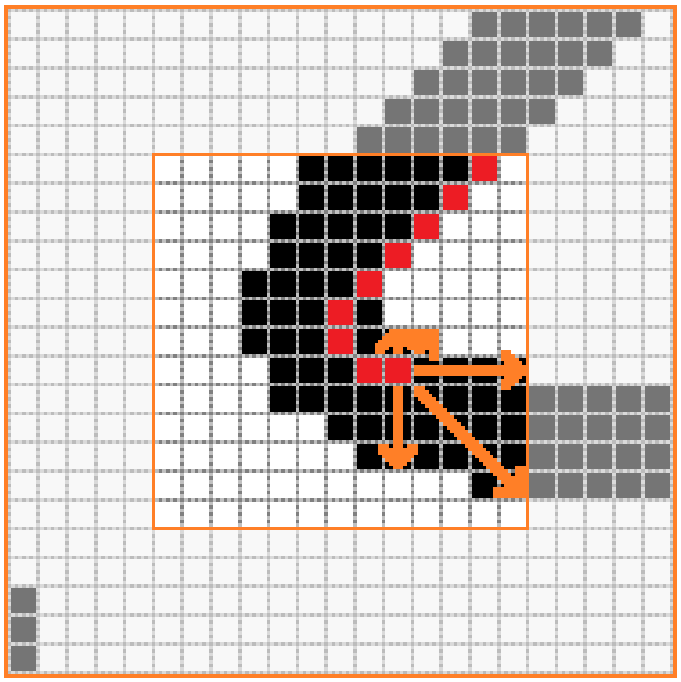

Figure 56. The five directions emerging from the second point on the second-largest direction along with the run lengths in five directions with two directions of equal length of four pixels. The East direction is selected since it was also the direction selected on the previous step

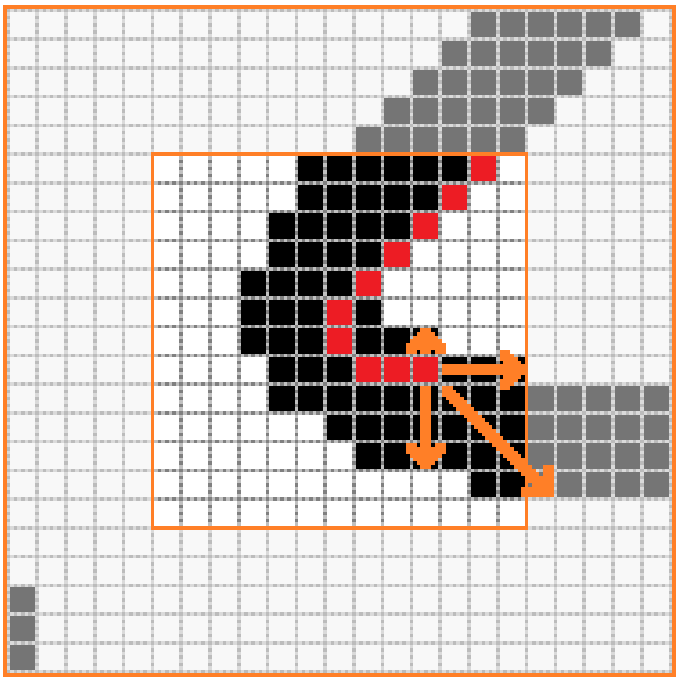

Figure 57. The five directions emerging from the third point on the second-largest direction along with the run lengths in five directions, with the largest one being in the South East direction

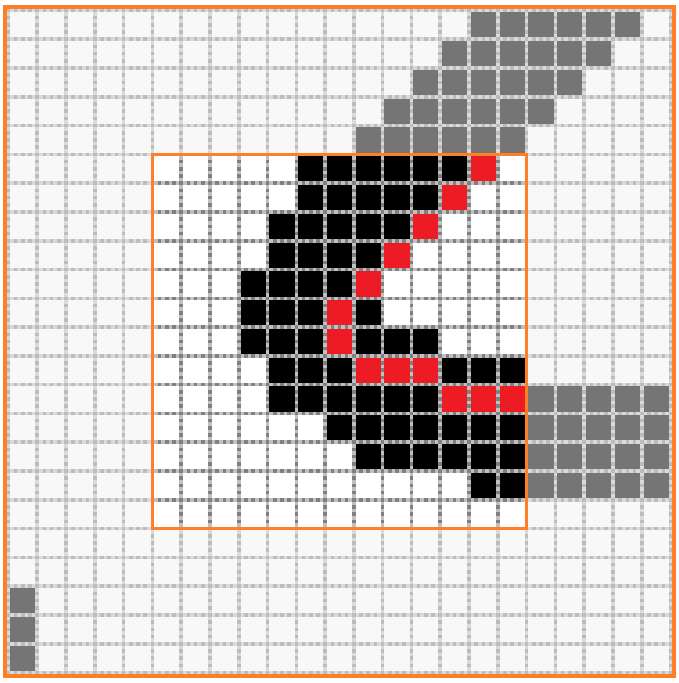

Figure 58. The Largest Run Length is followed until the border is reached

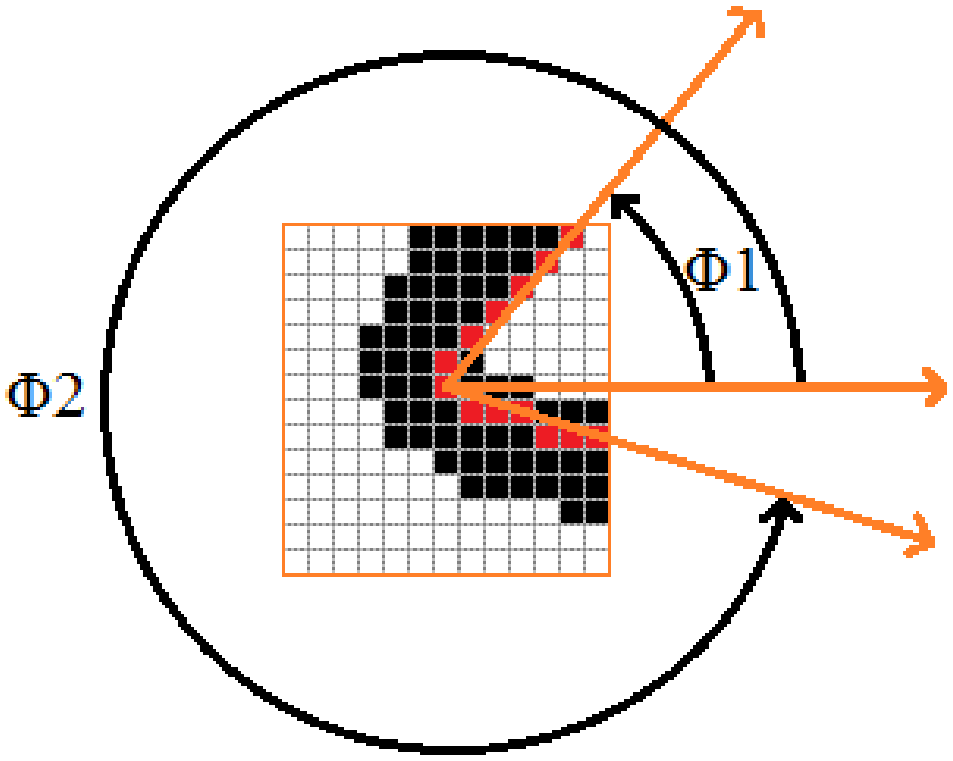

Figure 59. Final Stroke Run Length Directional Hinge with six pixels-long fragments

4.1.8 Edge-Skeleton-Hinge Combinations

An attempt was made to fill the entire feature space with information on the Edge-Skeleton Hinge Combinations method. All the feature spaces of the previous techniques have one thing in common: the bottom left part of the feature space, and the diagonal line from top left to bottom right is empty. This happens because the directions are considered only when $\varphi 1 < \varphi 2$. For example, see Fig. 60

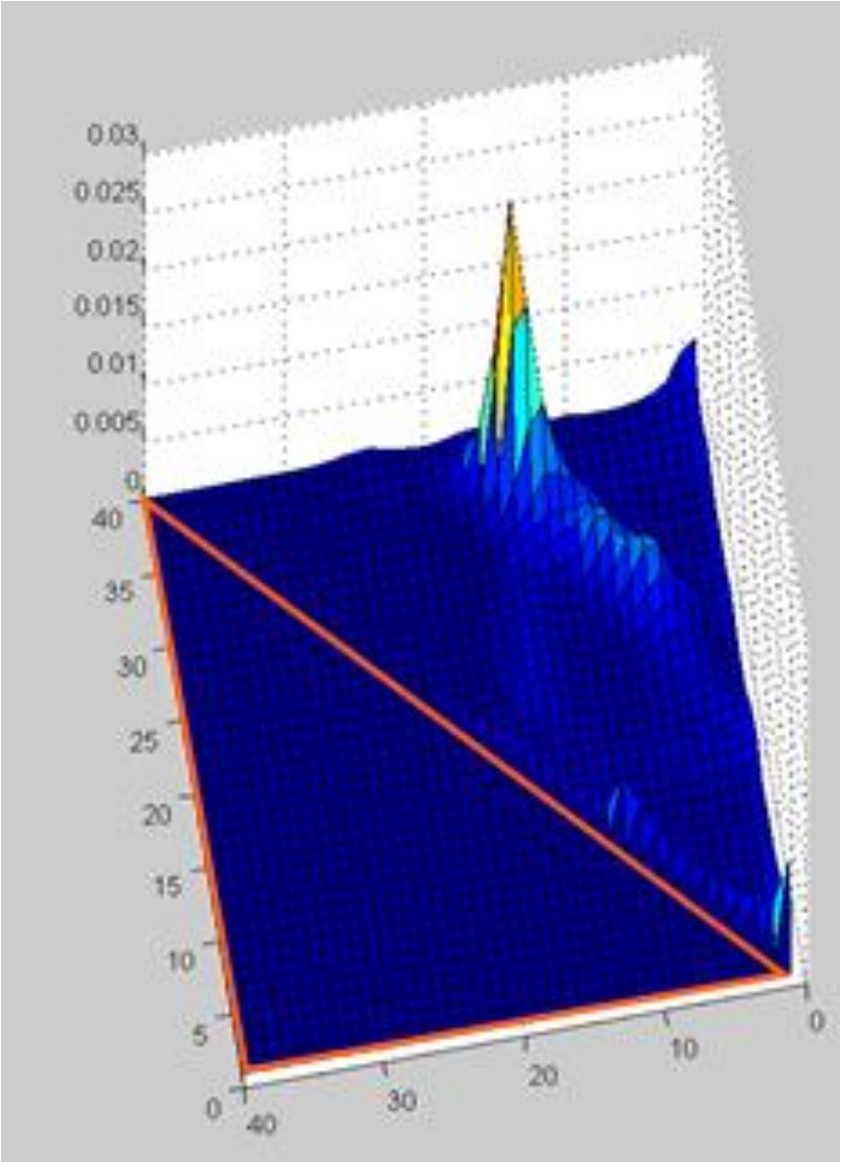

Figure 60. Skeleton-Hinge Feature with the area denoted with the triangle being empty

To achieve that, both features, the Skeleton Hinge and the Edge hinge, were considered. The skeleton Hinge feature space was saved in the upper right side of the feature space using the probability distribution Ps(Φ1, Φ2), while the Edge Hinge feature was saved in the bottom left side using an inverse probability distribution Pe(Φ2, Φ1). In Figures 61 and 62, examples from the test set and the train set are presented.

