الجمـهورية الجزائرية الديمقراطية الشعبيـة
وزارة التعليم العالي والبحث العلمي
جـامعـة سعـيـدة د. مـولاي الطـاهـر
كلية الرياضيات و الإعلام الآلي و الاتصالات السلكية و اللاسلكية
قسم: الإعلام الآلي

Mémoire de Master en informatique

Spécialité : Intelligence Artificielle – Princes et Applications

T h è m e

Analyse des documents manuscrits : Caractérisation de l'individualisation du scripteur

▪ **Présenté par :**

Wail Nadjib BOUMAZA

Aicha Ahlem MEGHERBI

▪ **Dirigé par :**

- Dr. Tayeb BAHRAM

Année universitaire 2025-2026

Contents

List of Figures

List of Tables

Chapter 1

General Introduction

1.1 Introduction

This thesis addresses the problem of automatic person recognition using scanned images of handwriting. Identifying (verifying) the author of a handwritten sample using automatic image-based methods is an interesting pattern recognition problem with direct applicability in the forensic and historic document analysis fields. Approaching this challenging problem raises a number of important research themes in computer vision:

- How can individual handwriting style be characterized using computer algorithms?
- What representations or features are most appropriate and how can they be combined?
- What performance can be achieved using automatic methods?

The current study describes a new and very effective technique that we have developed for automatic writer identification and verification (i.e., recognition). The goal of our research was to design state-of-the-art automatic methods involving only a reduced number of adjustable parameters and to create a robust writer recognition system capable of managing hundreds to thousands of writers. There are two distinguishing characteristics of our approach: human intervention is minimized in the writer identification process and we encode individual handwriting style using features designed to be independent of the textual content of the handwritten sample. Writer individuality is encoded using probability distribution functions extracted from handwritten text blocks and, in our methods, the computer is completely unaware of what has been written in the samples. The development of our writer recognition techniques takes place at a time when many biometric modalities undergo a transition from research to real full-scale deployment. Our methods also have practical feasibility and hold the promise of concrete applicability.

This thesis introduces a writer recognition technique that could have a significant impact on the field of forensic science (see Fig. 1.1). Our method has been statistically evaluated using multilingual benchmarking datasets: the ICDAR2011 (208 documents from 26 writers), the ICDAR2013 (1,000 documents from 250 writers), and the CVL (1,550 documents from 310 writers) datasets.

1.2 Motivation and Context

Biometric systems have become an integral part of modern security and identification applications. Traditional authentication methods, such as passwords and identification cards, are vulnerable to risks like theft, duplication, and unauthorized access. Consequently, biometric technologies have emerged as reliable alternatives that provide enhanced security, user convenience, and fraud prevention. Handwriting occupies a unique position among biometric modalities because it reflects an individual's physiological and behavioral characteristics. Influenced by hand movement, writing habits, muscular coordination, and cognitive behavior, handwriting is a highly individualized trait. Offline automatic writer identification and verification systems analyze scanned images of handwritten documents. These systems do not require dynamic information, such as pen pressure or writing speed. Instead, they rely solely on the visual appearance and structural characteristics of handwriting. Advancements in image processing techniques and the increased accessibility of digital handwritten documents have greatly contributed to the development of handwriting recognition systems. However, the variability and complexity of handwriting present significant challenges for researchers.

The field of writer recognition in monolingual contexts has advanced considerably, with many contributions addressing major world scripts. However, the field of writer identification and verification in multiscript settings remains largely unexplored. Few existing studies address how features behave across languages or how to maintain performance when writers use different scripts. Furthermore, most systems require a large writing sample, which is often unavailable in real-world applications such as forensic document analysis and historical manuscript indexing. This research addresses these challenges by developing a robust, scalable writer recognition system that can handle the following:

- ◇ Handwritten documents in multiple languages and scripts;
- ◇ A text-independent analysis is performed when the content of the writing is either unknown or irrelevant;
- ◇ Short and variable-length writing samples;
- ◇ High inter-writer similarity and intra-writer variability.

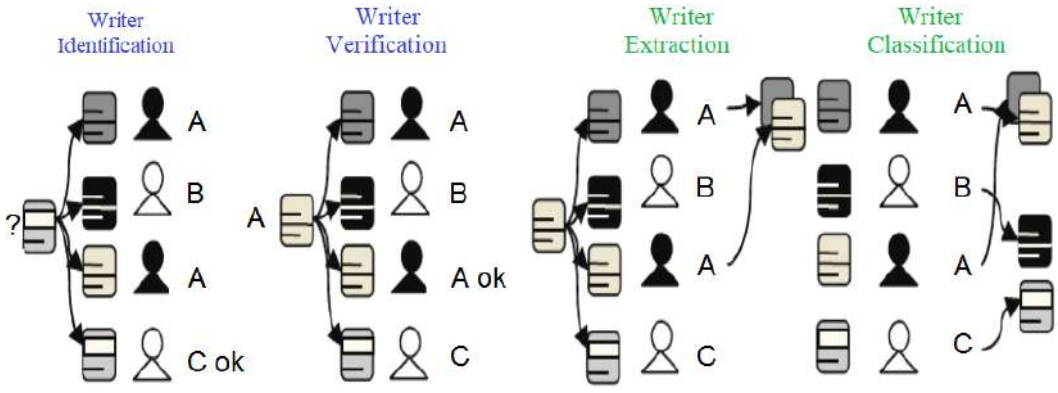


Figure 1.1: Illustration of different classification and recognition scenarios of writers.[3].

This work was also motivated by the limitations of current writer recognition methods. Texture- and codebook-based methods often fail to capture the subtle geometric nuances of handwriting. Meanwhile, deep learning approaches require large annotated datasets and significant computational resources. Although contour-based methods provide more interpretable and structurally meaningful representations, their effectiveness depends heavily on the quality of preprocessing and the reliability of contour extraction. Therefore, developing robust contour extraction and feature representation techniques is an important research challenge.

1.3 Objectives of the Research

The primary aim of this dissertation is to advance the research of offline writer identification and verification using multi-script handwritten texts within a fully text-independent framework. The proposed technique aims to demonstrate high accuracy and reliability in writer recognition across French, German, English and Greek scripts by achieving high IR and low EER values. The specific objectives of this project are as follows:

- ◇ Propose handwriting features that are both discriminative and descriptive. Draw inspiration from the features used by forensic experts and adapt them for multilingual contexts;
- ◇ Improve writer identification and verification performance by designing and evaluating complex feature representations that include combinations of texture descriptors, shape metrics, and stroke dynamics;
- ◇ Build a complete offline writer recognition system that encompasses design, development, unit testing, and integration;
- ◇ Use multilingual benchmark datasets to evaluate the system. If possible, participate in international competitions such as ICDAR or ICFHR;
- ◇ This could lead to further research on handwriting biometrics and multilingual text analysis, which could result in a doctoral thesis.

1.4 Dissertation Organization

This document is organized into the following six chapters:

- ▷ **Chapter 1 (General Introduction)** : This section (Section 1) introduces the fundamental concepts of biometrics and handwriting analysis. These include sources of variability among and within writers. It also reviews existing writer recognition systems.
- ▷ **Chapter 2 (Individual Characterization)** : Section 2 surveys the main research contributions to offline writer identification verification, covering handcrafted and learning-based approaches. It also places our work in the context of existing literature.
- ▷ **Chapter 3 (Writer Recognition: State-of-the-Art)** : Section 3 elaborates on the three main steps of the proposed system: preprocessing, feature extraction/combination, and classification (recognition or identification and verification).

- ▷ **Chapter 4 (Proposed Approach)** : Section 4 elaborates on the three main steps of the proposed system: preprocessing, feature extraction/combination, and classification (recognition).

- ▷ **Chapter 5 (Results and analysis)** : Section **??** outlines the three well-known handwritten databases utilized in this research. It offers information regarding enrollment and evaluation protocols and contrasts the outcomes of the proposed method with those of the leading techniques for writer identification and verification.

- ▷ **Chapter 6 (Conclusion)** : Section **??** concludes the thesis and outlines ideas and recommendations for future research.

1.5 Methodology and Technologies

These objectives will be achieved using the following methodology and tools:

- ⋆ **Programming Languages:** C++ and C# are used for system development and performance optimization.

- ⋆ **Platforms:** Cross-platform compatibility with Windows and Linux operating systems.

- ⋆ **Development tools:** Microsoft Visual Studio, CPPUnit for unit testing, and Test-Driven.net for integration tests.

- ⋆ **Modelling and design:** StarUML for system architecture modelling and the application of software engineering best practices using Design Patterns.

- ⋆ **Data preparation and visualisation:** Adobe Photoshop and GIMP for image pre-processing and annotation.

- ⋆ **Build acceleration and version control:** IncrediBuild for distributed build acceleration and TortoiseSVN for source code versioning.

- ⋆ **Documentation:** TeXStudio and MiKTeX for LaTeX writing and Beamer for presentation preparation.

Chapter 2

Individual Characterization

Every individual possesses unique characteristics. These characteristics form the basis of identification and authentication systems, which are becoming increasingly important for highly secure applications. In recent years, there has been a considerable increase in demand for automatic systems that can identify or verify identity, whether for controlling access to personal data or assisting forensic experts in analyzing handwritten documents. This chapter introduces the key concepts of writer recognition, i.e. identification and verification (or authentication).

2.1 Introduction

Biometrics refers to the technologies used to measure and analyze human characteristics, whether physiological or behavioural, for identification or authentication purposes. As each person has unique traits, biometrics is a reliable method of personal identification. Unlike traditional identifiers, such as names, passwords or ID cards, which can be lost, stolen or forgotten, biometric traits are intrinsic to the individual (see 2.1).

Biometric systems operate in two main modes: verification (1:1 comparison) and identification (1:N comparison). In verification mode, the system confirms a claimed identity by comparing the input biometric data with a stored template (i.e. producing an accept/reject decision). In identification mode, the system determines an individual's identity by comparing the input data with multiple stored templates to find the best match (i.e. the most likely match in closed-set identification or declaring no match in open-set identification) [4].

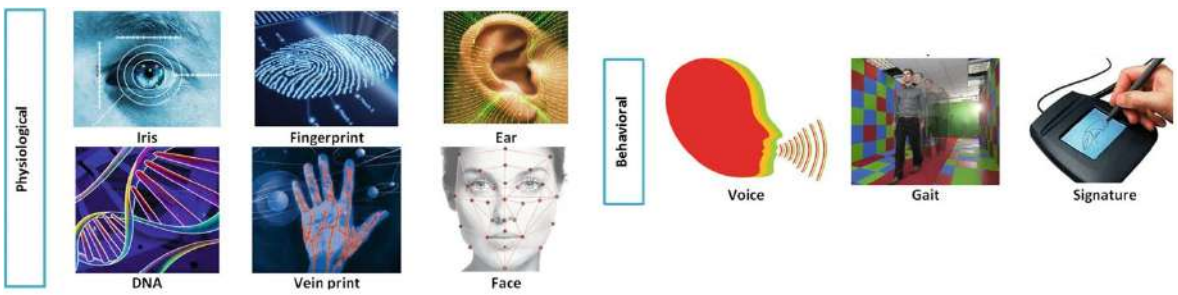


Figure 2.1: Biometric traits that can be used for identification or verification.

2.2 Handwriting as a Behavioral Biometric Trait

Offline handwriting, in particular, has attracted significant interest among various behavioural biometric modalities due to its natural use in many real-world scenarios, such as legal documents, forms, and historical archives. Handwriting contains many personalized features that can reflect an individual's motor habits, cognitive style and neuromuscular characteristics.

Writer recognition, also known as biometric recognition based on handwriting samples, plays a crucial role in the field of pattern recognition. It is a classical problem in forensic science with applications in several significant research domains. These include forensic examination and criminal justice systems, paleography (i.e., the study of historical and medieval handwriting), biometric recognition, security, and handwriting identification and verification. Forensic writer recognition typically involves the task of automatically determining the authorship of a given handwriting sample (e.g., a suspicious or fraudulent letter, ransom note, etc.). This task is commonly divided into two main modes: identification and verification. In the identification mode, which is a one-to-many comparison or multi-class classification problem, the objective is to determine the author (or writer) of an unknown handwritten sample from a set of known writers. In contrast, the verification mode focuses on determining whether two handwriting samples were produced by the same individual. This involves one-to-one comparisons and is framed as a binary classification problem [5].

Writer recognition process is challenging due to intra-writer variability (i.e. variations in an individual's writing over time or under different conditions) and inter-writer similarity (i.e. similarities between different individuals' writing styles) (see 2.2). Nevertheless, handwriting has one key advantage: it can be acquired non-intrusively using simple devices such as scanners or cameras, making it ideal for forensic and commercial applications [11].

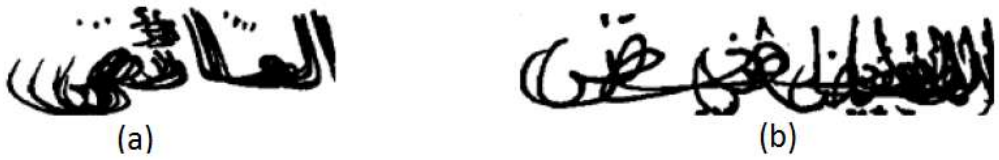


Figure 2.2: Illustration of intra- and inter-writer variability by superimposition: (a) a word written 6 times by the same writer and (b) the same word written by 6 different writers.