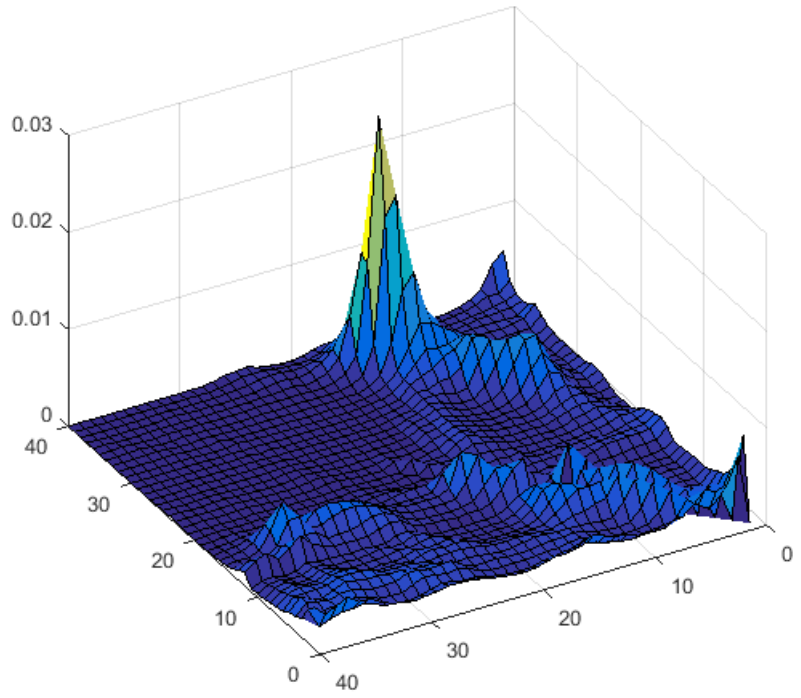

Figure 61. Feature spaces from the Edge-Skeleton Hinge Combinations on the Test sample

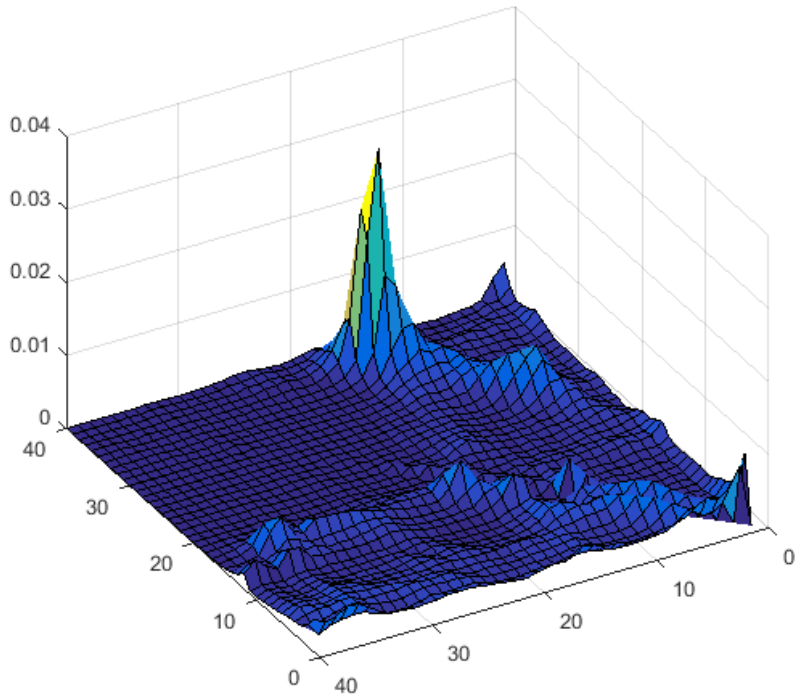

Figure 62. Feature spaces from the Edge-Skeleton Hinge Combinations on Train sample from the same writer.

In the second variation of this technique, an attempt was made to also fill the diagonal line from top left to bottom right. In order to achieve that, we used Φ as the difference of Φ2 from Φ1 and the probability distribution Pd(Φ, Φ)

Equation 5. Φ angle as the difference of Φ2 from Φ1

$$\Phi = \Phi 2 - \Phi 1$$

4.2Model-Based Features

In the Model-Based approach used in the works [34, 26], it is assumed that each writer produces a recognizable set of writer specific character shapes or allographs. This happens due to schooling and personal preferences. The core idea reflected in the above statement implies that a histogram of used allographs can characterise each writer. However, it is not feasible to have a predefined list of allographs. Instead, training is needed to automatically generate a codebook, which sufficiently captures allograph information from samples of handwriting.

The approach used in this work relies on a codebook of models of graphemes. Graphemes are small strokes of handwriting, which are extracted by applying a robust segmentation algorithm on a handwritten image. It should be mentioned that there is a distinction between graphemes and the fragments used in the statistical methods because of the different algorithms in use.

In Schomaker et al. [34], a codebook of graphemes is generated by training a Kohonen SOFM [100] on a large number of grapheme contours. The produced codebook is later used to construct feature vectors.

The process used to create feature vectors from the codebook is quite simple: From each text image, all graphemes are extracted and matched to the grapheme models of the codebook. Euclidean distance between the grapheme contours is used for the matching process. For each grapheme model in the codebook, every successful match is counted. The result is a histogram of graphemes, which characterize the writer and also identify him.

A limitation in this approach is the long training time of Kohonen SOFM. As reported in [34] a training time of up to 122 hours can be required. Besides that, Kohonem SOFM may get stuck in local minima.

Van Der Maaten et al. [26] proposed the use of random selection for the creation of graphemes rather than using Kohonem SOFM. In this method, no time-consuming training is performed, overcoming the time limitation. Instead of training, a random number of graphemes are drawn from the large set of graphemes.

Both approaches, when were combined with the edge-hinge feature, achieved an identification performance of 97% on the Firemaker DB for 150 distinct writers and a codebook of 400 graphemes.

Here, an improvement was attempted, using a different approach on the codebook generation, by only considering closed areas of the characters. Character closed areas are the least affected by writer slant, very important as slant is a characteristic of the writer that can affect the skeleton hinge distribution.

By combining skeleton hinge distribution with a codebook of graphemes only generated by character closed areas, it was expected to be an ideal way of securing skeleton hinge distribution against forge attempts. A forge attempt can be made by merely changing the slant. However, the results of this approach were not the expected ones.

Chapter 5

5. Directional Feature Interpretation

In this chapter, an attempt is made to explain and interpret the characteristics of handwritten text captured by the Directional Hinge methods described above. The common denominator of all the mentioned Directional methods that use a probability distribution $P(\varphi1,\varphi2)$ is that they capture information about the slant and the curvature of the handwritten text.

Slant is a salient feature of western handwriting [101] and is defined as the predominant angle of the downward stroke. Slanted characters can slope either to the left or right, although the slant is not always uniform and can change even in the same word. For an example, see Figures 63 and 64.

The importance of Slant on the task of Writer Identification systems can be seen in the various works found in the literature like [55, 41, 102, 103]. Furthermore, Forensic document examiners also find slant to be a significant consideration [104] and a discriminatory characteristic [105] to identify the writer. Again, Slant is among the most visible attributes of handwritten text, along with size and spaces. One of the most common characteristics in cases of forgery, i.e. when a writer tries to mimic other writers handwriting, is to try and imitate the writing slant. Although according to [106], the forger, while copying the Slant, might lose his attention for a moment and revert to his own unique style. For a forensic document examiner, this sudden and brief Slant change is suspicious and might contain information about the forger identity.

Finally, in cases of disguise, i.e. the writer tries to hide his identity by changing his writing style, and he achieves that by changing his Slant of writing. In experiments performed in [101], they found that while Slant is very important for writer identification, it is not as essential as believed as an isolated factor. This is also observable in the Edge-Direction distribution [25] that mostly captures the Slant of writing by the low accuracy achieved on the Firemaker dataset.

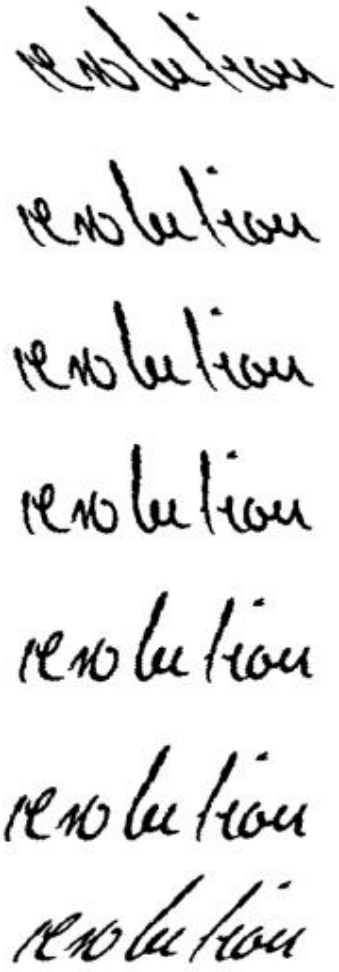

Figure 63. An example of different Slant Angles from left to right. Graphic is from [107]

Figure 64. An example of right slanted (a), left slanted (b) and variant slanted(c) word. Graphic is from [108]

Slant angle, for most writers, is visible in the feature space when it is projected to polar coordinates. For example, in Figures 65,66,67,68, two parts of handwritten pages can be seen along with their feature space projected in polar coordinates. For the first writer, i.e. Figures 65 and 66, it is observable that the slant of writing is precisely 90 degrees. The φ1 angle distribution maxes out at 90 degrees, while the φ2 angle distribution maxes out at 270 degrees. By following the peaks of φ2 towards φ1, the slant angle of writing is found. On the second writer, i.e. Figures 67 and 68, a right slant can be observed with an angle of approximately 81 degrees. It should be noted, however that the Slant could not be found in all the cases by finding the max of the angle distributions in the polar plot. This happens because the angles captured in the Directional feature come from all kinds of strokes, and not only the downward strokes.

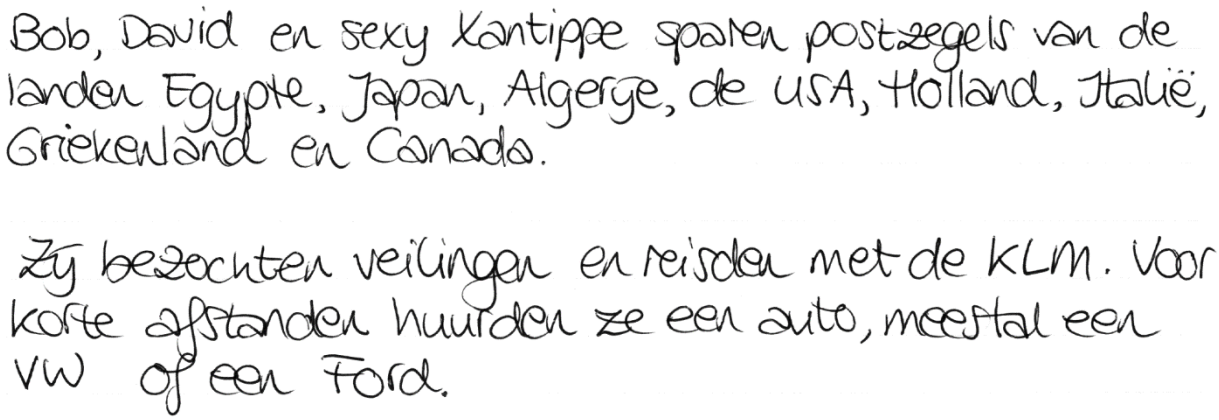

Figure 65. Part of text from Firemaker DataSet from writer 1657

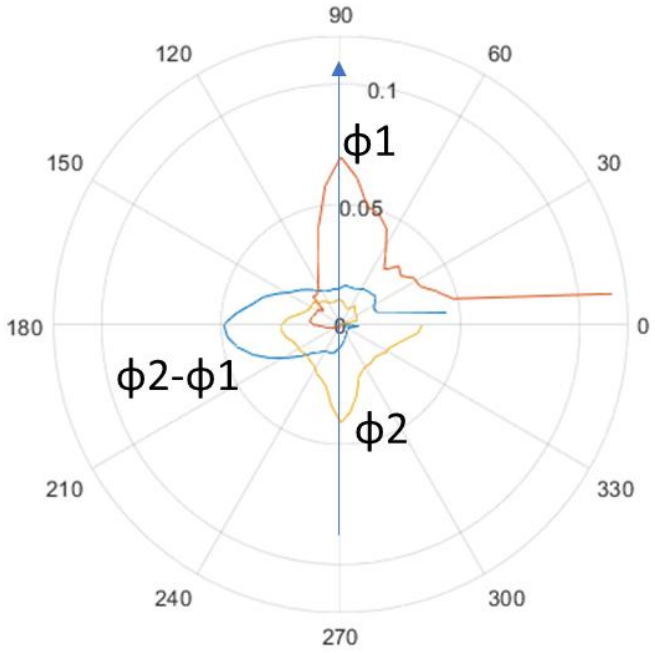

Figure 66 Polar Plot of the angles φ1, φ2 and their difference φ2-φ1 for writer 1657

Bob, David en sexy Xantippe sparen postzegels van de landen Egypte, Japan, Algerije, de USA, Holland, Italië, Griekenland en Canada.

Zij bezochten veilingen en reisden met de KLM. Voor korte afstanden huurden ze een auto, meestal een VW of een Ford.

Figure 67 Part of text from Firemaker DataSet from writer 17

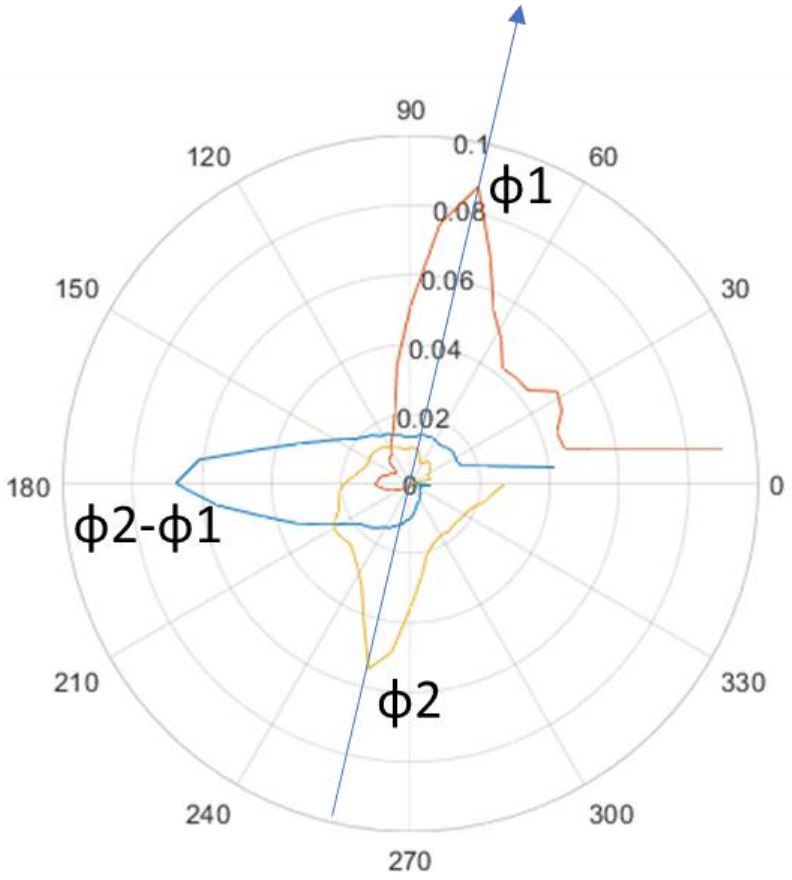

Figure 68 Polar Plot of the angles φ1, φ2 and their difference φ2-φ1 for writer 17

Curvature results from the movement of the wrist and the fingers [109] and in the spatial domain is expressed by the angular information of the handwritten curves [25]. It is also an essential characteristic that plays a significant role in Writer Identification [25, 26].

The curvature information also exists on figures 66 and 68, hidden in the probability distribution of all the φ1 angles and φ2 angles and their difference (φ2-φ1). Although the curvature is not quite as visible as Slant in the polar plots as a feature, its importance can be realized through the experimental results presented in chapter 6.

Although Slant and Curvature are significant features for writer identification, that doesn't mean that they can uniquely identify the writer. It is interesting to observe the rare cases of false matching and the feature similarity between two distinct writers.