2.3 Within-writer variance vs. Between-writer variation

Writer identification and verification are only possible to the extent that the variation in handwriting style between different writers exceeds the variations intrinsic to every single writer considered in isolation. The results reported in this thesis ultimately represent statistical analyses, on our datasets, of the relationship opposing the between-writer variation and the within-writer variability in feature space. The present study assumes that the handwriting was produced using a natural writing attitude. It is important

to observe that forged or disguised handwriting is not addressed in our approach. The forger tries to change the handwriting style usually by changing the slant and/or the chosen allographs. Using detailed manual analysis, forensic experts are sometimes able to correctly identify a forged handwritten sample. On the other hand, our proposed algorithms operate on the scanned handwriting faithfully considering all the graphical shapes encountered in the image under the premise that they are created by the habitual and natural script style of the writer.

2.4 Factors causing variability in handwriting

Figure 1.1 shows four factors causing variability in handwriting [11]:

1. *Affine transforms*: Affine transforms are under voluntary control. However, writing slant constitutes a habitual parameter which may be exploited in writer recognition;

2. *Neuro-biomechanical variability*: Neuro-biomechanical variability refers to the amount of effort which is spent on overcoming the low-pass characteristics of the biomechanical limb by conscious cognitive motor control;

3. *Sequencing variability*: Sequencing variability becomes evident from stochastic variations in the production of the strokes in a capital E or of strokes in Chinese characters, as well as stroke variations due to slips of the pen;

4. *Allographic variation*: Allographic variation refers to individual use of character shapes.

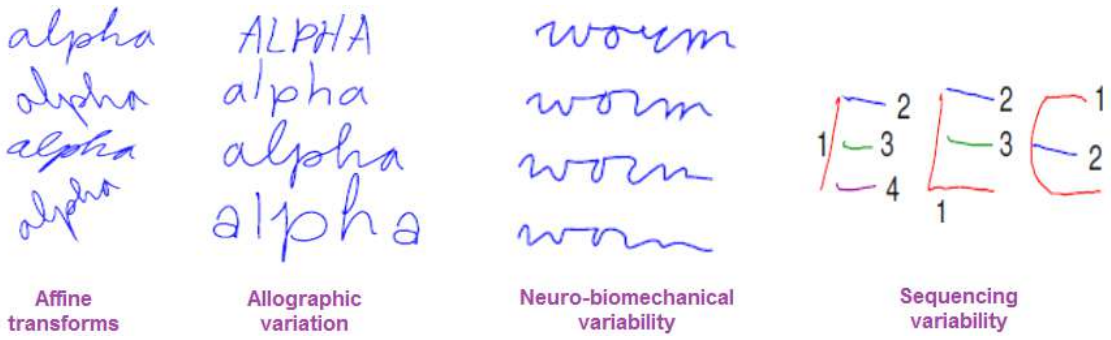


Figure 2.3: Factors causing handwriting variability [11].

2.5 Writer identification vs. Writer verification

Handwriting, as a form of behavioral biometric trait, encapsulates the neuromuscular patterns and cognitive processes unique to each individual. The act of writing is influenced by a combination of motor coordination, learned habits, psychological state, and physiological conditions, making it a rich source of biometric data. These characteristics are particularly useful for tasks such as writer identification and writer verification, which are two core problems in the domain of handwriting biometrics.

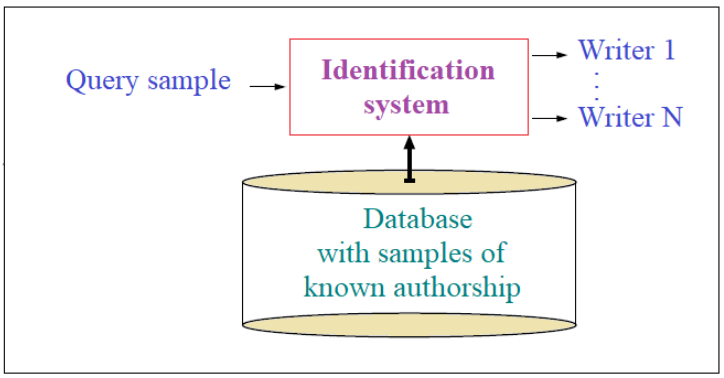


Figure 2.4: A writer identification system retrieves, from a database containing handwritings of known authorship, those samples that are most similar to the query. The hit list is then analyzed in detail by a human expert [11].

After image preprocessing, asserting writer identity based on handwriting images requires three main operational phases:

1. Feature extraction;
2. Feature matching / feature combination
3. Writer recognition (identification and verification).

Figure 2.5: A writer verification system compares two handwriting samples and takes an automatic decision whether or not the input samples were written by the same person [11].

A *writer identification* system performs a one-to-many search in a large database with handwriting samples of known authorship and returns a likely list of candidates (see Fig. 2.4). This list is further scrutinized by the forensic expert who takes the final decision regarding the identity of the author of the questioned sample [23]. *Writer verification* involves a one-to-one comparison with a decision whether or not the two samples are written by the same person (see Fig. 2.5). The decidability of this problem gives insight into the nature of handwriting individuality [6]. In writer identification searches, all the samples in the dataset are ordered with increasing dissimilarity (or distance) from the query sample. This represents a special case of image retrieval, where the retrieval process is based on features capturing handwriting individuality [5]. In writer verification trials, if the distance between two chosen samples is smaller than a predefined threshold, the samples are deemed to have been written by the same person. Otherwise, the samples are considered to have been written by different writers. Writer verification has potential applicability in a scenario in which a specific writer must be automatically detected in a stream of handwritten documents. In forensic practice, both identification and verification play a central role [11].

2.6 Online and Offline Writer Recognition

Traditionally, writer identification and verification systems are divided into two kinds: online and offline [40]. The online handwritten data contain more information about the writing style of a person, such as pentip position, pressure, and/or angles (as a function of time), that is not available in the offline data. However, offline writer identification and verification systems work on digitized images of writing. Consequently, compared to online writer identification, it is more difficult for offline systems to achieve significantly higher recognition rates, and offline systems are widely considered more challenging than their online counterpart.

2.7 Text-dependent vs. Text-independent methods

The techniques for writer recognition are traditionally categorized into two broad classes: *text-dependent* and *text-independent* methods. Text-dependent methods require the writing samples to be compared to contain the same fixed text. Signature verification, for example, can be considered in this category. These methods usually compare individual characters or words from a known transcription. Therefore, the text must first be recognized or segmented (manually or automatically) into characters or words before writer recognition can occur [7]. Text-independent methods, on the other hand, verify the author of a document regardless of its semantic content. These methods use features extracted from an entire text image or a region of interest. Only a minimal amount of handwriting is necessary to derive stable, text-insensitive features [12].

2.8 Conclusion

In this chapter, we learned that characterizing individuals is not a new concept. Based on the measurement of physical or behavioral characteristics, also known as biometrics, this concept dates back to the 14th century. Studies of biometrics have revealed four essential properties of characteristics: distinction, permanence, quantifiable, and universality. According to these properties, each characteristic has its own set of advantages and disadvantages. In order to identify an individual based on their handwriting, a behavioral biometric characteristic, this type of system must be implemented. We presented a schematic of this type of system in identification and verification modes. In identification mode, the user does not claim an identity; rather, the system recognizes the user. In verification mode, the system authenticates an identity claimed by the user. We found that writing is the central element of this thesis. We then described the characteristics and variability factors of handwriting, as well as the types of variations necessary for a writer identification or verification system. In the next chapter, we will review the various offline writer identification and verification approaches proposed in the literature.

Chapter 3

Writer Identification and Verification: State-of-the-Art

3.1 Introduction

Despite the development of electronic documents and predictions of a paperless world, the importance of handwritten documents has retained its place and the problems of identification and authentication of writers have been an active area of research over the past few years. A wide variety of systems that are based on the use of computer image processing and pattern recognition techniques have been proposed to solve the problems encountered in automatic analysis of handwriting and recognition of the writer of a questioned document. A comprehensive survey of the work in writer recognition until 2022 has been presented in [4]. Here, we will be more interested in surveying the approaches developed in the last several years, thanks to the renewed interest of the document analysis community for this domain.

This section offers a summary of research initiatives, highlighting progress in the filed of offline text-independent writer recognition (i.e., identification and verification). A key component of writer recognition systems is the essential process of feature extraction, which entails converting an input handwriting image into a feature vector representation (for instance, PDF or Probability Distribution Function as cited in [7]). The existing literature classifies feature extraction methods into two primary categories: *handcrafted features* and *learned features*.

3.2 Handcrafted features

Numerous effective techniques for well-established handcrafted features have been documented in the literature for the purpose of identifying and verifying the authorship of handwritten texts [5]. This initial category can be further divided into three primary classifications: 1. codebook-based techniques, 2. texture-based methods, and contour-based methods.

3.2.1 Codebook-based methods

The codebook or bag-of-shapes can be used as a distinctive measure (i.e., feature vector) to characterize the writing style of each writer [7, 42, 16, 28]. The shape occurrence probability is a characteristic for a given writer and may be employed to distinguish

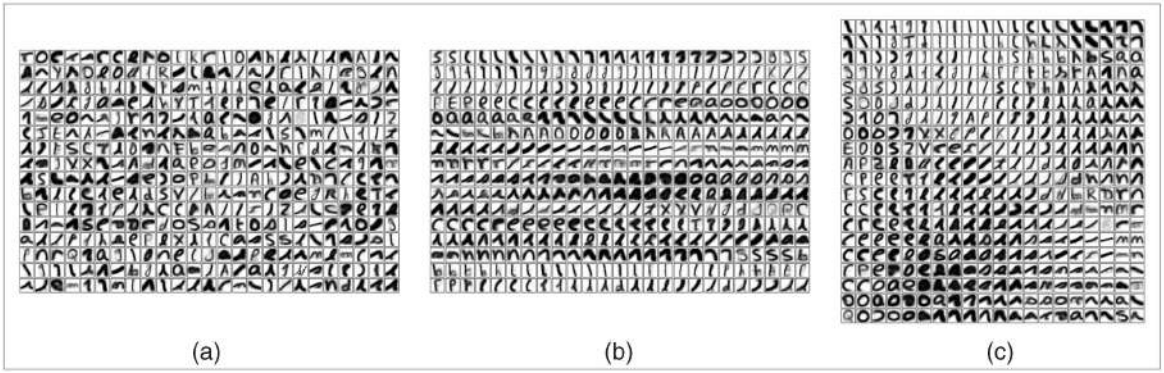


Figure 3.1: Examples of codebooks with 400 graphemes. For (a) kmeans and (b) ksom1D the graphemes have been placed 25 in a row, while, for (c) ksom2D, the 20×20 original SOM organization has been maintained.

between intra-cluster and inter-cluster similarity. Several methods are proposed in the literature to create the codebook, such as the Graphemes (writing fragments or small patterns) [11], Elliptic Graphemes [1], Junclets [25], Bagged Discrete Cosine Transform (BDCT) descriptors [30], and Implicit Shape (Patches around the key points) [8]. Regarding the allographic features, the writer is considered as a stochastic generator of graphemes and a set of emission shape probabilities is computed by building a normalized histogram. In 2007, Bulacu et al. [11] proposed the use of allographic features to characterize writer individuality thus making the approach quite similar to Bensefia et al. (2005). The probability distribution of each writer is computed using a common codebook of 20×20 graphemes generated with the ksom2D clustering algorithm for each document individually (see Fig. 3.2).

In 2015, Abdi and Khemakhem [1] used a beta-elliptic model (Elliptic Graphemes) to generate a synthetic codebook for Arabic writer recognition (see Fig. 3.1). A total of 60 feature vectors were extracted using template matching.

In the same year, He et al. [25] applied the detected junctions to writer identification using a compact representation, called Junclets. The junction detection in handwritten images is determined by analyzing the stroke-length distribution in every direction around a reference point inside the ink of texts. In 2017, Khan et al. [30] introduced a text independent writer identification method based on Bagged Discrete Cosine Transform (BDCT) descriptors. Universal codebooks are first employed to generate multiple

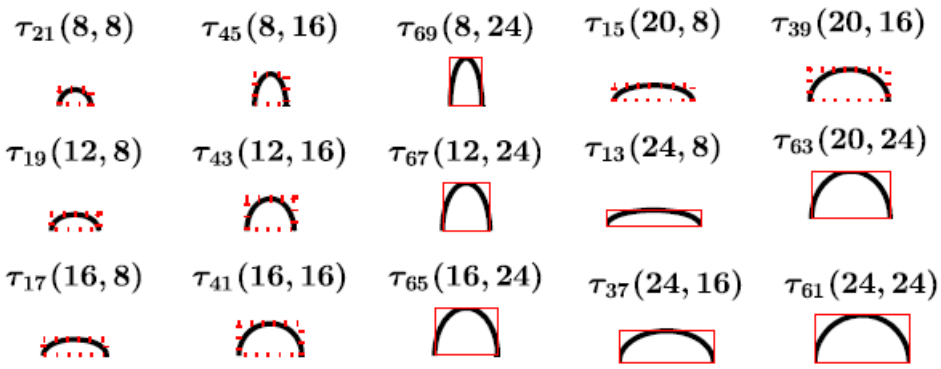


Figure 3.2: Examples of elliptic templates [1].