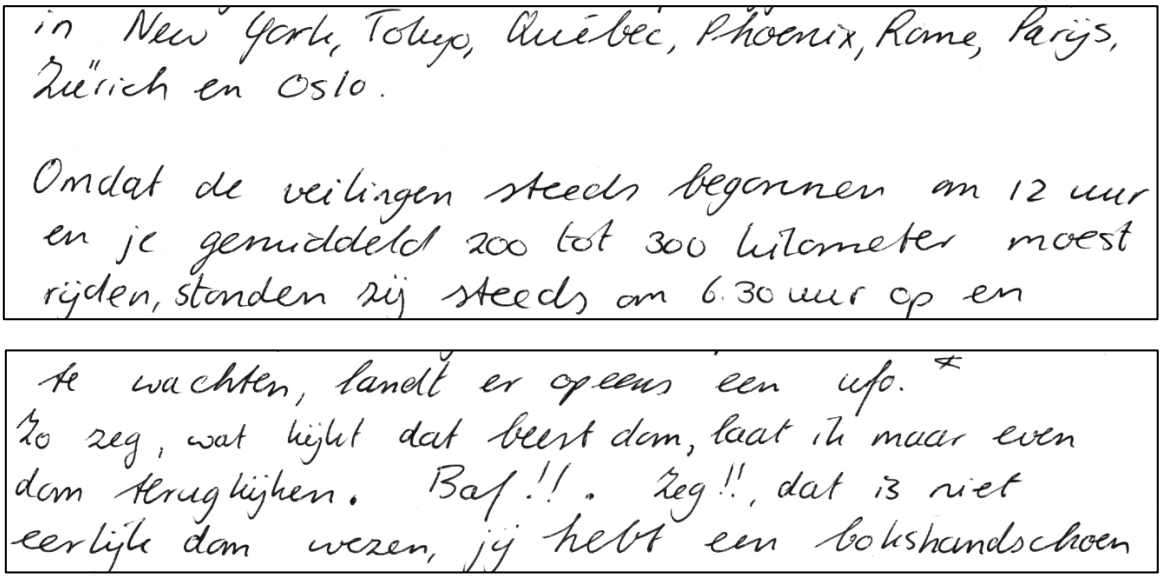

Figure 69 Train and Test samples from writer 52 of Firemaker Dataset

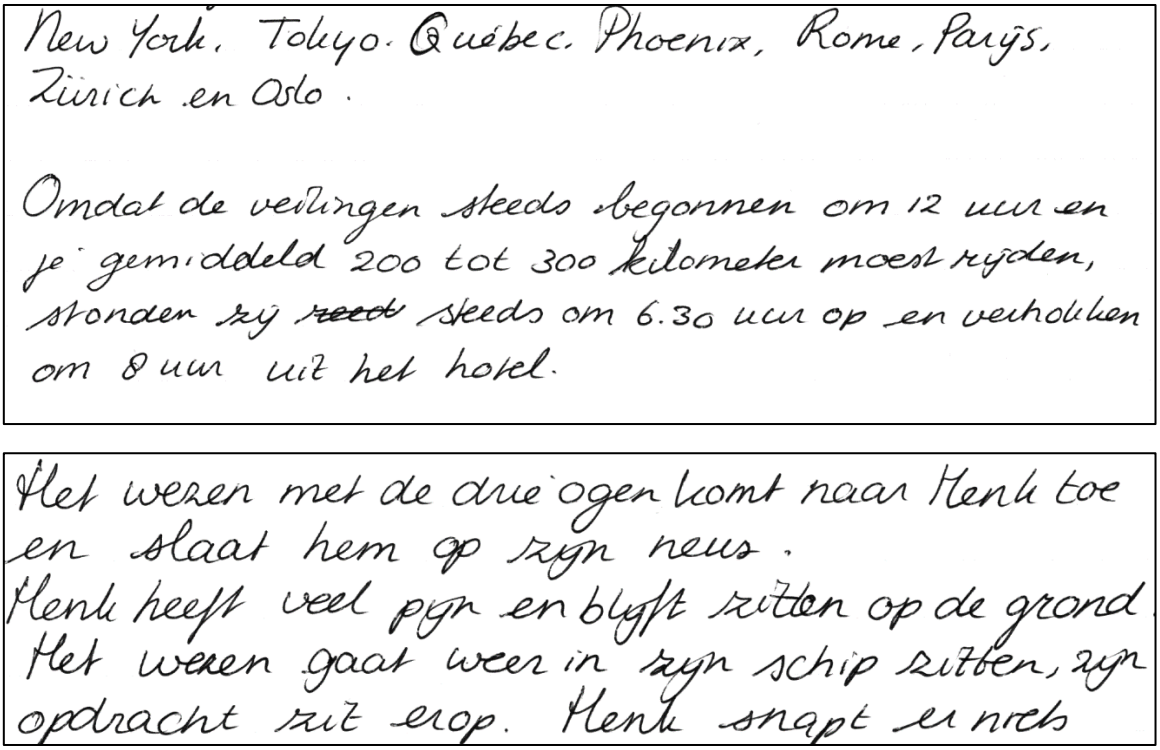

Figure 70 Train and Test samples from writer 23 of Firemaker Dataset

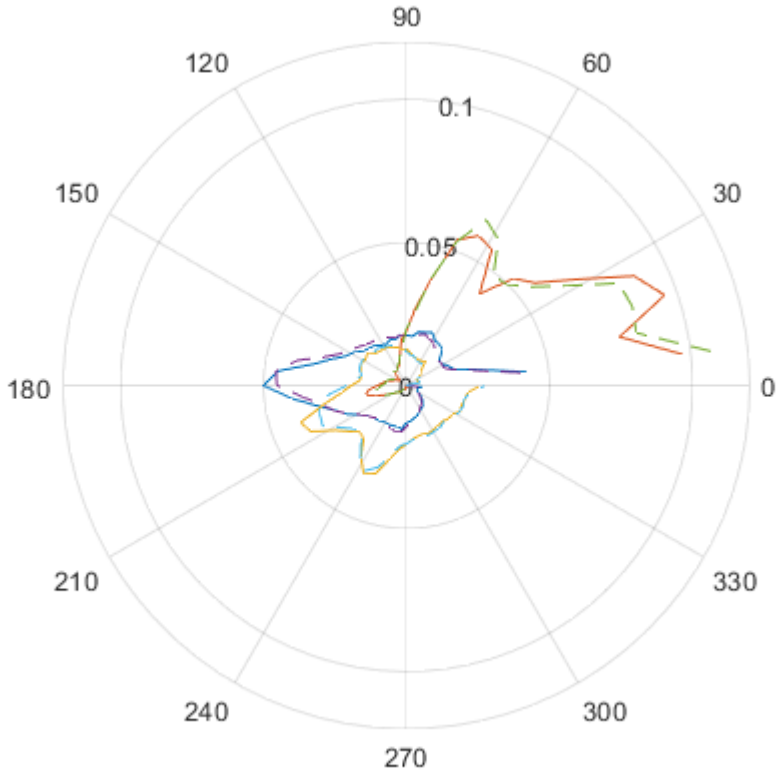

Figure 71 with solid lines writer 52 from train dataset and with dashed line writer 23 from test dataset.

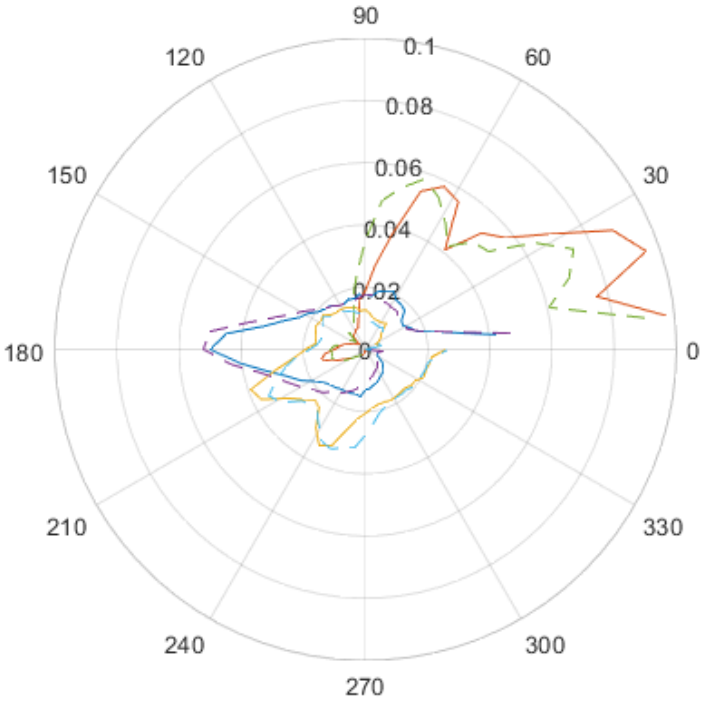

Figure 72 writer 52 differences between train and test dataset

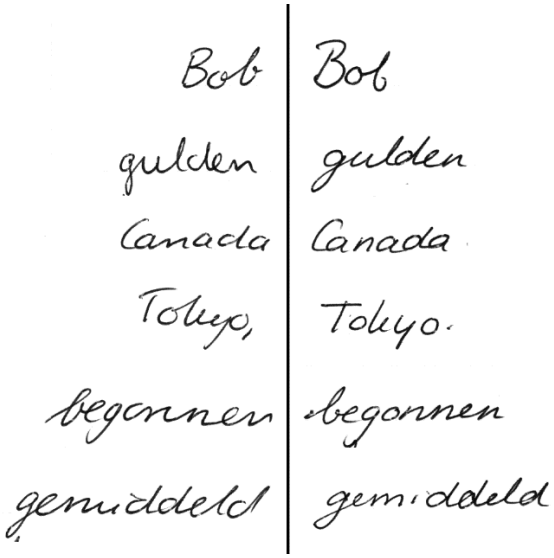

Figure 73 Same words from writer 52 and 23 from Firemaker data set

Although it might be an easy task for a forensic document examiner to distinguish between the two writers on Figures 69 and 70, Directional Distribution systems have a hard time doing so. The extracted angular information between those writers is so similar that even the same writer on train and test documents have more dissimilarities. Furthermore, a selection of words was made to showcase the similarities of the two writers in Figure 73. Someone without a forensic background and expertise could think it is the same writer.

Chapter 6

6. Experimental Results

6.1 Data Sets

6.1.1 Firemaker DB

One of the datasets used to evaluate the feature extraction techniques presented in this work was the Firemaker Database [13]. This data set was used to directly compare the achieved results with the reported ones by the other methods.

The Firemaker is a database of handwritten pages from 250 writers, including four pages per writer.

- Page 1 contains a copied text in natural writing style
- Page 2 contains a copied text in Upper-case text
- Page 3 contains copied forged text. The writers here try to impersonate another writer.
- Page 4 contains a self-generated description of a cartoon image in free writing style. In this last page, the text content and the amount of written ink varies considerably per writer.

All pages in Firemaker Database were scanned at 300-dpi grayscale. The text that was asked to be copied was specially designed in forensic praxis to cover a sufficient amount of different letters of the alphabet. In our experiments, only pages 1 and 4 were used. Page 1 was used as a train set. While page 4, was used as a test set.

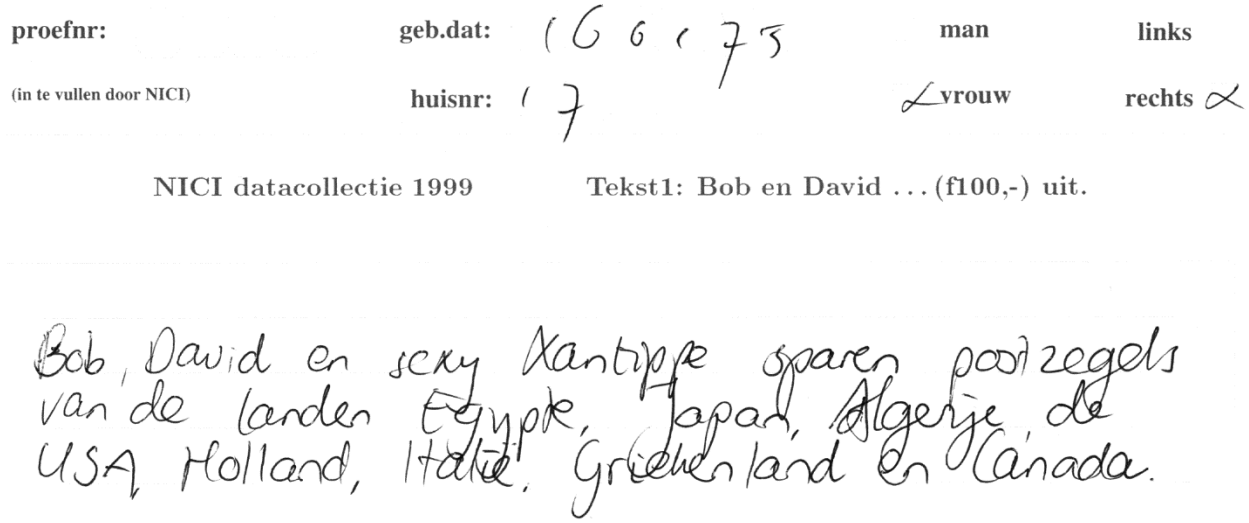

Figure 74. Part of a Document image from Firemaker DB

6.1.2 ICDAR 2017 Writer Identification Competition

Additional experiments were performed on the Skeleton Hinge Distribution feature using the ICDAR 2017 writer identification competition [14] dataset. The dataset used for this competition consists of 3600 document images, which 720 different writers have written. Each writer has contributed five documents. The performance was evaluated using ScriptNet, the competition platform which can output the mean average precision (map) and top 1 performance.

6.2 Experiments

6.2.1 Skeleton Hinge Distribution

Various experiments were performed, using combinations of several parameters, e.g. window sizes and matching classifiers. It is extremely hard to compare our results with results reported on other papers because of the variation in the data sets. Our results will be only comparable to methods that used the same dataset.

Furthermore, even on the same dataset, results can have a significant variation. Some methodologies only used a fragment of the entire dataset, without mentioning which one exactly. Also, there are differences in train and test sets. Even a slight change in these sets can change the entire outcome.

To train the system (Extract Skeleton Hinge training Features), only page 1 from the Firemaker DB was used. Each page was binarised, and the skeleton was extracted using Matlab. The used procedure is the one described in the previous section for skeleton hinge distribution.

The training procedure was fast, about 250 seconds on a laptop i7 2.5Ghz pc, and in comparison to the edge hinge distribution, about 35% faster. On the same machine, the edge-hinge distribution train took 384 seconds to complete.

To extract Skeleton Hinge test Features, only page 4 was used from the Firemaker DB. Testing process used the same procedure as the training process.

The test procedure was faster than training due to the variations in text sizes in page 4. Testing took around 200 seconds on a laptop i7 2.5 Ghz. Edge hinge distribution time was about 270 seconds. An improvement of about 35% can be observed here, too.

Different matching techniques were considered for writer identification—maximum accuracy achieved with the Nearest Neighbour classifier with Manhattan distance. Euclidean and chi-square distances were also considered for classifying, but they performed worse. KNN classifier was also considered with k 1.

Furthermore, clustering techniques, like K-means and Agglomerative Hierarchical Cluster Trees, and also machine learning techniques, like SVM, where considered.

6.2.1.1 Skeleton Hinge Features with the nearest neighbour classifier on Firemaker DB

Skeleton Hinge Distribution Feature identification results are presented in Table 3. These experiments used the entire data set of 250 writers. Like the edge-hinge combinations method, a combination of fragment lengths, i.e. window sizes, were used. Furthermore, for the nearest neighbour classifier Manhattan, Euclidian and chi-square distances were used. Our top result is identification accuracy of 90.8 % for a combination of fragment lengths of 5- and 9-pixel length window and Manhattan distance.

Table 4. Skeleton Hinge Distribution Accuracy (Percentage) on Firemaker DB

Fragment Length	**Skeleton Hinge Distribution Accuracy (Percentage)**		
	Manhattan Distance	***Euclidian Distance***	***Chi-square Distance***
3	80%	72%	53.2%
5	89.6%	77.2%	66%
7	90%	81.6%	69.6%
9	88%	85.2%	76%
3 , 5	85.2%	75.2%	58.4%
3 , 7	85.6%	75.6%	55.2%
3 , 9	86%	74.8%	53.2%
5 , 7	90%	78.8%	64.4%
5 , 9	**90.8%**	78.8%	67.2%
7 , 9	90%	83.2%	73.6%
3 , 5 , 7	86.8%	76.8%	60%
3 , 7 , 9	89.6%	76.8%	55.6%
5 , 7 , 9	90%	79.2%	68.8%
3 , 5 , 7 , 9	89,6%	76.8%	60.4%

6.2.1.2 Skeleton Hinge Features with the nearest neighbour classifier on ICDAR 2017

Skeleton Hinge Distribution Feature identification results or ICDAR 2017 dataset are presented in Table 4. These experiments used the entire data set of 3600 document images, which 720 different writers have written. Like the edge-hinge combinations method, a combination of fragment lengths, i.e. window sizes, were used. For the nearest neighbour classifier, Manhattan distances were used. Our top result is identification accuracy of 68.44% with a mean average precision (map) of 47.02% for a combination of fragment lengths of 3-5-7 and a 9-pixel length window. Finally, in Table 5, an overview of results as reported in [14] compared to our results is presented.

Table 5. Skeleton Hinge Distribution Accuracy (Percentage) on ICDAR 2017 writer identification competition Data Set

Fragment Length	***WI-map***	***WI-precision***
3	40.57%	60.83%
5	44.80%	66%
7	43.96%	64.63%
9	40.99%	61.41%
3 , 5	44.50%	65.58%
3 , 7	46.20%	67.33%
3 , 9	46.33%	67.52%
5 , 7	45.98%	67.22%
5 , 9	46.34%	67.66%
7 , 9	43.73%	64.33%
3 , 5 , 7	46.34%	67.75%
3 , 7 , 9	46.92%	68.36%
5 , 7 , 9	46.11%	67.33%
3 , 5 , 7 , 9	**47.02%**	**68.44%**

Table 6. Skeleton Hinge Distribution Accuracy (Percentage) on ICDAR 2017 writer identification competition Data Set as reported in [14]

Method	***WI-map***	***WI-precision***
Skeleton Hinge	**47.02%**	**68.44%**
Barcelona	45.9%	67%
Fribourg	30.7%	47.8%
Groningen	54.2%	76.1%
Hamburg	46.9%	67.1%
Tebessa I	52.5%	74.4%
Tebessa II	55.6%	55.6%

6.2.1.3 Skeleton Hinge Features with K-means and Hierarchical cluster tree identification Results

An attempt was made to identify writers using the k-means algorithm and partitioning the collection into clusters on Firemaker DB. The entire collection consisted of 250 writers with two pages per writer, one page in training data, and one page in test data. Skeleton hinge distribution features were extracted from 500 pages and partitioned into 250 clusters. Standard K-means technique was used, as well as Kmeans with different distance parameters were explored. Only clusters that included both pages from each writer were considered as correctly identified. Identification accuracy reached 66.8% using [3 5 7 9] skeleton hinge distribution combinations.

Furthermore, experiments of clustering the 500 pages using agglomerative hierarchical cluster tree were made using two parameters, Agglomerative clusters from linkages and Agglomerative clusters directly from data. Only clusters containing both pages from the same writer were considered as correctly identified. Accuracy in both methods reached 63.6% using [3 5 7 9] skeleton hinge distribution combinations.

Skeleton Hinge Distribution Features clustered with K-means and Hierarchical cluster tree identification results are presented in Table 6.

Table 7. K-means and Hierarchical cluster tree identification Results on Firemaker DB

Clustering Method	Parameter	Accuracy
K-means	normal	66.8%
K-means	cityblock	46.4%
K-means	cosine	66.8%
K-means	correlation	66.8%
Hierarchical Cluster Tree	linkages	63.6%
Hierarchical Cluster Tree	data	63.6%

6.2.1.4 Skeleton Hinge Features with Nearest Neighbor using KNN results

Besides using a simple Euclidean distance measure, the KNN algorithm was used to find the nearest neighbour of every document of the training set in the test set on Firemaker DB. From each set, skeleton hinge distribution combinations with fragment lengths 3,5,7,9 were extracted. Different distance measures were used. In most of them, accuracy reached 76.8%, while with city blocks distance, accuracy reached 89.6%. Skeleton Hinge Features with Nearest Neighbor using KNN results are presented in Table 7.

Table 8. Skeleton Hinge Identification Accuracy using KNN on Firemaker DB

Method(k=1)	Distance	Accuracy
KNN	Chebychev	51.6%
KNN	Minkowski	76.8%
KNN	Cosine	76.8%
KNN	Correlation	76.8%
KNN	Hamming	1.2%
KNN	Seuclidean	0.4%
KNN	Cityblock	89.6%

6.2.1.5 Skeleton Hinge Features with Support Vector Machines results

Support vector machines (SVM) were used as well to identify the writer on Firemaker DB. A simple scheme of "one-vs-all" was used in an iterative process. In each iteration, a single document from the training set, consisting of 250 writers, was assigned to the class known and the rest to the class unknown. An SVM was trained using the skeleton hinge distribution combinations with fragment lengths 3,5,7,9 extracted from the training set and the class information assigned to them. Next, a new iteration was used to classify the documents in the test data set, after extracting the skeleton hinge distribution combinations with fragment lengths 3,5,7,9, according to the trained model. Accuracy was 53.6%

SVM with 'one-vs-one' scheme was also considered but trained only in the first 100 writers. In each iteration, a classifier was trained to distinguish between documents of 2 distinct writers. All the possible non-overlapping combinations were considered. A total of 4950 classifiers were trained. The SVM classifiers were trained by using the [3 5 7 9] skeleton hinge distribution combinations from the train set. Next, a new iteration was used to classify the documents in the test set. Every handwritten document was classified using the trained classifiers. Matching is achieved with a voting procedure. The most voted class is assigned to the document. Accuracy for 100 writers achieved 63%. Skeleton Hinge Features with Support Vector Machines results are presented in Table 8.

Table 9. Skeleton Hinge Features with Support Vector Machines results on Firemaker DB

Scheme	**Number of writers**	**Accuracy**
One-vs-all	250	53.6%
One-vs-one	100	63%

6.2.2 Codebook of Graphemes and Skeleton Hinge Distribution

In addition, an attempt was made to combine skeleton hinge distribution with a codebook of graphemes method on Firemaker DB. The results of this experiment are presented in Table II. The model-based methods [34, 26] reported accuracy of up to 97% on 150 writers, using a codebook of size 400 when the results were combined with edge-directional features. Unfortunately, it was impossible to train a codebook of 400 graphemes for 250 writers due to memory issues.

Instead, a codebook of 225 graphemes was trained for 250 writers. Maximum accuracy of 95,6% was reached. It is necessary to mention that the other methods reported 97 % accuracy on 150 writers with a codebook of 400 graphemes. In our case, an experiment was also performed using 150 writers of the data set and a codebook of 225 graphemes. An accuracy of 96% was achieved.

Codebook of Graphemes and Skeleton Hinge Distribution feature identification results are presented in Table 9.

Table 10. Skeleton Hinge Distribution Combined with Codebook of Graphemes Method Accuracy (Percentage) on Firemaker DB

Number of Writers	**CodeBook Size**	**Skeleton Hinge Distribution Combined with Codebook of Graphemes Method**		
		Manhattan Distance	***Euclidian Distance***	***Chi-square Distance***
250	225	95.6%	91.2%	78.8%
150	225	96%	94.7%	86.7%

6.2.3 Quantised Skeleton Hinge Distribution

In this section, experiments performed only for combining fragment lengths 5 and 9 since that combination achieved an accuracy of 90.8% on the Skeleton Hinge Distribution on Firemaker DB. The experiments performed on 250 writers using Manhattan Distance and quantised in 1,2,3,4,5,9 intensity levels. Quantised Skeleton Hinge Distribution feature identification results are presented in Table 10.

Table 11. Quantised Skeleton Hinge Distribution Accuracy (Percentage) on Firemaker DB

Number Of Quantizations	**Quantised Skeleton Hinge Distribution**
1	90.8%
2	92%
3	**92.4%**
4	89.6%
5	91.2%
9	88.4%

6.2.4 Weighted Skeleton Hinge

In this section, experiments were performed with combinations of various fragment length sizes (i.e. window sizes) where considered on pages 1 and 4 of the Firemaker DB, conducted on 250 writers using Manhattan Distance. For comparison with previous methods, we included the results of Edge Hinge Combinations (EHC), as reported in [26] and Skeleton Hinge Distribution (SHD) as reported in [97]. Weighted Skeleton Hinge Distribution (WSHD) features identification results are presented in Table 11.

Table 12. EHC, SHD, WSHD identification Accuracy (Percentage) with Manhattan Distance on Firemaker DB

Fragment Length Combinations	**EHC Accuracy (Percentage)**	**SHD Accuracy (Percentage)**	**WSHD Accuracy (Percentage)**
3	68%	80%	82%
5	70%	89.6%	88.8%
7	70%	90%	90%
9	69%	88%	88.8%
3 , 5	77%	85.2%	85.2%
3 , 7	77%	85.6%	85.6%
3 , 9	79%	86%	86%
5 , 7	74%	90%	89.6%
5 , 9	77%	**90.8%**	**91.2%**
7 , 9	72%	90%	89.6%
3 , 5 , 7	80%	86.8%	87.6%
3 , 7 , 9	78%	89.6%	89.2%
5 , 7 , 9	76%	90%	90.8%
3 , 5 , 7 , 9	**81%**	89.6%	88.4%

6.2.5 Run Length Directional Hinge

In this section, experiments were performed with combinations of various fragment length sizes (i.e. window sizes) were considered on pages 1 and 4 of the Firemaker DB, performed on 250 writers using Manhattan Distance. For comparison with previous methods, we included the results of Edge Hinge Combinations (EHC), as reported in [26] and Skeleton Hinge Distribution (SHD) as reported in [97]. Run Length Directional Hinge Distribution (RLDHD)features identification results are presented in Table 12

Table 13. EHC, SHD, RLDHD identification Accuracy (Percentage) with Manhattan Distance on Firemaker DB