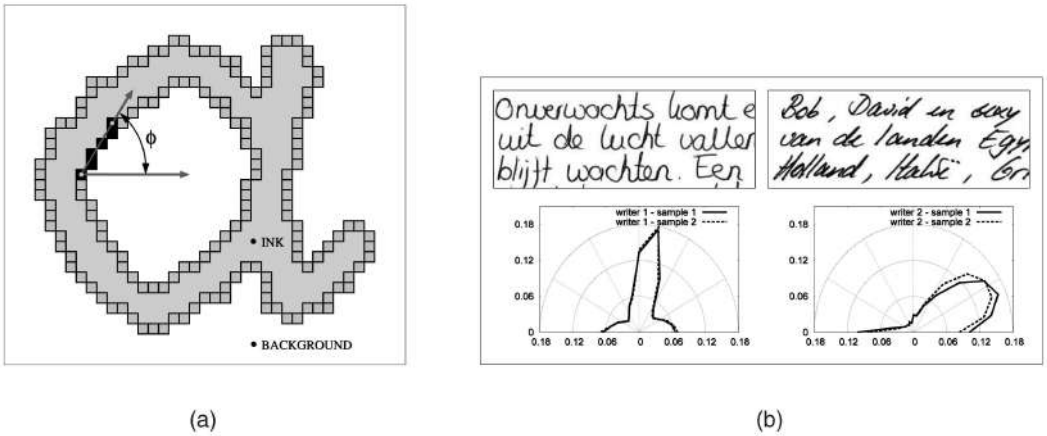


Figure 3.3: (a) Schematic description for the extraction method of the contour-direction PDF (feature f1). The handwritten letter "a," provided as an example, would be roughly twice as large in reality. (b) Examples of lowercase handwriting from two different subjects [11].

predictor models and the final result on a handwritten sample is obtained using the majority voting rule from all predictor models. In 2019, Bennour et al. [8] exploited the key points in handwriting to generate the implicit shape codebook and identify the right writers. The problem with this type of methods is the complexity of the segmentation and feature extraction algorithms, which are highly expensive in terms of execution time.

3.2.2 Texture-based methods

Texture-based approaches to perform offline automatic writer identification take into account each grayscale handwriting image as a unique texture and rely on extracting features from the entire image (whole document) [11, 15, 22], blocks [9], connected-components [4, 5, 6, 7, 12, 29], or writing fragments [20] to produce texture descriptors (i.e. features, measures, or attributes). A Probability Distribution Function (PDF or vector of probabilities) in a given document is computed and employed to characterize the writer's style [11]. In this case, the appearance probability; i.e. histogram bin, is

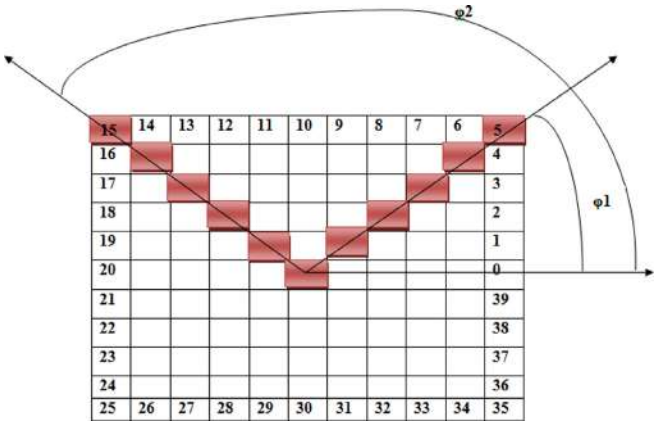


Figure 3.4: Extraction of edge-hinge distribution [27].

a characteristic for each writer and may be employed to distinguish between inter-class and intra-class similarities.

In 2007, Bulacu et al. [11] proposed to perform handwriting writer identification by using the textural features. Probability Distribution Functions (PDFs) are calculated using the entire handwriting image (document) and its contours for contour direction (see Fig. 3.3), contour-Hinge, direction co-occurrence and Run-Length on white distributions (RL). In 2013, Bertolini et al. (2013) published a texture-based descriptor for writer identification. The authors used Local Binary Patterns (LBP) in a comparative study with a Local Phase Quantization (LPQ) and concluded that LPQ performed better than their LBP variant.

One year later, Wu et al. (2014) wished-for a method for offline text-independent writer identification based on the Scale Invariant Feature Transform (SIFT). In particular, the SIFT Descriptor Signature (SDS) and the Scale Orientation Histogram (SOH) are extracted from handwriting images to characterize different writers. In 2016, we offered a set of textural features to characterize writer individuality [7]. These features include Direction-Length, Angle-Length, Direction Co-Probability, and Angle Co-Probability of connected components. In 2018, Singh et al. [43] divided cursive handwriting into nine texture blocks to compute histograms of the Local Binary Pattern (LBP) and the Center Symmetric Local Binary Co-occurrence Pattern (CSLBCoP). For each writing sample, a set of 9 blocks were created and feature vectors were computed for writer identification. Kessentini et al. [27] combined Edge-Hinge with a fragment length of 6 and 7 pixels (see Fig. 3.4) and Run-length features in the Dempster-Shafer Theory (DST) model to improve the identification rates (in the same year). In 2019, Hannad et al. [19] presented an approach for the identification and characterization of different writers that combines two textural features: the Histogram of Oriented Gradients (HOG) and Gray Level Run Length (GLRL) Matrices. Before the end of 2019, Khan et al. [29] applied a combination of the Scale-Invariant Feature Transform (SIFT) and RootSIFT descriptors in a set of Gaussian mixture models (dissimilarity SGMM and DGMM) to compare and classify the handwritten documents. In 2020, Chahi et al. [12] developed a Local gradient full-Scale Transform Patterns (LSTP) based method for writer identification. The authors extracted a set of feature code maps from resized

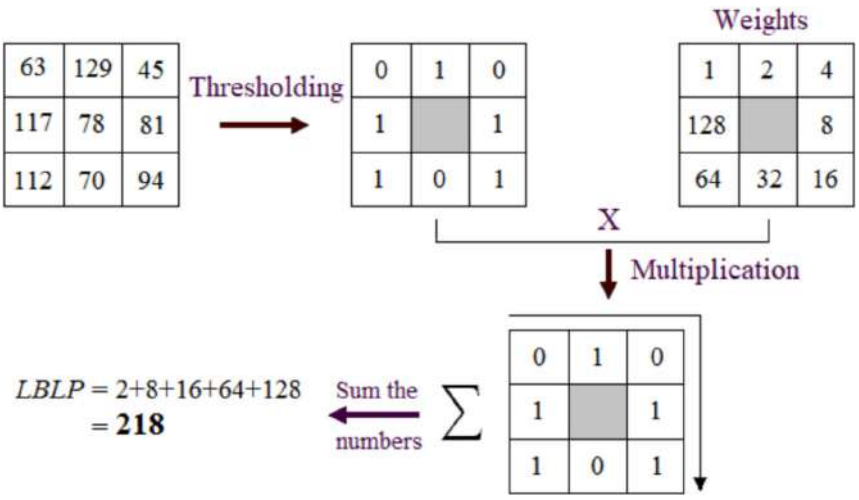


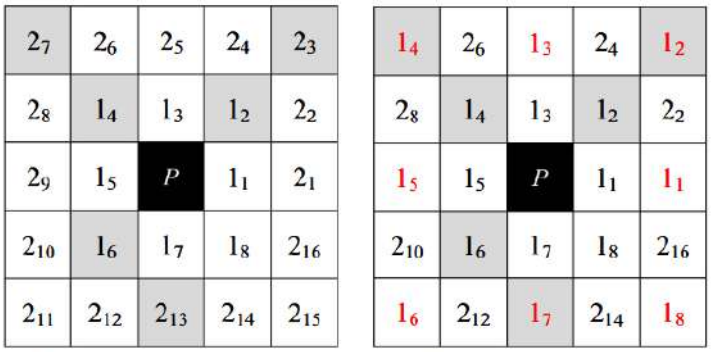
Figure 3.5: Example of LBLP code calculation [5].

small regions of interest of the writing (connected components sub-images) based on the distribution of local intensity gradients. Lastly, Bahram et al. [10] introduced a feature termed the Local Black Pattern Histogram (LBLP, see Fig. 3.5), which is grounded in texture connected-components and involves monitoring each black point for the purpose of text-independent writer identification (in 2025).

3.2.3 Contour-based methods

Contour-based techniques constitute a unique category within the field of offline writer identification and verification [27]. In these methodologies, Probability Distribution Functions (PDFs) of local attributes or features are derived from the inner and outer contour pixels, opposed to the conventional image pixels. These attributes are meticulously designed to capture characteristics unique to individual writer, such as angle, direction, curvature, slant, ink-trace widths, and letter shapes [4].

In 2015, Xiong et al. [46] approached writer identification from a texture contour perspective, utilizing the SIFT descriptor along with contour directional features (CDF) (see Fig. 3.6).



(a) Grid microstructure feature (b) Contour-directional feature

Figure 3.6: Comparison of GMF and CDF [46].

One year later, Bahram et al. [7] introduced four features: direction-length (DL), direction co-probability (DCP), angle-length (AL), and angle co-probability (ACP). These features are based on texture contours and involve monitoring each contour point to identify writers, regardless of the text. Expanding upon the co-occurrence feature for writer recognition, contour-based features can be formulated, including CoHinge and QuadHinge. Importantly, Bahram [4] released a notable writer identification system in 2022 that employs texture contour-based features. This system harnesses the joint distribution of the Modified Local Binary Pattern (MLBP) and the Ink-trace Width and Shape Letters (IWSL) measurements (MLBP-IWSL) to capture complex handwriting details, encompassing direction, curvature, slant, and shapes, thereby characterizing individual writing styles. Finally, in the same year, Bahram et al. [5] investigated the relationship between ink direction and ink width (width patterns or width ink traces) to aid in writer recognition through the Local Contour Pattern with Ink-trace Width and Shape Letters (LCP-IWSL) feature.

3.3 Learned features

In this section, features that have traditionally been crafted by experts, including those based on texture, contour, and codebook methodologies, can now be automatically derived through the implementation of Convolutional Neural Networks (CNNs) [44]. This approach enables the generation of features that are learned directly from the data, specifically aimed at offline writer identification and verification [47]. A review of existing literature reveals that most methodologies utilized for feature extraction are trained on handwriting samples comprising fragments, words, or lines. Notably, in 2015, Fiel and Sablatnig [17], along with Christlein et al. [14], emerged as pioneers in the use of CNNs within the field of writer recognition, employing scanned handwriting images. Following this, Javidi and Jampour [26] presented a hybrid deep architecture that merges handcrafted and deep features to define writer individuality, offering a streamlined extended version of FragNet that leverages auxiliary information from the Handwriting Thickness Descriptor (HTD). Kumar and Sharma [32] proposed a segmentation-free model that employs a CNN to identify query writers, referred to as SEG-WI. Moreover, in 2021, reference [24] introduced a technique for extracting global-context information from handwritten word images using CNNs, incorporating a recurrent neural network (RNN) to model the spatial relationships among sequences of fragments. Another CNN-based methodology was elaborated in [41], concentrating on deep feature extraction around key points within small handwriting patches. Key points were detected using the FAST and Harris corner detectors, while the feature encoding process utilized V LAD and triangulation embedding (TE). More recently, in 2023, Chahi et al. [13] unveiled a multipath deep CNN-based technique named WriterINet, designed to learn and directly identify writers. Kumar [34] introduced a writer identification modified Histogram of Oriented Gradients (HOG) to create writer descriptors, which can be used to determine the authorship of a specific handwritten word image. Furthermore, in 2024, Briber and Chibani [10] developed a closed system consisting of multiple convolutional layers. frequently trained on entire documents, to achieve superior performance, albeit at a considerable computational expense. Finally, in 2025, Zhao et al. [48] introduced a writer identification method based on an integration strategy combining complementary learning models. This approach involves combining multiple probabilistic mass functions, i.e. the properties of the top-k values.

3.4 Comparative Analysis and Challenges

It is crucial to emphasize that offline writer recognition employing textural features (such as contours or images) within a single-script context has shown high recognition rates, has proven efficient regarding processing time, and is generally preferred when at least three lines of text are accessible for training and testing purposes [6]. These approaches are typically not intended for the automatic identification or verification of authors from handwriting samples that incorporate multiple scripts, such as Arabic and French [41]. Moreover, they require a significant amount of handwritten text samples from each individual to attain effective classification and are sensitive to structural variations [4, 41]. It is important to highlight that techniques based on learned features exhibit robustness and enhanced recognition performance, especially at the word level [33]. However, these methods necessitate a substantial number of samples and entail

considerable time investment concerning parameter tuning and calculations during the training phase, with a tendency towards over fitting. In contrast, codebook-based approaches, similar to forensic analysis, necessitate a complex process, incur significant execution times, and often exhibit limitations in overall recognition performance. In light of the aforementioned observations, this paper introduces a texture- and contour-based technique for offline writer identification and verification, focusing on both image and contour analysis. The handwriting features utilized in this study are derived from images of short paragraphs, each comprising approximately three lines. These features have been employed to characterize distinct handwriting styles across various scripts and languages, including English, French, German, Greek, and a hybrid language. A comprehensive description of our methodology can be found in Chapter 4, while Chapter **??** discusses the efficacy of the proposed features in writer recognition, utilizing established benchmark databases.

3.5 Conclusion

Identifying writers in multilingual contexts remains challenging, primarily due to variability in scripts and differences in text length. While previous benchmarks such as ICDAR 2011, ICDAR 2013 and CVL have primarily focused on monolingual (Latin script) scenarios, these datasets remain valuable for evaluating new approaches.

In this study, we proposed two novel text-independent features for writer recognition and evaluated their effectiveness on short and long handwritten samples. Using the ICDAR 2011 and 2013 datasets, we demonstrated that these features can capture writer-specific traits with high discriminative power across varying conditions. The experimental results, detailed in the following chapters, confirm the potential of our features to generalize beyond monolingual settings and limited text lengths. These findings suggest promising avenues for developing robust, language-independent writer identification systems.

Chapter 4

Proposed approach

4.1 Introduction

In this section, two primary representations of the scanned handwriting images are employed during the feature extraction step: the binary connected-components and their exterior contours. The contour contains a sequence of pixels located on the ink-background boundary of a connected-component. This is a very effective vectorial representation that will allow a fast computation of the proposed features. In addition, contours are important cues to describing writing shapes. For writer characterization, we propose two complementary features based on the Local Contour Pattern with Ink Width and Stroke Length (LCP-IWSL) framework to characterize writers.

1. $LCP-IWSL_B$ (Black pixels): This feature was extracted from connected black components, or ink traces, to capture the local texture patterns and structural attributes of handwriting. These attributes included variations in stroke width and length;

2. $LCP-IWSL_C$ (Outer contour pixels): It is extracted from the outer contours of connected components, encoding complementary texture and contour information. This improves the ability to distinguish between writers.

As illustrated in Fig. 4.1, our proposed approach consists of three main processing steps:

1. Preprocessing: Normalizes and enhances scanned handwriting images to ensure consistent feature extraction across different datasets;

2. Feature Extraction: Computes $LCP-IWSL_B$ (black pixels) and $LCP-IWSL_C$ (white pixels) from binary connected components and their contours. Then, it combines them to encode characteristics specific to the writer;

3. Classification (recognition): It uses distance metrics, such as chi-squared (χ^2) to match query samples with a database of known writers.

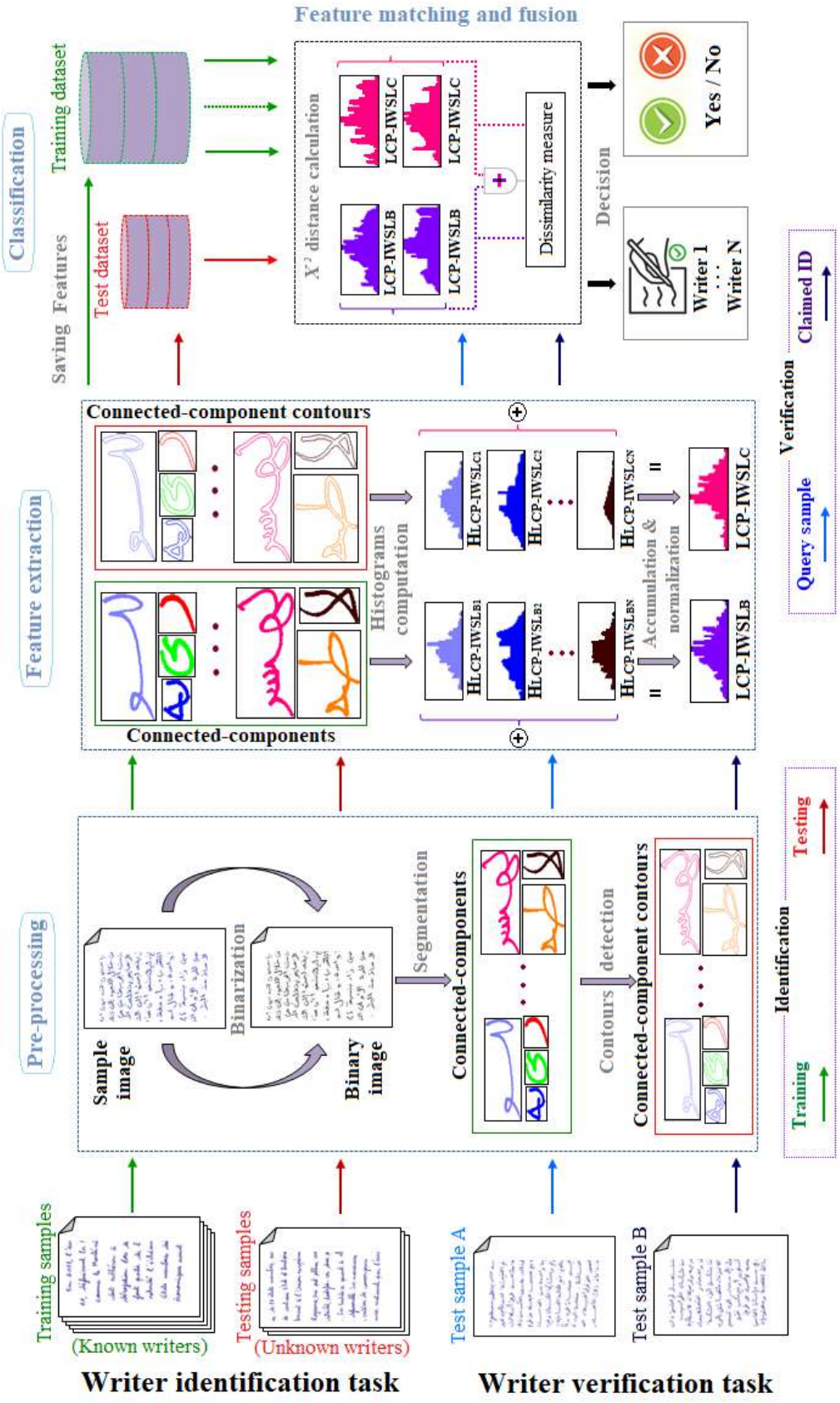


Figure 4.1: The framework of our writer recognition (identification and verification) system.

4.2 Preprocessing

Like most of the pattern recognition problems, the preprocessing step plays an important role in the performance of the writer identification system. For our experiments, the input data (i.e., feature extraction step) are in black connected-components and their contours, so the pre-processing step is necessary and it consists in binarizing the document images (foreground/background separation), extracting the connected-components, removing the non-significant connected components (i.e., small components, dots, etc.) and extracting the interior and exterior contours of connected components. In this section, a very effective vectorial representation is considered which will subsequently allow a fast calculation of the proposed feature.

4.2.1 Binarization

The vast majority of offline writer recognition techniques use binary input images rather than color or grayscale ones [6]. While no comprehensive studies appear to have been conducted, the available results suggest that this format is the most useful despite the loss of information. While grayscale may be appropriate for certain features (e.g. attempting to reconstruct pen pressure), it generally produces images that are too complex for further processing.

Binarization is a key step in preparing images for analysis. It takes a grayscale or color image and simplifies it by converting it to black and white. In this simplified image, only two colors exist: black for important details such as text or handwriting and white for the background, such as the paper. This simplification makes tasks such as character recognition, writer identification and verification, and shape and outline detection much easier. Depending on how the threshold is determined, binarization methods are commonly divided into two main categories.

1. *Global Binarisation* : Global binarisation applies a single threshold value to the entire image. Each pixel is classified as foreground or background based on this one global threshold.

   —Advantages: Simple, fast, and effective for images with uniform lighting and contrast.

   —Common method: Otsu's method, which automatically selects an optimal threshold by maximizing inter-class variance.

2. *Local Binarisation* : Local binarisation, also known as adaptive binarisation, computes a different threshold for each pixel based on the local neighborhood (a small window around each pixel). This makes it more suitable for documents with irregular lighting, shadows, and low quality.

   —Advantages: Durable to variations in lighting and background noise.

   —Common methods: Niblack, Sauvola, and Wolf algorithms, which dynamically adjust thresholds using local mean and standard deviation.

We decided to use Otsu's method [39] to binarize the handwriting images for our work. Our images are clean and well-scanned with good lighting and a clear distinction between text and background, so this method is ideal for our needs (see Fig. 4.2).

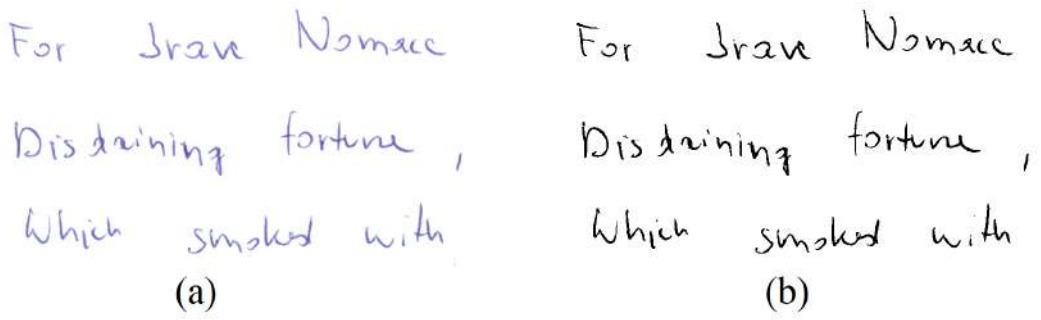


Figure 4.2: Global Image Thresholding using Otsu's Method [39].

Otsu's method automatically identifies the optimal threshold to distinguish black text from a white background, making it ideal for this situation. It's also fast and simple to use, making it a practical choice. Using this method allowed us to obtain clean binary images, making subsequent steps, such as contour detection and writer analysis, easier and more reliable.

4.2.2 Connected Components Extraction

Extracting connected components is a critical step in processing binary images. The goal is to identify and separate individual objects made up of connected pixels. In a black-and-white image, for instance, the black pixels typically represent meaningful content, such as handwritten text, symbols, or shapes, while the white pixels represent the background. The extraction process examines the image to find groups of black pixels connected to each other based on a connectivity rule. This rule typically considers 4-connectivity, which includes pixels connected horizontally or vertically, or 8-connectivity, which includes diagonal neighbors (see Fig. 4.3). Pixels that meet this condition are grouped together as one connected component. Once these groups have been identified, each connected component is assigned a unique label. This labeling allows us to treat each component as an individual object for further analysis. In handwriting recognition or writer identification, for example, connected components often correspond to letters, strokes, or words requiring separate analysis.

This step is crucial because it simplifies complex images by breaking them down into smaller, meaningful parts. After extracting connected components, other important tasks such as contour tracing, feature extraction, or classification become more manageable and accurate

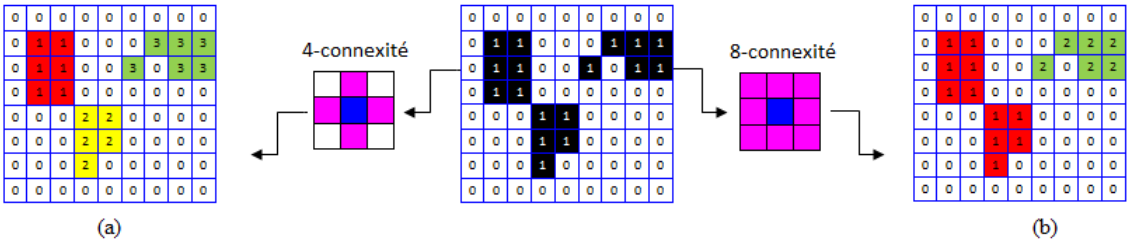


Figure 4.3: Connected components extraction, (a) 4-connectivity and (b) 8-connectivity.

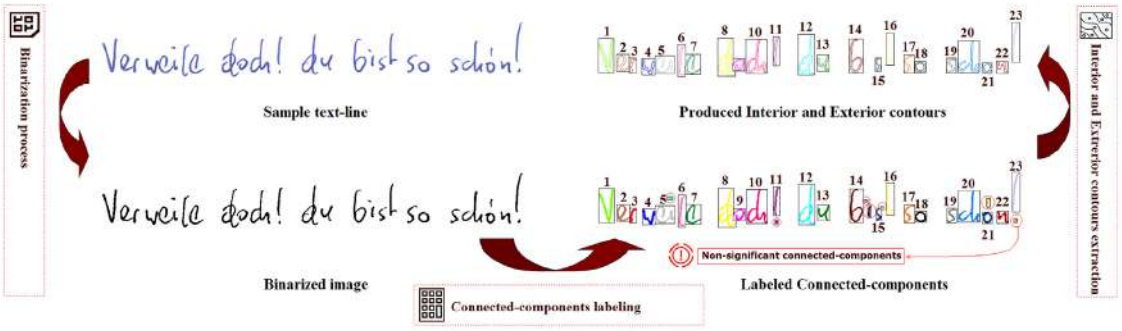


Figure 4.4: Preprocessing of an example handwritten text line taken from CVL [31] database (from [4]).

4.2.3 Removal of small connected components

As stated in Bahram et al. [7], a connected component can be defined as an ink blob or complete region of ink trace, determined by a set of pixels, i.e., all connected to each other, drawn without lifting the pen from paper. In binary images of handwriting, the foreground (black) pixels correspond to the black ink trace and the white pixels correspond to the background. In order to capture more details of the directional strokes, the slant and skew corrections of the handwritten document are not used in this work. Subsequently, some non-significant connected components like diacritics small-components with a width or height smaller than one stroke-width, scatter noise, periods, commas, and dots, are discarded in this step (see Fig. 4.3).