Fragment Length Combinations	EHC Accuracy (Percentage)	SHD Accuracy (Percentage)	RLDHD Accuracy (Percentage)
3	68%	80%	85.2%
5	70%	89.6%	89.2%
7	70%	90%	90.4%
9	69%	88%	**91.2%**
3 , 5	77%	85.2%	88%
3 , 7	77%	85.6%	89.2%
3 , 9	79%	86%	89.2%
5 , 7	74%	90%	89.6%
5 , 9	77%	**90.8%**	89.2%
7 , 9	72%	90%	90.4%
3 , 5 , 7	80%	86.8%	88.8%
3 , 7 , 9	78%	89.6%	89.2%
5 , 7 , 9	76%	90%	90%
3 , 5 , 7 , 9	**81%**	89.6%	89.2%

6.2.6 Edge Skeleton Hinge Combination

In this section, experiments were performed with combinations of various fragment length sizes (i.e. window sizes) were considered on pages 1 and 4 of the Firemaker DB, conducted on 250 writers using Manhattan Distance. For comparison with previous methods, we included the results of Edge Hinge Combinations (EHC), as reported in [26] and Skeleton Hinge Distribution (SHD) as reported in [97]. Edge Skeleton Hinge Combination (ESHC) features identification results are presented in Table 13

Table 14. EHC, SHD, ESHC identification Accuracy (Percentage) with Manhattan Distance on Firemaker DB

Fragment Length Combinations	EHC Accuracy (Percentage)	SHD Accuracy (Percentage)	ESHC Accuracy (Percentage)
3	68%	80%	79.4%
5	70%	89.6%	89.2%
7	70%	90%	89.6%
9	69%	88%	87.2%
3 , 5	77%	85.2%	84.6%
3 , 7	77%	85.6%	85.2%
3 , 9	79%	86%	85.6%

5 , 7	74%	90%	89.6%
5 , 9	77%	**90.8%**	**90.2%**
7 , 9	72%	90%	89.4%
3 , 5 , 7	80%	86.8%	86.8%
3 , 7 , 9	78%	89.6%	89.2%
5 , 7 , 9	76%	90%	89.4%
3 , 5 , 7 , 9	**81%**	89.6%	89.2%

6.2.7 Directional Features Comparison

In this section, a comparison of top identification results reported in the literature on Firemaker DB, and in the sections above for the directional features of Edge Direction Distribution, Edge Hinge Combinations, Skeleton Hinge Distribution, Quantized Skeleton Hinge Distribution, Weighted Skeleton Hinge Distribution, Run Length Directional Hinge Distribution, Quill-Hinge and Junctions are presented in Table 14. Furthermore, a graphical representation of the results on Edge Hinge Combinations, Skeleton Hinge Distribution, Weighted Skeleton Hinge Distribution and Run Length Directional Hinge Distribution on common fragment length combinations is given in Fig. 75

Table 15. EDD, EHC, SHD, QSHD, WSHD, RLDHD, ESHC , and methods from literature identification Accuracy (Percentage) with Manhattan Distance on Firemaker DB

Method	**Accuracy Reported**
EDD	35%
EHC	68%
SHD	90.8%
QSHD	92.4%
WSHD	91.2%
ESHC	90.2%
RLDHD	91.2%
Edge-Hinge [25]	63%
Codebook of Graphemes combined with Edge-Hinge [34]	97%
Edge-Hinge combinations [26]	81%
Codebook of Graphemes combined with Edge-Hinge Combinations [26]	97%
Contour-Hinge combined with Writer-Specific Grapheme Emission PDF [110]	83%
SDS+SOH [111]	92.4%

Quill-Hinge [27]	86%
Junclets [112]	80.6%
Junclets+Hinge [112]	89.8%
BW-LBC [51]	94.4%
CLGP [52]	97.6%
Dissimilarity GMM (DGMM) [54]	97.98%
GR-RNN [21]	98.8%

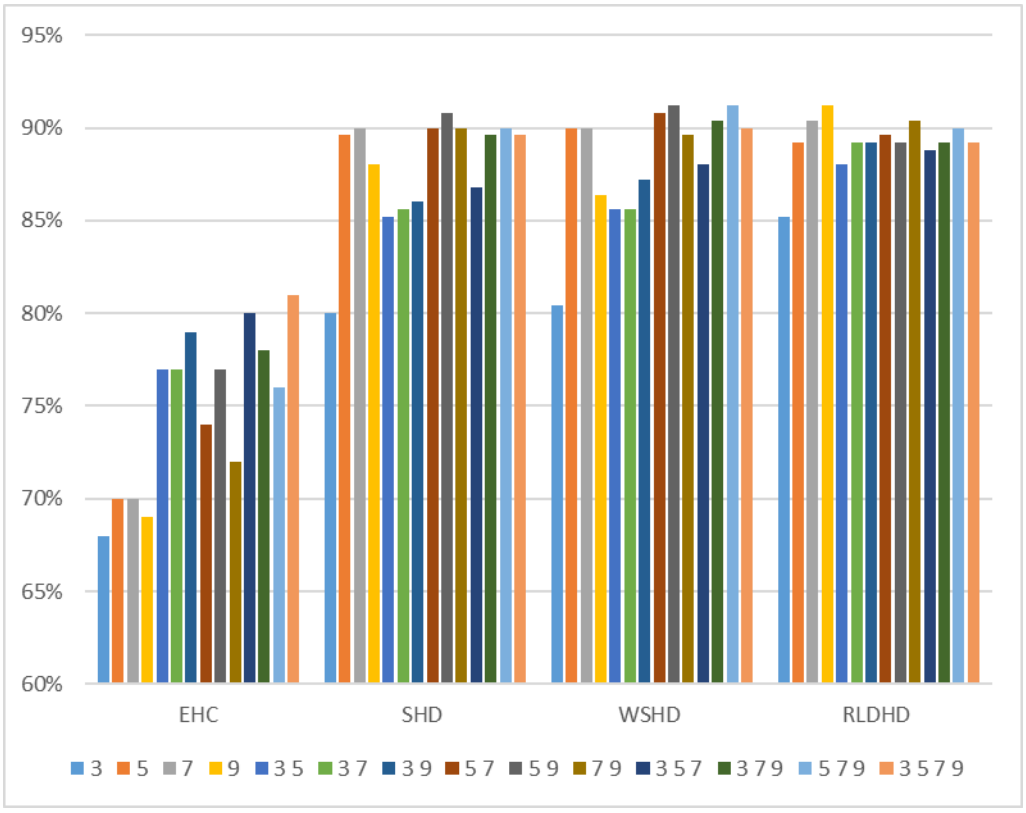

Figure 75. EHC, SHD, WSHD, RLDHD identification accuracy on Firemaker DB

6.2.8 Filtering with Text Localisation method

In this section, experiments were performed with combinations of various fragment length sizes (i.e. window sizes) that were considered on the filtered pages 1 and 4 of the dataset, performed on 250 writers using Manhattan Distance. To Filter the pages, the Text Localization technique was used, as described in Chapter 3.2. This method is appropriate to localise only pure text using some rules and dismiss all the noise produced by the writer. For comparison with previous methods, we included the results of Skeleton Hinge Distribution (SHD) as reported in [97] and Edge Skeleton Hinge Combination (ESHC) as presented previously. For consistency, the same methods were used for the filtered pages. Identification results are presented in Table 15

Table 16. SHD, filtered SHD, ESHC and filtered ESHC identification Accuracy (Percentage) with Manhattan Distance

Fragment Length Combinations	**SHD Accuracy (Percentage)**	**Filtered SHD Accuracy (Percentage)**	**ESHC Accuracy (Percentage)**	**Filtered ESHC Accuracy (Percentage)**
3	80%	80%	79.4%	83.2%
5	89.6%	88.8%	89.2%	90%
7	90%	90%	89.6%	90.4%
9	88%	88%	87.2%	89.2%
3 , 5	85.2%	85.2%	84.6%	86%
3 , 7	85.6%	87.2%	85.2%	88%
3 , 9	86%	86%	85.6%	88.8%
5 , 7	90%	90.4%	89.6%	90%
5 , 9	**90.8%**	**91.2%**	**90.2%**	90.4%
7 , 9	90%	90.8%	89.4%	**90.8%**
3 , 5 , 7	86.8%	86.8%	86.8%	87.6%
3 , 7 , 9	89.6%	89.2%	89.2%	90%
5 , 7 , 9	90%	90.4%	89.4%	90.4%
3 , 5 , 7 , 9	89.6%	89.6%	89.2%	89.6%

Chapter 7

7. Conclusion & Future Work

In this work, several features for writer identification were presented. Our experiments indicate that even by using a single feature, writer identification accuracy yields promising results. While most of our experiments that achieved top accuracy are performed using nearest neighbour matching, machine learning methods performance was also beyond our expectations. Machine learning algorithms achieve better performance according to the size of the train set. Furthermore, machine learning algorithms which use few train examples. i.e. one sample per writer usually do not perform well. In the work presented here, the train set for each writer had only one train example.

By revisiting our assumption that all stroke widths, i.e. line thickness, should be considered the same size. By applying skeletonisation to characters, this criterion is met. All stroke widths are transformed into a single pixel line. The experimental results proved that the previous assumption is correct. Furthermore, on our technique for run Length Directional Hinge, and by using a one-pixel width fragment also proved that it is possible to achieve even higher accuracy without any loss of information.

Moreover, with the Weighted Skeleton Hinge Distributions, it is hard to prove our second assumption that main body size can affect writer identification since the accuracy has only increased by 0.4% and thus it is not a significant improvement although even this slight improvement leaves space for future improvements on this technique.

We strongly believe that further improvements can be achieved. A combination of features along with the skeleton hinge distribution like different statistical features can be used. Further research is needed in the area.

Furthermore, the methods described in this work can be used for a variety of different applications. Some suggestions of possible future applications are presented here.

First of all, the techniques presented here could be used to detect different handwritings in a single text document. This feature can be very useful, especially in student exams. It has been observed that during exams, some students write notes or exact solutions on other student's papers, in order to cheat. Our proposed techniques can help professors identify the ones that cheat. Also some of the times some other student might write the entire exam for their friends. With this system, all students handwritings can be verified.

Skeleton hinge distribution features, can also be used as a writer verification system, by applying a threshold. If the distance between the two samples is lower than a predefined threshold, then verification of the writer can be made.

In the same way, it can be used as a user authentication method or as addition to two factor authentication for mobile phones. Online features can be added as well to improve results.

Furthermore, it can be used for historical documents of unknown origin. There exist numerous documents, that their origin till today is unknown. With our proposed methods, a match of these documents and their writers can be made, which might give a better perspective on history.

Lastly, the skeleton hinge distribution feature suggested in this work might be a fit for other applications. It is believed that it can be advantageous in slant correction systems. It might also have applications in word spotting systems as well, but further research is needed to determine that.

References

[1] A. K. Jain, R. Bolle, S. Pankanti and eds., "Biometrics. Personal Identification in Networked Society," *Kluwer Academic,* 1999.

[2] A. K. Jain, R. Bolle, S. Pankanti and eds., "Biometrics identification," *Comm. ACM 43(2),* p. pp. 91 – 98, 2000.