4.2.4 Contour detection

Contour detection is the process of tracing the outlines or borders of shapes in a binary image. In the context of handwriting analysis, these shapes represent letters or strokes. This step helps us understand the exact form and structure of each component, which is essential for a detailed analysis. Two types of contours are usually extracted: 1- *Outer contours* and 2- *Inner contours* (see Fig. 4.5). The Moore-Neighbor Tracing Algorithm is a widely used method for detecting contours [18]. It is simple and effective for binary images (see Fig. 4.6). As illustrated in Fig. 4.1 and Algorithm 1, each handwritten document, comprising various words or lines of text, is segmented into images of connected components $\{CC_i\}$. The pages from the CVL database [31], initially

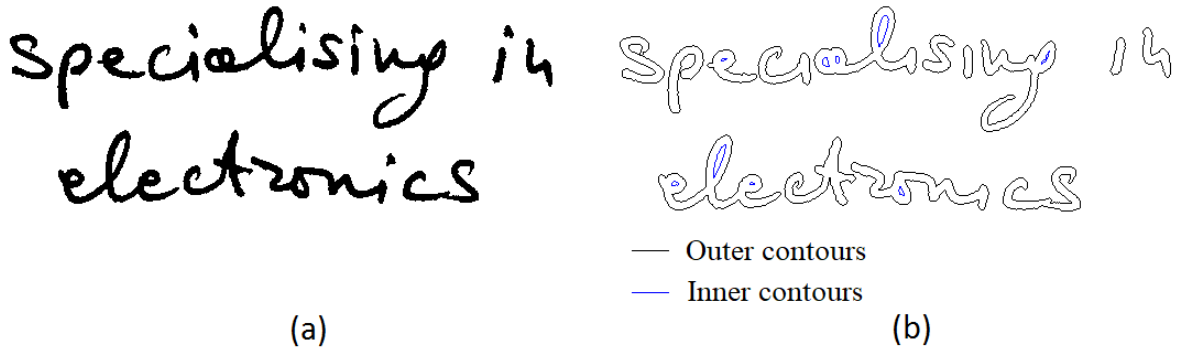


Figure 4.5: Outer and Inner contours detection [5].

available in RGB format [6].

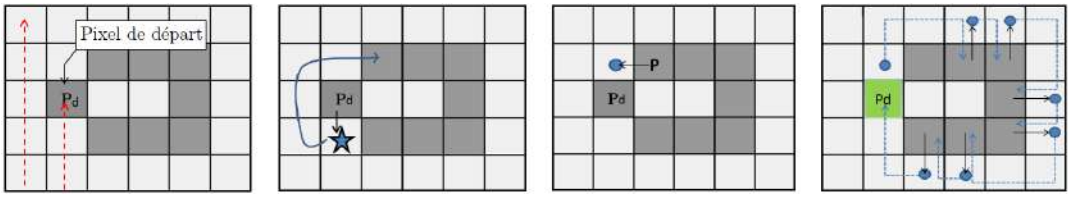


Figure 4.6: The different steps of the Moore algorithm — Neighbor Tracing [18].

Algorithm 1 : Preprocessing step

Input: Gray-scale image I_G
Output: $\bigcup_{i=1}^{N} CC_i, \quad \bigcup_{i=1}^{N} OC_i$

1: $I_B \leftarrow$ ***Otsu***(I_G)
2: **for** $i \leftarrow 1$ **to** N **do**
3: $\quad CC_i =$ ***CCExtract***$(I_B, X_{min}(i), Y_{min}(i), width(i), hight(i))$
4: $\quad$ **if** $(\psi(CC_i) > \xi)$ **then**
5: $\quad\quad$ **Add**(CC_i)
6: $\quad\quad$ **Add**$(OC_i \leftarrow$ ***Moore***$(CC_i))$
7: $\quad$ **else**
8: $\quad\quad$ **Skip**(CC_i)
9: $\quad$ **end if**
10: **end for**
11: **return** $(\bigcup_{i=1}^{N} CC_i, \bigcup_{i=1}^{N} OC_i)$

4.3 Feature Extraction

Feature extraction is the process of converting an input handwriting image into feature vectors with the aim of increasing the performance of the model. It is the heart of handwriting-based biometrics [1]. In the case of statistical features, the feature vector is calculated by carrying out a large number of measurements, accumulating the results in a histogram, normalizing and interpreting it into a probability distribution [11]. In this section, we characterize the individual writing style of a given writer by computing two texture- and contour-based features: $LCP\text{-}IWSL_B$ and $LCP\text{-}IWSL_C$.

4.3.1 LBP operator

The traditional Local Binary Pattern (LBP) operator proposed by Ojala et al. in [37] and improved in [38] for texture analysis and description is a fast and computationally efficient texture descriptor for grayscale images. The LBP descriptor has been used to identify the authorship of handwritten documents in most of the texture techniques in the literature [9, 22], which divides the entire image into regions to extract unique and useful texture features. This powerful and most popular operator labels each center (or referenced) pixel p_c of a grayscale image I_G by thresholding its eight surrounding pixels $(p_0, \ldots, p_7)$, and considering the result as a binary string (i.e., obtained by concatenating

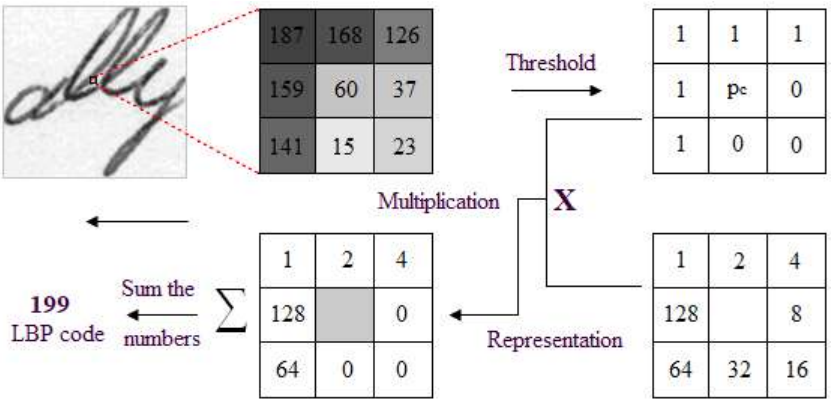


Figure 4.7: Example of the traditional LBP operator [37, 38].

all these binary values) or a decimal number. Fig. 4.7 shows an example of the original grayscale LBP code, which is computed by comparing the center pixel (x_c, y_c) intensity with eight neighbors (x_k, y_k) in a 3×3 neighborhood.

A limitation of the basic LBP operator itself is its small 3×3 neighborhood around the central pixel: this may leave it unable to capture the dominant features with large-scale structures. Subsequently, the original LBP operator was generalized to use different neighborhood sizes, i.e. to capture discriminative features at different scales [38]. For a given pixel p_c , its P neighbors $\{p_k\}_{,k=0...P-1}$ are obtained using a bilinear interpolation. Formally, the LBP operator of the center pixel p_c at position (x_c, y_c), can be expressed as (cf. Eq. 4.1)

$$LBP_{P,R}(x_c, y_c) = \sum_{k=0}^{P-1} s(i_k - i_c)\ 2^k \tag{4.1}$$

where i_c represents the gray value of the center pixel p_c, i_k represents the gray values of P surrounding pixels on a circle of radius $R(R > 0)$, P is the number of neighbors, 2^k is the binomial factor, and s defines the sign function as follows (cf. Eq. 4.2):

$$s(i_k - i_c) = \begin{cases} 1, & if \quad i_k \geq i_c \\ 0, & if \quad i_k < i_c \end{cases} \tag{4.2}$$

Fig. 4.8 illustrates an example of the extended LBP operator using different neighborhood sizes (i.e. different values of P and R).

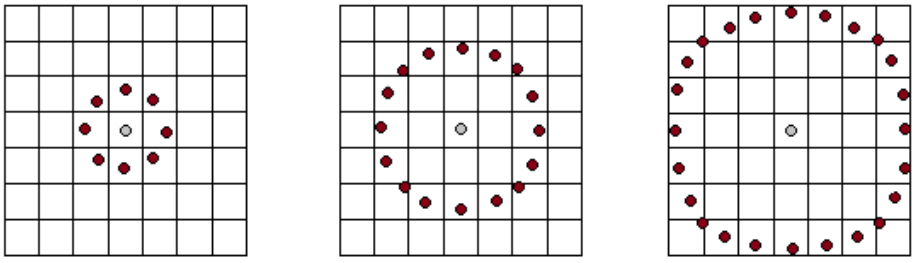


Figure 4.8: Examples of the extended LBP operator [38] (three different values of P and R).

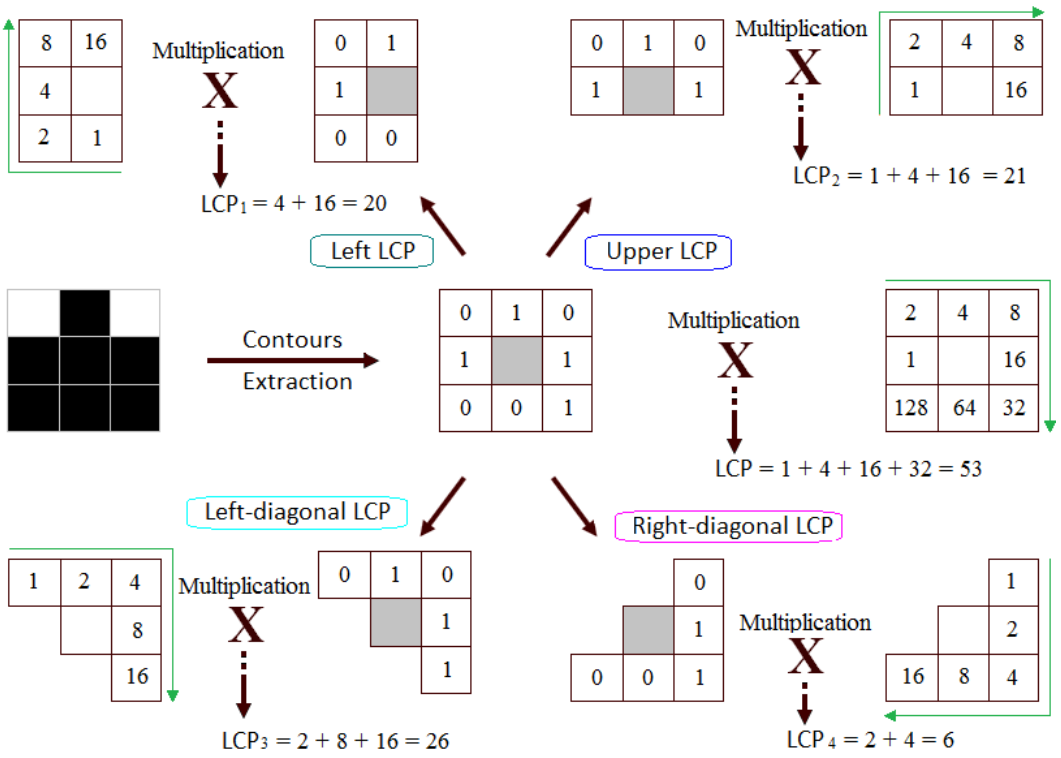


Figure 4.9: Computing the *LCP* code and its splitting into *Left*, *Upper*, *Left-* and *Right-Diagonals LCP* codes ($P = 8$ and $R = 1.0$).

4.3.2 LCP-IWSL feature

As stated in [20, 22], the LBP histograms give a rough idea of the texture information of the writing. However, this may not be very effective in capturing the fine details of the writing. Therefore, we propose a local geometrical structure called the Local Contour Pattern (LCP).

LCP operator The fundamental objective of this study is the computation of texture measures from the contours $\mathcal{C} = \{C_0, C_1, \ldots, C_{N-1}\}$ of black connected-components $I_B = \{CC_0, CC_1, \ldots, CC_{N-1}\}$ to characterize a handwritten sample. Where N is the number of connected-components in the document I_B (binary image). The binary LCP operator labels each black pixel p_c of coordinates (x, y) in connected-component CC_i ($p_c \in CC_i$ and $p_c \notin C_i$) by examining its surrounding P pixels $(n_0, \ldots, n_{P-1})$. This binary code LCP will be obtained by reading the values in a clockwise fashion and can be defined as (cf. Eq.4.3):

$$LCP_{P,R}(x, y) = \sum_{k=0}^{P-1} f(n_k) 2^k \tag{4.3}$$

where the coordinates of neighbors can be calculated as (cf. Eq.4.4)

$$(x_{n_k}, y_{n_k}) = \left(x + R\cos(\frac{2\pi p}{P}), y + R\sin(\frac{2\pi p}{P}) \right) \tag{4.4}$$

(P, R) denotes a neighborhood of P points on a circle of radius R of the connected-component CC_i, and the function $f(v)$ is defined as follows (cf. Eq.4.5):

$$f(v) = \begin{cases} 1, & if \quad v \in C_i \\ 0, & \text{otherwise} \end{cases} \tag{4.5}$$

Fig. 4.9 shows an example of LCP code calculation, the central black pixel p_c is labeled as 00110101 in binary or 53 in decimal ($P = 8$ and $R = 1.0$).

It is worth noting that, in general, the left-to-right languages (e.g. Latin, Greek) are written from left to right (horizontal lines) and from top to bottom. Some languages, such as Arabic and Farsi texts run like these, but from right to left. For this purpose, two Local Contour Pattern operators (LCP) are retained: *Left* LCP (or LCP_1) and *Upper* LCP (or LCP_2). In Fig. 4.9 the LCP_1 and LCP_2 from a connected-component (CC_i) are illustrated. For each pixel p_c at position (x, y) in a connected-component CC_i, the resulting LCP_1 and LCP_2 can be expressed in decimal as follows (cf. Eq.4.6 and 4.7):

$$LCP_1(p_c) = \sum_{k=0}^{1} f(n_{k+6})\ 2^k + \sum_{k=2}^{4} f(n_{k-2})\ 2^k \tag{4.6}$$

$$LCP_2(p_c) = \sum_{k=0}^{4} f(n_k)\ 2^k \tag{4.7}$$

In offline writer identification, diagonal features are essential and also play an important role as they achieve very satisfactory results on the most popular handwritten databases [4]. Therefore, we proposed two LCP operators, namely Left-Diagonal LCP (denoted by LCP_3) and Right-Diagonal LCP (denoted by LCP_4) within this step (see Fig. 4.8). The designed LCPs are computed for all candidate pixels according to the following formulas (cf. Eq.4.8 and 4.9):

$$LCP_3(p_c) = \sum_{k=0}^{4} f(n_{k+1})\ 2^k \tag{4.8}$$

$$LCP_4(p_c) = \sum_{k=0}^{4} f(n_{k+3})\ 2^k \tag{4.9}$$

Fig. 4.9 shows an example of Local Contour Pattern and its splitting into four LCP codes.

IWSL measurements The Run-Length (RL) distribution is the first feature proposed by Arazi et al [2] for automatic writer identification. Therefore, this is introduced as a powerful global descriptor for capturing the individual characteristics of handwriting. A *run* in a given direction is defined as the set of consecutive (connected) pixels having the same property, such as the gray level intensity [15]. The Run-Length can be computed from the binarized image, and it considers the white pixels corresponding to the white background and the black pixels corresponding to the black ink (foreground). The background run-lengths capture information about the region inter and intra letters (words) spacing and the roundness of characters, whereas the black run-lengths

capture information about the pen type used for writing (ink trace width). For example, the black ('1') and white ('0') run-lengths of the black and white bit sequences "11000010111001001" are encoded as: (2 1 2 1) and (4 1 2 2), respectively. The black (or white) run-lengths can be determined in four directions: horizontal, vertical, or diagonal orientations (right and left) (more details in [15, 22]). For each direction, the length of runs is agglomerated in a histogram, which represents the texture. This histogram is normalized and interpreted in a probability distribution (PDF), while it is used as a characteristic of each writer. In order to capture the textural information in a grayscale image, the run-length matrices are computed in a given scanning direction. In this case, a gray-level run is described as a number of pixels with the same gray level (intensity value) g. A Gray-level run-length matrix ($GLRLM$) is computed as follows. Each element $M(g, h)$ of matrix M corresponds to the number of runs with pixels of intensity value g and length of run (number of occurrences) h in the grayscale image without using any binarization algorithm. Therefore, run-length matrices can be computed by scanning either horizontally (0°), vertically (90°), or diagonally (45°, 135°). Similarly to the black and white run-length descriptors presented above, the histograms of grayscale run lengths are normalized to $PDFs$ and used in the identification process.

In this step, we propose a new measure that is mainly designed to capture the Ink-trace Width and Shape Letters in a writing, called $IWSL$. Here, we introduce the use of outer contour pixels. Therefore, an $IWSL$ measurement is the number of consecutive black and white pixels between the two outer contour points in a given direction. Formally, the proposed $IWSL$ can be determined as follows:

If $S = \{p_{cs}, \ldots, p_{ce}\}$ is a binary sequence of connected pixels (of connected-component CC_i) at a bi-level image ($p_{cs} \in C_i$, $p_{es} \in C_i$, and each pixel is stored as 1 or 0) in a given direction, p_{cs} and p_{ce} are respectively the start and end position of this sequence, and $p_1 \ldots p_n$ the pixels between the two outer contour points p_{cs} and p_{ce}.

The $IWSL$ is then computed as the distance from $p_{cs} = (x_{cs}, y_{cs})$ to $p_{ce} = (x_{ce}, y_{ce})$ and is evaluated using the Euclidean norm (cf. Eq.4.10):

$$IWSL = \sqrt{(x_{ce} - x_{cs})^2 + (y_{ce} - y_{cs})^2} \tag{4.10}$$

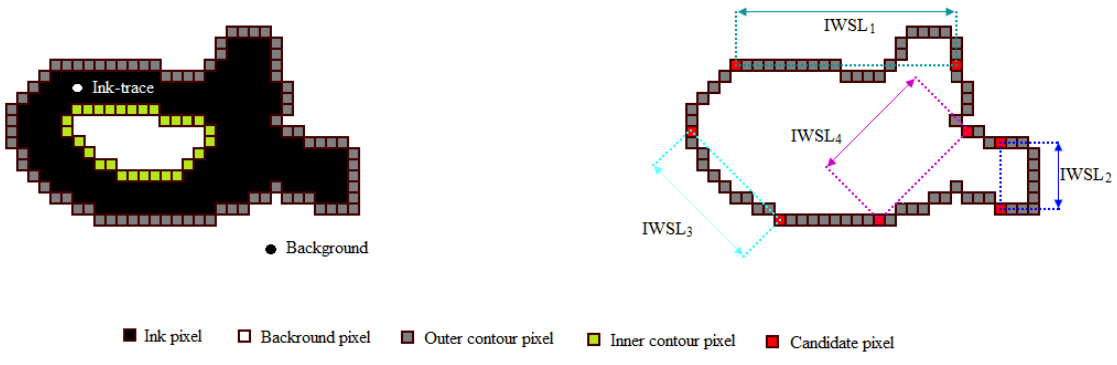


Figure 4.10: Computing local $IWSL$ in the four directions.

The $IWSL$ measure of outer contour pixels is computed as similar to the techniques described in [15, 22] while the four main directions: horizontal ($IWSL_1$), vertical ($IWSL_2$), left-diagonal ($IWSL_3$) and right-diagonal ($IWSL_4$) are exploited to capture the writing details (e.g. character shapes, average width of letters, ink-width). This measure is illustrated in Fig. 4.10 on a bi-level image.

IWSL for Black ($IWSL_B$) and Outer contour pixels ($IWSL_C$) The IWSL measurement can be applied to both black and black-and-white pixels, such as ink traces and the pixels between outer contours, respectively, to capture comprehensive structural information about handwriting patterns. This dual analysis provides complementary features that enhance the discriminative ability of the IWSL descriptor.

- $IWSL_kB = \sqrt{(x_{ce_B} - x_{cs_B})^2 + (y_{ce_B} - y_{cs_B})^2}$: This measurement captures the variations in stroke thickness that are characteristic of different writing instruments. It also captures information about character formation and pen trajectory (see Fig. 4.10).
- $IWSL_kC = \sqrt{(x_{ce_C} - x_{cs_C})^2 + (y_{ce_C} - y_{cs_C})^2}$: This measurement provides valuable information about the consistency in inter-character and inter-word spacing, writing density and spatial organization patterns and negative space distribution reflecting individual writing habits (see Fig. 4.10).

LCP-IWSL feature construction Over the past few decades, complex features (i.e., co-occurrence, co-probability) have been used for offline automatic (text-dependent and text-independent) writer identification such as *Hinge* [11], *Quill* [**?**], *coHinge* and *QuadHinge* [21], is one of the main and very effective features that has started to attract significant attention from researchers, especially in the field of forensic handwriting analysis.

In this approach, we propose a complex feature that is capable of introducing more information: Local Contour Pattern - Ink-trace Width and Shape Letters ($LCP - IWSL$) feature (cf. Eq. 4.11).

$$LCP - IWSL = \prod_{k=1}^{4} LCP_k - IWSL_k \tag{4.11}$$

where $\prod$ is the concatenation operator and $LCP_k - IWSL_k$ obtained by normalizing the histogram $H_{LCP_k\text{-}IWSL_k}$, which is defined as follows (cf. Eq.4.12):

$$LCP_k\text{-}IWSL_k = \frac{H_{LCP_k\text{-}IWSL_k}}{\|H_{LCP_k\text{-}IWSL_k}\| + \varepsilon} \tag{4.12}$$

where ε is a very small value close to zero and $H_{LCP_k\text{-}IWSL_k}$ is a histogram computed by using Eq.(4.13):

$$H_{LCP_k\text{-}IWSL_k}(l) = \sum_{i=0}^{N-1} \sum_{p_c \in CC_i} \delta\{l, ((LCP_k - 1)N_{IWSL} + IWSL_k)\}, \quad l = 1 \ldots (N_{LCP} \times N_{IWSL}) \tag{4.13}$$

It is important to note that the final feature vectors for $LCP - IWSL_B$ (black pixels) and $LCP - IWSL_C$ (outer contour pixels) are both of size $4,340$. Further details regarding the computation of the normalized histograms can be found in reference [4].

4.4 Classification (writer recognition)

We derive texture-based contour-based features $LCP - IWSL_B$ (black pixels) and $LCP - IWSL_C$ (outer contour pixels) from the black linked components and their outer contours prior to proceeding with classification, specifically writer identification and verification, which constitutes the final stage of the framework. Consequently, the dissimilarity measure between two distinct handwritten samples, as indicated by their feature histogram vectors, was employed to assess intraclass and interclass variations in the biometric writing traces.

4.4.1 Dissimilarity measure

In order to identify authors, it is essential to calculate the dissimilarity of feature histograms through a distance metric. We investigated and evaluated various established distance metric alternatives, including Euclidean, Chi-square, Manhattan, Bhattacharyya, and Minkowski (up to order 5). Following tests conducted on six widely used databases, we determined that the Chi-square (χ^2) distance proved to be the most effective and is utilized in this research.

This distance function is highly efficient and straightforward, making it ideal for probability distribution characteristics with tiny values [4]. Assume S_1 and S_2 are two handwritten samples, and $X = (x_1, x_2, \ldots, x_{size})$ and $Y = (y_1, y_2, \ldots, y_{size})$ denote the feature vectors (histograms or distributions) that will be compared. The Chi-square distance is used to compare the dissimilarity of two feature vectors (see Eq. 4.14).

$$d_{\chi^2}(X, Y) = \sum_{i=1}^{size} D(x_i, y_i) \tag{4.14}$$

The function $D(x_i, y_i)$ is defined as

$$D(x_i, y_i) = \begin{cases} 0, & if \quad x_i = y_i = 0, \quad i = 1 \ldots size \\ \dfrac{(x_i - y_i)^2}{(x_i + y_i)}, & otherwise \end{cases} \tag{4.15}$$

The dissimilarity of each feature vector (i.e., $LCP - IWSL_B$ and $LCP - IWSL_B$) is represented by $size$, and i is an index to the elements of X and Y.

4.4.2 Writer identification

The writer identification task involves comparing an input image of a handwritten document ($Q \in Ts$) to a dataset of known writers' handwriting (Tr), where Ts and Tr are the test and training datasets, respectively. For more information on this comparison, see [4]. The writer w_i is among the recorded in the system $\{w_1, w_2, \ldots, w_N\}$, where N represents the total number of writers in the training dataset.

The k-Nearest Neighbor (k-NN) classifier effectively recognized authors based on handwriting characteristics extracted from images. This classifier is frequently employed for pattern classification owing to its straightforwardness and efficiency. Furthermore, this approach utilizes established distance metrics to generate a ranked list of training samples that exhibit similarity in feature space to the query sample. This process

involves calculating the dissimilarity values between the test data and the reference data.

This study uses k nearest neighbor classification method (k-NN) along χ^2 distance and Holdout strategy [41, 34] to determine authorship. All of the text samples from each database have been divided into 2 sets (training and test datasets). As done in [29], one set have been used for training and the second set has been used for testing. Identification results are obtained after a hold cross validation. Furthermore, the effectiveness of our system is quantified as a percentage of the Top1, Top5 and Top10 identification rates.

4.4.3 Writer verification

For writer verification (identity verification), the system compares two samples of handwriting (S_A and S_B) to determine if they were created by the same person ($DS(S_A, S_B) = 1$) or not ($DS(S_A, S_B) = 0$) based on a threshold value τ. Using χ^2 (and Mh), we compute the distance between the normalized histograms X and Y retrieved from S_A and S_B, respectively, and choose the decision DS (refer to Eq. 4.16).

$$DS(S_A, S_B) = \begin{cases} 1, & if \quad d_{\chi^2}(X, Y) \leq \tau \\ 0, & otherwise \end{cases} \tag{4.16}$$

This step considers the following two errors: False Accept (FA) and False Reject (FR). The first mistake is assuming two handwriting samples are from the same person, which is not the case. The second definition is incorrectly assuming two samples were contributed by the same person.

By adjusting the decision threshold τ, a Receiver Operating Characteristic (ROC) curve is obtained that illustrates the inevitable trade-off between the two error rates (see Eq. 4.17). The Equal Error Rate (EER) corresponds to the point on the ROC curve where $FAR \approx FRR$ and it quantifies in a single number the writer verification performance.

$$FAR = \int_0^{\tau} P_D(x)dx, \qquad FRR = \int_{\tau}^{\infty} P_S(x)dx \tag{4.17}$$

The distributions of chance $P_D(x)$ and $P_S(x)$ represent the distances between PDFs of handwriting samples from various individuals and the same individual, respectively.