[3] J. Bigun, F. Smeraldi and Eds., "Audio- and video-based biometric person authentication.".*Proc. of the 3rd Int. Conf. AVBPA, Halmstadt, Sweden. 2001*..

[4] R. Plamondon and G. Lorette, "Automatic signature verification and writer identification".*the state of the art. Pattern Recognition, 22, pp. 107–131. 1989*..

[5] A. Bensefia, A. Nosary, T. Paquet and L. Heutte, "Writer identification by writer's invariants.".*Frontiers in Handwriting Recognition, 2002. Proceedings. Eighth International Workshop on. IEEE.*

[6] A. Séropian, "Analyse de document et identification de scripteurs.," *Doctorat de l'université de Toulon et du Var, France, 18.,* 2003.

[7] M. Philipp, "Fakten zu fish, das forensische informations-system handschriften des bundeskriminalamtes-eine analyse nach über 5 jahren wirkbetrieb. [technical report], Technical report, Kriminaltechnisches Institut 53, Bundeskriminalamt, Wiesbaden," *Technical report, Kriminaltechnisches Institut 53, Bundeskriminalamt, Thaerstrasse 11, 65173 Wiesbaden, Germany, 1996. in German.,* 1996.

[8] L. Schomaker, "Writer identification and verification.," *In Advances in Biometrics (pp. 247-264). Springer, London.,* 2008.

[9] A. Nosary, L. Heutte, T. Paquet and Y. Lecourtier, "Defining writer's invariants to adapt the recognition task.," *Document Analysis and Recognition, 1999. ICDAR'99. Proceedings of the Fifth International Conference on. IEEE,* 1999.

[10] R. Plamondon and G. Lorette, "Automatic signature verification and writer identification - the state of the art," *Pattern Recognition 22(2), 107–131.,* 1989.

[11] H. E. Said, T. N. Tan and K. D. Baker, "Personal identification based on handwriting.," *Pattern Recognition, 33(1), 149-160.,* 2000.

[12] U. V. Marti, R. Messerli and H. Bunke, "Writer identification using text line based features.".*In Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on (pp. 101-105). IEEE.*.

[13] L. Schomaker and L. Vuurpijl, "Forensic writer identification: A benchmark data set and a comparison of two systems," *[internal report for the Netherlands Forensic Institute]. Technical report, Nijmegen: NICI,* 2000.

[14] S. Fiel, F. Kleber, M. Diem, V. Christlein, G. Louloudis, S. Nikos and B. Gatos, "Icdar2017 competition on historical document writer identification (historical-wi).," *In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR),* pp. 1377-1382.

[15] L. Schomaker, "Advances in Writer Identification and Verification," *Ninth International Conference on Document Analysis and Recognition,* p. 1268–1273*,* 2007.

[16] M. Popović, M. A. Dhali and L. Schomaker, "Artificial intelligence based writer identification generates new evidence for the unknown scribes of the Dead Sea Scrolls exemplified by the Great Isaiah Scroll (1QIsaa)," *PloS one 16.4 e0249769,* 2021.

[17] S. Fiel and R. Sablatnig, "Writer identification and retrieval using a convolutional neural network," *International Conference on Computer Analysis of Images and Patterns Springer, Cham,* 2015.

[18] A. Rehman, S. Naz and M. I. Razzak, "Writer identification using machine learning approaches: a comprehensive review," *Multimedia Tools and Applications 78.8,* pp. 10889-10931, 2019.

[19] S. He and L. Schomaker, "Deep adaptive learning for writer identification based on single handwritten word images," *Pattern Recognition, 88,* pp. 64-74, 2019.

[20] S. He and L. Schomaker, "Fragnet: Writer identification using deep fragment networks," *IEEE Transactions on Information Forensics and Security, 15,* pp. 3013-3022, 2020.

[21] S. He and L. Schomaker, "GR-RNN: Global-Context Residual Recurrent Neural Networks for Writer Identification," *Pattern Recognition, 107975,* 2021.

[22] Y. Tang and X. Wu, "Text-independent writer identification via CNN features and joint Bayesian," *15th International Conference on Frontiers in Handwriting Recognition (ICFHR). IEEE,* 2016.

[23] E. Commission, "Europe fit for the Digital Age: Artificial Intelligence," [Online]. Available: https://ec.europa.eu/commission/presscorner/detail/en/ip_21_1682.

[24] L. Schomaker, "Dilemmas in the application of artificial- intelligence methods in digital paleography," [Online]. Available: https://www.youtube.com/watch?v=chVdYOBnOuw.

[25] M. Bulacu, L. Schomaker and L. Vuurpijl, "Writer identification using edge-based directional features," *IEEE,* 2003.

[26] L. Van Der Maaten and E. O. Postma, "Improving automatic writer identification.," *BNAIC,* 2005.

[27] A. A. Brink, J. Smit, M. L. Bulacu and L. R. B. Schomaker, "Writer identification using directional ink-trace width measurements," *Pattern Recognition 45.1,* pp. 162-171, 2012.

[28] S. He and L. Schomaker, "Co-occurrence features for writer identification.," *In 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR),* pp. 78-83.

[29] P. Diamantatos, E. Kavallieratou and S. Gritzalis, "Skeleton Hinge Distribution for Writer Identification.," *International Journal on Artificial Intelligence Tools,* 2016.

[30] P. Diamantatos, V. Verras and E. Kavallieratou, "Detecting Main Body Size in Document Images.," *Document Analysis and Recognition (ICDAR), 12th International Conference on. IEEE,* 2013.

[31] E. N. Zois and V. Anastassopoulos, "Morphological Waveform Coding for Writer Identification," *Pattern Recognition, vol. 33, no. 3, pp. 385-398.,* 2000.

[32] S. N. Srihari, M. J. Beal, K. Bandi, V. Shah and P. Krishnamurthy, "A Statistical Model for Writer Verification," *Proc. Eighth Int'l Conf. Document Analysis and Recognition (ICDAR), pp. 1105-1109.,* 2005.

[33] A. Bensefia, T. Paquet and L. Heutte, "Handwritten Document Analysis for Automatic Writer Recognition," *Electronic Letters on Computer Vision and Image Analysis, vol. 5, no. 2,,* pp. pp. 72-86, 2005.

[34] L. Schomaker, M. Bulacu and K. Franke, "Automatic writer identification using fragmented connected-component contours.," *In Proceedings of the 9 th IWFHR, pages 185–190, Tokyo, Japan,* 2004.

[35] A. Schlapbach and H. Bunke, "A writer identification and verification system using HMM based recognizers," *Pattern Analysis Application (Springer)10,33–43, doi:10.1007/s10044-006-0047-5.*, 2007.

[36] V. Pervouchine and G. Leedham, "Extraction and analysis of forensic document examiner features used for writer identification," *Pattern Recognition Journal 40, 1004–1013.*, 2007.

[37] I. Bar-Yosef, I. Beckman, K. Kedem and I. Dinstein, "Binarization, character extraction, and writer identification of historical Hebrew calligraphy documents," *International Journal on Document Analysis and Recognition, vol. 9, no. 2,* pp. pp. 89-99, 2007.

[38] B. Li, Z. Sun and T. Tan, "Hierarchical Shape Primitive Features for Online Text-independent Writer Identification," *Proc. of 2th ICB, pages 201–210.*, 2007.

[39] Z. He, X. You and Y. Y. Tang, "Writer identification of Chinese handwriting documents using hidden Markov tree model," *Pattern Recognition Journal 41, 1295–13072008-06-15.*, 2008.

[40] Y. Yan, Q. Chen, W. Deng and F. Yuan, "Chinese Handwriting Identification Based on Stable Spectral Feature of Texture Images," *International Journal of Intelligent Engineering and Systems, Vol.2, No.1.*, 2009.

[41] M. Bulacu and L. Schomaker, "Text-independent writer identification and verification using textural and allographic features," *IEEE Transactions on Pattern Analysis and Machine Intelligence(PAMI) 29(4) Special Issue—Biometrics: Progress and Directions.*, p. 701–717, 2007.

[42] A. Al-Dmour and R. Zitar, "Arabic writer identification based on hybrid spectral- statistical measures," *Journal of Experimental & Theoretical Artificial Intelligence 19(4),* p. 307–332, 2007.

[43] X. Wu, Y. Tang and W. Bu, "Offline text-independent writer identification based on scale invariant feature transform.," *IEEE Transactions on Information Forensics and Security, 9(3),* pp. 526-536.

[44] A. Nicolaou, A. D. Bagdanov, M. Liwicki and D. Karatzas, "Sparse radial sampling LBP for writer identification.," *In 2015 13th International Conference on Document Analysis and Recognition (ICDAR),* pp. 716-720.

[45] H. Mohammed, V. Mäergner, T. Konidaris and H. S. Stiehl, "Normalised Local Naïve Bayes Nearest-Neighbour Classifier for Offline Writer Identification.," *In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR),* pp. 1013-1018.

[46] A. J. Newell and L. D. Griffin, "Writer identification using oriented basic image features and the delta encoding," *Pattern Recognition, 47(6),* 2014.

[47] G. Abdeljalil, C. Djeddi, I. Siddiqi and S. Al-Maadeed, "Writer Identification on Historical Documents using Oriented Basic Image Features.," *In 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR),* pp. 369-373.

[48] F. Nadia and H. Kamel, "Personal Identification based on Texture Analysis of Arabic Handwriting Text," *In: IEEE International Conference on Information and Communications Technologies (ICTTA'06), vol. (1),* pp. 1302-1307, 2007.

[49] S. Gazzah and N. B. Amara, "Arabic Handwriting Texture Analysis for Writer Identification using the DWT-lifting Scheme.," *In: 9th International Conference on Document Analysis and Recognition (ICDAR'07), vol. (2),* pp. 1133-1137, 2007.

[50] S. Al-Ma'adeed, E. Mohammed, D. Al Kassis and F. Al-Muslih, "Writer identification using edge-based directional probability distribution features for Arabic words," *in: IEEE/ACS International Conference on Computer Systems and Applications (AICCSA),* p. 582—590, 2008.

[51] A. Chahi, Y. Ruichek and R. Touahni, "Block wise local binary count for off-line text-independent writer identification," *Expert Systems with Applications,* pp. 1-14, 2018.

[52] A. Chahi, Y. Ruichek and R. Touahni, "Cross multi-scale locally encoded gradient patterns for off-line text-independent writer identification," *Engineering Applications of Artificial Intelligence 89, 103459,* 2020.

[53] L. Xing and Y. Qiao, "Deepwriter: A multi-stream deep CNN for text-independent writer identification," *15th international conference on frontiers in handwriting recognition (ICFHR). IEEE,* 2016.

[54] F. A. Khan, F. Khelifi, M. A. Tahir and A. Bouridane, "Dissimilarity Gaussian Mixture Models for Efficient Offline Handwritten Text-Independent Identification Using SIFT and RootSIFT Descriptors," *IEEE Transactions on Information Forensics and Security,* pp. vol. 14, no. 2, pp. 289-303, Feb. 2019.

[55] M. Bulacu, L. Schomaker and L. Vuurpijl, "Writer identification using edge-based directional features," *IEEE,* 2003.