For the ICDAR2013 dataset (250 writers, 2 samples per write in case of English language), $P_S(x)$ has been constructed using the 250 same-writer distances, while $P_D(x)$ has been constructed using all the $C_{500}^2 - 250 = 124,500$ different writer distances arising in the dataset.

4.5 Conclusion

This chapter presented the steps necessary to extract discriminative features from handwritten document images. We described the preprocessing pipeline, which includes binarization using Otsu's algorithm, connected component extraction, and the elimination of small components. Two feature extraction methods were proposed: $LCP - IWSL_B$ (black pixels) and $LCP - IWSL_C$ (outer contour pixels). Both methods extract features

from connected components and their outer contours. The next chapter will present the verification system and the experimental results obtained using these features.

Chapter 5

Results and analysis

5.1 Introduction

In this section, we first introduce the experimental setup. This describes eight well-known databases, as well as their enrollment and evaluation protocols. These are used to validate the effectiveness of our proposed writer identification and verification system. We then present and discuss the results of the empirical study in detail, comparing our technique with the state of the art (i.e. the experimental results). All experiments were executed on an HP Z2 Tower G9 Workstation Desktop PC RCTO, equipped with a 12th Gen Intel(R) Core(TM) i9-12900K processor running at 3.19 GHz and 32 GB of RAM, operating on the Windows 10 Pro 64-bit OEM system. Our software, named UMTAWRS (University Moulay Tahar Automatic Writer Recognition System), was developed using Microsoft Visual Studio C++ version 2.0.50727 SP2, and the experimental results are presented in tables and figures.

5.2 Experimental setup

In this section, we evaluated the performance of our proposed method using three publicly available databases containing challenging languages: English, German, French and Greek from the ICDAR 2011 database [36], English and Greek from the ICDAR 2013 database [35], and English and German from the CVL database [31]. These databases are discussed in the following subsections and are designed to be fair and comparable with the literature.

5.2.1 ICDAR2011 database

As detailed in [36], the ICDAR2011 Writer Identification Contest dataset was designed to evaluate writer identification systems in a multilingual context. It comprises contributions from 26 writers, each of whom provided eight handwritten pages, resulting in a total of 208 document images. The dataset encompasses texts in four languages: English, French, German and Greek, with two pages per language and per writer. To ensure script authenticity, the Greek documents were written by native Greek writers, while the other languages were written by the same writers to introduce variability in handwriting across different scripts. Each page contains copied text, providing a consistent evaluation scenario for content across writers and enabling the system to focus on

writer-specific stylistic features. For our experiments, we used a subset of ICDAR2011, called the CICDAR2011 dataset, in which only the first two lines of text on each page were truncated. Two scenarios were conducted to measure the performance of our method: mono-script (see Table 5.1) and multi-script (see Table 5.2).

5.2.2 ICDAR2013 database

The ICDAR2013 [35] benchmarking database was part of the ICDAR2013 writer identification competition. It was collected from 250 different scribes who were asked to copy two pages of handwritten text in English and two in Greek. The number of text lines that are contributed by the scribes varies between two and six lines, and the collected handwritten pages were also scanned at 300 dpi in grayscale. The experiment dataset contains 400 writing pages produced by 100 scribes, whereas the benchmarking dataset consists of 1, 000 pages written by 250 different scribes. In our experiment, we used the entire benchmarking ICDAR2013 database according to the mono-script (see Table 5.1) and multi-script (see Table 5.2).

5.2.3 CVL database

The CVL [31] is a standard offline handwriting English and German text database that is used to evaluate and validate single-/multi-script writer identification and verification, word spotting, and writer retrieval. This database contains a total of 1, 604 document samples that have been written by 310 different writers. Seven documents (i.e. six in English and one in German) are produced by 27 writers. The remaining 282 were asked to write four documents in English and one in German. The texts are scanned in the RGB 24-bit color format at 300 dpi. Therefore, similar to [6], we select only the first four English and the German documents per writer in our experimental setup. Two scenarios were conducted to measure the performance of our method: mono-script (see Table 5.1) and multi-script (see Table 5.2).

Figure 5.1 shows sample images from three benchmark databases containing a variety of script languages. To ensure comparable and fair experimental conditions with state-of-the-art systems, the same experimental setup was used as in Chahi et al. [12] for the ICDAR 2011 database, and as in Bahram et al. [6] for the ICDAR 2013 and CVL databases. The modified benchmark databases used in our writer identification and verification experiments are presented in Tables 5.1 and 5.2.

5.3 Experimental results

In order to assess the efficacy of our proposed system, we performed tests using two texture- and contour-based features ($LCP - IWSL_B$ and $LCP - IWSL_C$), adhering to the methodology detailed in Section 4 and illustrated in Fig. 4.1.

In the field of author identification, we implemented widely recognized standards, particularly the simple, commonly used cross-validation method known as "Holdout," along with the Top1, Top5, and Top10 classification outcomes. These determine the closest neighbor for an unidentified document. We employed these techniques to evaluate and confirm our methodology across all analyzed handwriting databases.

Table 5.1: Description of the tested databases according to the **mono-script protocol**.

Benchmark	Year	# Writers	Language	# samples / writer
CICDAR2011 [36]	2011	26	English French German Greek	2 samples
ICDAR2013 [35]	2013	250	English Greek	2 samples
CVL [31]	2013	310	English	4 samples

Table 5.2: Description of the tested databases according to the **multi-scripts protocol**.

Benchmark	Year	# Writers	Language	# samples / writer
CICDAR2011 [36]	2011	26	English, French, German, and Greek	8 samples
ICDAR2013 [35]	2013	250	English and Greek	4 samples
CVL [31]	2012	310	English and German	5 samples

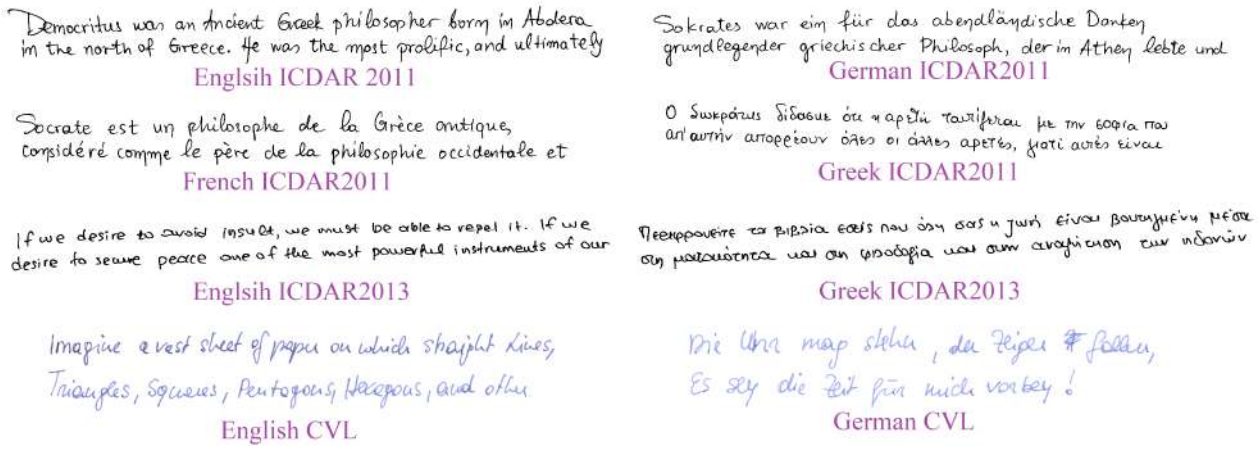


Figure 5.1: Handwriting samples form different databases: ICDAR2011 [36]), ICDAR2013 [35], and CVL [31].

The effectiveness of the writer verification task is evaluated by using a global threshold on the output scores of the k-NN classifier. The False Acceptance Rate (FAR) and False Rejection Rate (FRR) are computed by varying the decision threshold τ. FAR represents samples belonging to non-genuine writers that are assigned to genuine writers by the system. Similarly, the FRR represents scenarios in which writing samples belonging to genuine writers are rejected by the system.

The overall performance of the system is quantified by calculating the Equal Error Rate (EER), which is determined by selecting the pair (FAR, FRR) with the smallest

absolute difference and computing their average. Furthermore, a Receiver Operating Characteristic (ROC) curve is obtained that illustrates the inevitable trade-off between the two error rates. A lower EER ($FAR \approx FRR$) value signifies a greater accuracy of the system performance. In general, a false acceptance is much more unacceptable as opposed to false rejections especially for a biometric system that requires a high level of security. Hence, for a good biometric system, it is desirable that the GAR should increase rapidly against the FAR value. It is important to note that for very small values of FAR, GAR achieves very high percentages demonstrating the effectiveness of the proposed system in scenarios where false acceptances are costly.

5.3.1 Evaluation of mono-script performance

Table 5.3: Performance of writer identification and verification on the ICDAR2011, ICDAR2013, and CVL databases according to the **mono-script protocol**.

Feature	Database	Language	Identification			Verification	
			Top1	Top5	Top10	EER	θ_{opt}
$LCP\text{-}IWSL_B$	CICDAR2011	English	81.62	96.15	100.0	11.53	0.109
$LCP\text{-}IWSL_C$			84.62	100.0	100.0	7.69	0.117
$LCP\text{-}IWSL_B$		French	88.46	100.0	100.0	4.15	0.085
$LCP\text{-}IWSL_C$			88.46	100.0	100.0	4.57	0.104
$LCP\text{-}IWSL_B$		German	84..62	96.15	100.0	3.84	0.087
$LCP\text{-}IWSL_C$			92.31	100.0	100.0	3.84	0.101
$LCP\text{-}IWSL_B$		Greek	57.69	100.0	100.0	14.53	0.101
$LCP\text{-}IWSL_C$			69.23	100.0	100.0	11.53	0.113
$LCP\text{-}IWSL_B$	ICDAR2013	English	96.40	99.60	100.0	1.60	0.052
$LCP\text{-}IWSL_C$			99.60	100.0	100.0	1.19	0.061
$LCP\text{-}IWSL_B$		Greek	97.60	100.0	100.0	1.19	0.047
$LCP\text{-}IWSL_C$			100.0	100.0	100.0	1.15	0.058
$LCP\text{-}IWSL_B$	CVL	English	96.77	98.38	98.38	3.21	0.062
$LCP\text{-}IWSL_C$			97.74	98.38	98.38	2.94	0.072

To establish a performance baseline for the proposed writer identification and verification system, we will first evaluate its performance using monoscript datasets. More specifically, we will evaluate its performance on the English, French, German, and Greek subsets of the ICDAR 2011 dataset, the English and Greek subsets of the ICDAR 2013 dataset, and the English and German subsets of the CVL 2013 dataset. This phase independently verifies the effectiveness of the contour- and texture-based features ($LCP\text{-}IWSL_B$ and $LCP\text{-}IWSL_C$), isolating the impact of characteristics specific to the script.

The goal is to evaluate the system's performance while maintaining consistency within scripts and introducing variability across them in multilingual conditions. Table 5.3 shows the Top1, Top5, and Top10 recognition rates for the writer identification task as determined by the standard holdout validation protocol. For the writer verification task, we calculate the equal error rate (EER) by applying a range of thresholds (θ) to the similarity scores of handwritten samples.

Table 5.3 shows how robust and discriminative the proposed $LCP-IWSL$ contour- and texture-based features are across all four scripts (English, French, German, and Greek). The holdout validation strategy and Chi-squared (χ^2) distance consistently yield the highest accuracy scores (i.e., identification and verification). These results confirm the suitability of the chi-squared distance for the proposed descriptors, thus validating its adoption in our approach.

It is important to note that the $LCP-IWSL_C$ feature, which is derived from outer contour of connected components, generally outperforms the $LCP-IWSL_B$ feature. This is particularly evident in Top1, Top5, Top10, and EER recognition accuracies. This highlights the effectiveness of the $LCP-IWSL_C$ feature in capturing stylistic nuances. These results demonstrate internal consistency across scripts and tasks. This affirms the competitiveness of the proposed system against current offline writer identification and verification methods (see Bahram's work [6]). The results support deploying the system in real-world mono-script applications, where script variability and generalization are critical.

5.3.2 Evaluation of multi-scripts performance

This section presents the experiments performed to study the effectiveness of the proposed features for writer recognition and to validate the hypothesis that the writing style of an individual remains more or less the same across different scripts. The performance measures used are the Top1, Top5, Top10 identification rates and the Equal-Error-Rate (EER) for verification task.

Table 5.4 shows the performance of $LCP-IWSL$ features on the ICDAR2011 8-samples, ICDAR2013 4-samples, and CVL 2-samples cropped multi-scripts datasets using chi-squared distance. The Top1 accuracy is highest for ICDAR2011 and ICDAR2013 (100.0% in hold-out strategy and EER=7.68%), followed by CVL (Top1=93.55% and EER=3.5%). χ^2 consistently improves performance over individual $LCP-IWSL_C$ features, confirming its effectiveness in multi-script identification.

Table 5.4: Performance of writer identification and verification on the ICDAR2011, ICDAR2013, and CVL databases according to the **multi-scripts protocol**.