[56] A. A. Brink, J. Smit, M. L. Bulacu and L. R. B. Schomaker, "Writer identification using directional ink-trace width measurements.," *Pattern Recognition 45.1,* pp. 162-171, 2012.

[57] S. He, M. Wiering and L. Schomaker, "Junction detection in handwritten documents and its application to writer identification.," *Pattern Recognition 48.12 (2015): 4036-4048.*, 2015.

[58] J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," *Pattern Recognition 33.2 (2000): 225-236.*, 2000.

[59] P. Stathis, E. Kavallieratou and N. Papamarkos, "An evaluation technique for binarization algorithms," *J. UCS, vol. 14, no. 18, pp. 3011–3030*, 2008.

[60] A. Mishra, K. Alahari and C. V. Jawahar, "An MRF model for binarization of natural scene text," *International Conference on Document Analysis and Recognition*, p. pp. 11–16, 2011.

[61] T. Wakahara and K. Kita, "Binarization of color character strings in scene images using k-means clustering and support vector machines".*in ICDAR, 2011, pp. 274–278.*.

[62] Y. F. Pan, X. Hou and C. L. Liu, "Text localization in natural scene images based on conditional random field".*in ICDAR, 2009, pp. 6–10.*

[63] L. Gomez and D. Karatzas, "Multi-script text extraction from natural scenes," *Proceedings of the 12th Int. Conf. on Document Analysis and Recognition (ICDAR 2013). Vol. 1.*, 2013.

[64] J. Canny, "A computational approach to edge detection.," *Pattern Analysis and Machine Intelligence, IEEE Transactions on 6*, pp. 679-698, 1986.

[65] R. Mehrotra, K. R. Namuduri and N. Ranganathan, "Gabor filter-based edge detection.".*Pattern Recognition 25.12 (1992): 1479-1494.*.

[66] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns.".*Communications of the ACM 27.3 (1984): 236-239.*.

[67] L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms.".*Image Processing, IEEE Transactions on 2.2 (1993): 176-201.*.

[68] F. Chang, C. J. Chen and C. J. Lu, "A linear-time component-labeling algorithm using contour tracing technique.," *Computer Vision and Image Understanding 93.2*, pp. 206-220, 2004.

[69] U. V. Marti and H. Bunke, "Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system," *Int. Journal of Pattern Recognition and Artifical Intelligence, 15(1), pp. 65–90*, 2001.

[70] M. Côté, E. Lecolinet, M. Cheriet and C. Y. Suen, "Automatic reading of cursive scripts using a reading model and perceptual concepts.," *IJDAR, vol 1, pp. 3-17,* 1998.

[71] C. K. Cheng and M. Blumenstein, "The neural-based segmentation of cursive words using enhanced heuristics," *Proc. of the 8th International Conference on Document Analysis and Recognition,* pp. pp.650-654, 2005.

[72] H. Lee and B. Verma, "A novel multiple experts and fusion based segmentation algorithm for cursive handwriting recognition," *Proc. of the International Joint Conference on Neural Networks, pp.2994-2999,* 2008.

[73] A. Vinciarelli and J. Luettin, "A new normalization technique for cursive handwritten words," *Pattern Recognition Lett. 22 (9),* p. 1043–1050, 2001.

[74] B. Gatos, I. Pratikakis and K. Ntirogiannis, "Segmentation based recovery of arbitrarily warped document images.," *In Proc. Int. Conf. on Document Analysis and Recognition, Curitiba, Brazil, Sep.,* 2007.

[75] D. V. Sharma and S. Wadhwa, "Dewarping Machine Printed Documents of Gurmukhi Script," *Information Systems for Indian Languages, Communications in Computer and Information Science series , v.139,* pp. pp. 117-123, 2011.

[76] N. Doulgeri and E. Kavallieratou, "Retrieval of Historical Documents by Word Spotting," *IS&T/SPIE Electronic Imaging,* 2009.

[77] T. Adamek, N. E. O'Connor and A. F. Smeaton, "Word matching using single closed contours for indexing handwritten historical documents," *Int. J. Doc. Anal.Recognit. 9 (2) (2007) 153–165.,* 2007.

[78] J. A. Rodríguez-Serrano and F. Perronnin, "Handwritten word-spotting using hidden Markov models and universal vocabularies," *Pattern Recognition, vol. 42, no. 9, pp. 2106–2116,* 2009.

[79] E. Kavallieratou, N. Dromazou, N. Fakotakis and G. Kokkinakis, "An Integrated System for Handwritten Document Image Processing," *International Journal of Pattern Recognition and Artificial Intelligence, Vol. 17, No. 4,* pp. pp. 101-120, 2003.

[80] N. Vasilopoulos and E. Kavallieratou, "A classification-free word-spotting system," *Proc. SPIE 8658, Document Recognition and Retrieval XX,* 2013.

[81] J. S. Lim, "Two-dimensional signal and image processing," *Englewood Cliffs, NJ, Prentice Hall, 1990, 710 p. 1,* 1990.

[82] Y. Zhong, K. Karu and A. K. Jain, "Locating text in complex color images," *Pattern Recognition, vol. 28, no. 10, pp. 1523–1535, Oct.,* 1995.

[83] C. Yao, X. Bai, W. Liu, Y. Ma and Z. Tu, "Detecting Texts of Arbitary Orientations in Natural Images," *CVPR,* 2012.

[84] V. C. Dinh, S. S. Chun, S. Cha, H. Ryu and S. Sull, "An Efficient Method for Text Detection in Video Based on Stroke Width Similarity," *ACCV,* 2007.

[85] A. Mishra, K. Alahari and C. V. Jawahar, "Top-down and bottom-up cues for scene text recognition,," *in CVPR,* p. pp. 2687–2694, 2012.

[86] B. Epshtein, E. Ofek and Y. Wexler, "Detecting text in natural scenes with stroke width transform," *CVPR,* 2010.

[87] J. Zhang and R. Kasturi, "Character energy and link energy-based text extraction in scene images," *ACCV,* 2010.

[88] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, R. Young and H. Miyao, "ICDAR 2003 robust reading competitions: Entries, results, and future directions," *Int. Jour. on Document Analysis and Recognition, vol. 7, no. 2-3, pp. 105–122, Jul,* 2005.

[89] S. M. Lucas, "ICDAR 2005 text locating competition results".*in Proc. Int. Conf. on Document Analysis and Recognition, Seoul, Korea, Aug. 2005, pp. 80–84.*

[90] A. Shahab, F. Shafait and A. Dengel, "ICDAR 2011 robust reading competition challenge 2 : Reading text in scene images," *In 2011 international conference on document analysis and recognition,* pp. pp. 1491-1496, 2011.

[91] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i Bigorda, S. R. Mestre and L. P. De Las Heras, "ICDAR 2013 Robust Reading Competition.".

[92] P. Diamantatos, E. Kavallieratou and P. Gomez-Gil, "Binarization: a Tool for Text Localization.," *In 2014 14th International Conference on Frontiers in Handwriting Recognition,* pp. 649-654.

[93] "http://www.unipen.org/trigraphslant.html".

[94] CCIR, "Recommendation. 601-2: Encoding parameters of digital television for studio.," *International Consultative Committee for Radio,* 1990.

[95] A. Shahab, F. Shafait and A. Dengel, "ICDAR 2011 robust reading competition challenge 2: Reading text in scene images.," *In 2011 international conference on document analysis and recognition,* pp. 1491-1496.

[96] S. M. Awaida and S. A. Mahmoud, "State of the art in off-line writer identification of handwritten text and survey of writer identification of Arabic text," *Educational Research and Reviews, 7(20),* pp. 445-463, 2012.

[97] P. Diamantatos, E. Kavallieratou and S. Gritzalis, "Skeleton Hinge Distribution for Writer Identification.," *International Journal on Artificial Intelligence Tools,* 2016.

[98] "http://homepage.tudelft.nl/19j49/Software.html".

[99] R. C. Gonzalez, R. E. Woods and S. L. Eddins, "Digital image processing using Matlab.," *Princeton Hall Pearson Education Inc., New Jersey,* 2004.

[100] T. Kohonen., "Self-organization and associative memory: 3rd edition.," *Springer-Verlag New York,Inc., New York, NY, USA,* 1989.

[101] 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),* pp. 449-457, 2011.

[102] J. P. Crettez, "A set of handwriting families: style recognition," *In Proceedings of 3rd International Conference on Document Analysis and Recognition (Vol. 1, pp. 489-494). IEEE.,* 1995.

[103] F. J. Maarse, The study of handwriting movement: Peripheral models and signal processing techniques., Lisse [etc.]: Swets & Zeitlinger., 1987.

[104] R. A. Huber and A. M. Headrick, Handwriting identification: facts and fundamentals, CRC press, 1999.

[105] H. J. J. Hardy and W. Fagel, "Methodological aspects of handwriting identification.," *Journal of Forensic Document Examination, 28,* pp. 125-147, 2018.

[106] K. M. Koppenhaver, Forensic document examination: principles and practice., 2007.

[107] E. Kavallieratou, L.-S. L. and N. Vasilopoulos, "Slant removal technique for historical document images.," *Journal of Imaging, 4(6),* p. 80, 2018.

[108] E. Kavallieratou, N. Fakotakis and G. Kokkinakis, "Slant estimation algorithm for OCR systems," *Pattern Recognition, 34(12),* pp. 2515-2522, 2001.

[109] L. R. Schomaker, A. J. Thomassen and H. L. Teulings, "A computational model of cursive handwriting," *In Computer recognition and human production of handwriting ,* pp. 153-177, 1989.

[110] M. Bulacu and L. Schomaker, "Text-independent writer identification and verification using textural and allographic features," *IEEE Transactions on Pattern Analysis and Machine Intelligence(PAMI) 29(4) Special Issue—Biometrics: Progress and Directions.*, p. 701–717, 2007.

[111] X. Wu, Y. Tang and W. Bu, "Offline text-independent writer identification based on scale invariant feature transform," *IEEE Transactions on Information Forensics and Security, 9(3),* pp. 526-536.

[112] S. He, M. Wiering and L. Schomaker, "Junction detection in handwritten documents and its application to writer identification," *Pattern Recognition 48.12 (2015): 4036-4048.*, 2015.

[113] M. L. Bulacu, "Statistical pattern recognition for automatic writer identification and verification".*2007.*

[114] N. Otsu, "A threshold selection method from gray-level histograms.," *Automatica 11.285-296 (1975): 23-27.*, 1975.

[115] H. E. Said, T. N. Tan and K. D. Baker, "Personal Identification Based on Handwriting," *Pattern Recognition, vol. 33, no. 1, pp. 149-160,* 2000.

[116] S. Srihari, C. Huang and H. Srinivasan, "On the discriminability of the handwriting of twins".*Journal of Forensic Sciences 53.2 (2008): 430-446.*

[117] C. Vielhauer, "Biometric user authentication for IT security: from fundamentals to handwriting.," *Vol. 18.Springer*, 2005.