Feature	Database	Language	Identification			Verification	
			Top1	Top5	Top10	EER	θ_{opt}
LCP-$IWSL_B$	CICDAR2011	All four,	96.15	100.0	100.0	8.82	0.100
LCP-$IWSL_C$		languages	100.0	100.0	100.0	7.68	0.114
LCP-$IWSL_B$	ICDAR2013	English &	96.15	100.0	100.0	8.82	0.100
LCP-$IWSL_C$		Greek	100.0	100.0	100.0	7.68	0.114
LCP-$IWSL_B$	CVL	English &	92.25	96.45	97.10	3.78	0.065
LCP-$IWSL_C$		German	93.55	97.10	98.39	3.50	0.075

5.4 Conclusion

Very satisfactory results were obtained in all cases on both mono-script and multi-scripts document databases (ICDAR2011, ICDAR2013, and CVL). Our proposed approach places itself among the best in the literature.

Chapter 6

Conclusions and perspectives

The conversion of unprocessed data into either manually crafted or learned features is a vital component of author recognition. Both traditional methods and deep convolutional neural networks (CNNs) require considerable time, memory, and computational power. Additionally, while approaches that employ learned features have shown promise, they remain relatively under-researched in the domain of author recognition tasks, particularly in handwriting verification. In this study, we present a method for offline text-independent author recognition from handwriting images, leveraging machine learning and pattern recognition algorithms. Furthermore, these methods do not necessitate extensive computational resources regarding time or memory consumption. This approach utilizes two primary sources of behavioral data to define author individuality. Initially, the $LCP - IWSL_B$ histograms (which are based on texture of connected components) can deliver comprehensive information about letter shapes and ink traces, encompassing ink thickness. Secondly, $LCP - IWSL_C$ distributions (which are based on outer contours) provide further insights into the distribution of direction, slant, and curvature—essentially, the attributes of writing. The proposed handcrafted features comprise probability distributions derived from the binary connected components and outer contours of scanned samples. These distributions present a robust and text-independent representation of an individual's writing style. Importantly, the classification accuracy obtained with the outer contour pixels feature $LCP - IWSL_C$ exceeds that of the black pixels $LCP - IWSL_B$. Our research, carried out on three challenging handwriting datasets, aims to further substantiate these results. The primary limitation of this system is its inability to process very short texts, such as a single word. Consequently, to ensure the reliability of the proposed features, a significant number of lines must be examined (approximately three lines of text for both training and testing purposes). Regarding future research, it is advisable to evaluate the effectiveness of these descriptors in gender classification as well as in the identification and verification of writers across various scripts, particularly when the training and testing datasets are derived from different scripts. Furthermore, the proposed methodology could be improved to integrate handwriting style information through handwritten word images, and it is crucial to develop strategies for recognizing unknown writers or registering new ones.

Bibliography

[1] M.N. Abdi and M. Khemakhem. A model-based approach to offline text-independent arabic writer identification and verification. *Pattern Recognition*, 48(5):1890–1903, 2015.

[2] B. Arazi. Handwriting identification by means of run-length measurements. *IEEE Transactions on Systems, Man, and Cybernetics*, 7(12):878–881, 1977.

[3] V. Atanasiu, L. Likforman-Sulem, and N. Vincent. Writer retrieval - exploration of a novel biometric scenario using perceptual features derived from script orientation. In *2011 International Conference on Document Analysis and Recognition*, pages 628–632, Beijing, China, 2011.

[4] T. Bahram. A texture-based approach for offline writer identification. *Journal of King Saud University - Computer and Information Sciences*, 34(8, Part A):5204–5222, 2022.

[5] T. Bahram. Offline writer identification using lcp-iwsl and lblp descriptors for detecting texture information. *Multimedia Tools and Applications*, 84:40091–40120, 2025.

[6] T. Bahram. An improved text-independent writer recognition framework using handcrafted descriptors. *Journal of King Saud University - Computer and Information Sciences*, 2026.

[7] T. Bahram, A. Benyettou, and D. Ziadi. A set of features for text-independent writer identification. *International Review on Computers and Software (I.RE.CO.S.)*, 56(2):898–906, 2016.

[8] A. Bennour, C. Djeddi, A. Gattal, I. Siddiqi, and T. Mekhaznia. Handwriting based writer recognition using implicit shape codebook. *Forensic Science International*, 301:91–100, 2019.

[9] D. Bertolini, L.S. Oliveira, E. Justino, and R. Sabourin. Texture-based descriptors for writer identification and verification. *Expert Systems with Applications*, 40(6):2069–2080, 2013.

[10] A. Briber and Y. Chibani. Open writer identification from handwritten text fragments using lite convolutional neural network. *International Journal on Document Analysis and Recognition (IJDAR)*, 27(4):529–551, 2024.

[11] M. Bulacu and L. Schomaker. Text-independent writer identification and verification using textural and allographic features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):701–717, 2007.

[12] A. Chahi, Y. El merabet, Y. Ruichek, and R. Touahn. Local gradient full-scale transform patterns based off-line text-independent writer identification. *Applied Soft Computing*, 92:106277, 2020.

[13] A. Chahi, Y. El Merabet, Y. Ruichek, and R. Touahni. Writerinet: a multi-path deep CNN for offline text-independent writer identification. *International Journal on Document Analysis and Recognition (IJDAR)*, 26(2):89–107, 2023.

[14] V. Christlein, D. Bernecker, A.K. Maier, and E. Angelopoulou. Offline writer identification using convolutional neural network activation features. In J. Gall, P.V. Gehler, and B. Leibe, editors, *Pattern Recognition - 37th German Conference, GCPR 2015*, pages 540–552, Aachen, Germany, 2015. Springer.

[15] C. Djeddi, I. Siddiqi, L.S. Meslati, and A. Ennaji. Text-independent writer recognition using multi-script handwritten texts. *Pattern Recognition Letters*, 34(10):1196–1202, 2013.

[16] A. Durou, I. Aref, S. Al-Maadeed, A. Bouridane, and E. Benkhelifa. Writer identification approach based on bag of words with obi features. *Information Processing & Management*, 56(2):354–366, 2019.

[17] S. Fiel and R. Sablatnig. Writer identification and retrieval using a convolutional neural network. In G. Azzopardi and N. Petkov, editors, *Computer Analysis of Images and Patterns - 16th International Conference, CAIP 2015*, pages 26–37, Valletta, Malta, 2015. Springer.

[18] R.C. Gonzalez and R.E. Woods. *Digital image processing*. Prentice Hall (Addison-Wesley, 3rd edition), Upper Saddle River, N.J., 2008.

[19] Y. Hannad, I. Siddiqi, C. Djeddi, and M.E. El-Kettani. Improving arabic writer identification using score-level fusion of textural descriptors. *IET Biometrics*, 8(3):221–229, 2019.

[20] Y. Hannad, I. Siddiqi, and M.E. El-Kettani. Writer identification using texture descriptors of handwritten fragmen. *Expert Systems with Applications*, 47:14–22, 2016.

[21] S. He and L. Schomaker. Co-occurrence features for writer identification. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 78–83, Shenzhen, China, 2016. IEEE.

[22] S. He and L. Schomaker. Writer identification using curvature-free features. *Pattern Recognition*, 63:451–464, 2017.

[23] S. He and L. Schomaker. FragNet: Writer identification using deep fragment networks. *IEEE Transactions on Information Forensics and Security*, 15:3013–3022, 2020.

[24] S. He and L. Schomaker. GR-RNN: global-context residual recurrent neural networks for writer identification. *Pattern Recognition*, 117:107975, 2021.

[25] S. He, M. Wiering, and L. Schomaker. Junction detection in handwritten documents and its application to writer identification. *Pattern Recognition*, 48(12):4036–4048, 2015.

[26] M. Javidi and M. Jampour. A deep learning framework for text-independent writer identification. *Engineering Applications of Artificial Intelligence*, 95:103912, 2020.

[27] Y. Kessentini, S. BenAbderrahim, and C. Djeddi. Evidential combination of svm classifiers for writer recognition. *Neurocomputing*, 313:1–13, 2018.

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

[29] F.A. Khan, F. Khelifi, and M.A. Tahir. Dissimilarity gaussian mixture models for efficient offline handwritten text-independent identification using sift and rootsift descriptors. *IEEE Transactions on Information Forensics and Security*, 14(2):289–303, 2019.

[30] F.A. Khan, M.A Tahir, F. Khelifi, A. Bouridane, and R. Almotaeryi. Robust off-line text independent writer identification using bagged discrete cosine transform features. *Expert Systems with Applications*, 71:404–415, 2017.

[31] F. Kleber, S. Fiel, M. Diem, and R. Sablatnig. CVL-Database: An off-line database for writer retrieval, writer identification and word spotting. In *2013 12th International Conference on Document Analysis and Recognition*, pages 560–564, Washington, DC, USA, 2013. IEEE.

[32] P. Kumar and A. Sharma. Segmentation-free writer identification based on convolutional neural network. *Computers & Electrical Engineering*, 85:106707, 2020.

[33] V. Kumar and S. Sundaram. Siamese-based offline word level writer identification in a reduced subspace. *Engineering Applications of Artificial Intelligence*, 130:107720, 2024.

[34] V. Kumar and S. Sundaram. Utilization of information from CNN feature maps for offline word-level writer identification. *Expert Systems With Applications*, 238(Part A):121709, 2024.

[35] G. Louloudis, B. Gatos, N. Stamatopoulos, and A. Papandreou. ICDAR 2013 competition on writer identification. In *2013 12th International Conference on Document Analysis and Recognition*, pages 1397–1401, Washington, DC, USA, 2013. IEEE.

[36] G. Louloudis, N. Stamatopoulos, and B. Gatos. ICDAR 2011 writer identification contest. In *2011 International Conference on Document Analysis and Recognition, ICDAR 2011*, pages 1475–1479, Beijing, China, 2011. IEEE.

[37] T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, 1996.

[38] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[39] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

[40] L. Schomaker. Advances in writer identification and verification. In *9th International Conference on Document Analysis and Recognition (ICDAR 2007)*, pages 1268–1273. IEEE Computer Society, 2007.

[41] A. Semma, Y. Hannad, I. Siddiqi, C. Djeddi, and M.E. El-Kettani. Writer identification using deep learning with fast keypoints and harris corner detector. *Expert Systems with Applications*, 184:115473, 2021.

[42] I. Siddiqi and N. Vincent. Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features. *Pattern Recognition*, 43(11):3853–3865, 2010.

[43] P. Singh, P. Pratim-Roy, and B. Raman. Writer identification using texture features: A comparative study. *Computers & Electrical Engineering*, 71:1–12, 2018.

[44] Y. Sun, D. Fan, H. Wu, Z. Wang, and J. Tian. Offline mongolian handwriting identification based on convolutional neural network. *Electronics*, 13(1):111, 2024.

[45] 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):526–536, 2014.

[46] Y.J. Xiong, Y. Wen, P.S.P. Wang, and Y. Lu. Text-independent writer identification using SIFT descriptor and contour-directional feature. In *13th International Conference on Document Analysis and Recognition, ICDAR 2015*, pages 91–95. IEEE, 2015.

[47] J. Yang, M. Shokouhifar, P.L. Yee, A.A. Khan, M. Awais, and Z. Mousavi. DT2F-TLNet: A novel text-independent writer identification and verification model using a combination of deep type-2 fuzzy architecture and transfer learning networks based on handwriting data. *Expert Systems with Applications*, 242:122704, 2024.

[48] B. Zhao, X. Cao, W. Zhang, X. Liu, Q. Miao, and Y. Li. Compnet: Boosting image recognition and writer identification via complementary neural network post-processing. *Pattern Recognition*, 157:110880, 2025.

ملخص

تتناول الأعمال المقدَّمة في هذه الرسالة مشكلة **التعرّف الآلي على الكاتب من خلال الوثائق المكتوبة بخط اليد غير المتصلة (offline)**. وتُعد مرحلة **استخراج السمات** من أهم المراحل في أنظمة التعرّف على الكاتب، إذ تهدف إلى اختيار أكثر المعلومات تمييزًا وملاءمةً لمهمة التصنيف.

في هذه الرسالة، تُمثَّل عينات الكتابة اليدوية بمجموعة من السمات، تشمل اتجاه الضربات، والانحناء، وسُمك الخطوط، وحجم الكتابة. وتعتمد المنهجية المقترحة على تقنية مبتكرة لاستخراج السمات، تسهم في تحسين دقة كلٍّ من **تحديد هوية الكاتب (Writer Verification)** و**التحقق من هوية الكاتب (Writer Identification)**. ويُستخدم واصفان أساسيان، هما **LCP–IWSL$_B$** (البكسلات السوداء) و **LCP–IWSL$_C$** (بكسلات المحيط الخارجي)، بصورة مستقلة لتمثيل الخصائص الفردية المميزة لكل كاتب. وتُنفَّذ عملية التصنيف باستخدام خوارزمية **(k-Nearest Neighbors, k-NN)** بالاعتماد على **مسافة (Chi-Square, χ^2)** كمقياس للتشابه. وقد جرى تقييم المنهجية المقترحة باستخدام ثلاث قواعد بيانات للوثائق المكتوبة بخط اليد تضم عينات تعود إلى 26 و250 و310 كاتبًا، حيث حققت نتائج واعدة.

وتُظهر النتائج التجريبية متانة المنهجية المقترحة في كلٍّ من سيناريوهات الكتابة أحادية النص ومتعددة النصوص، ولا سيما في البيئات الهجينة التي تحاكي ظروف الكتابة متعددة اللغات في التطبيقات الواقعية. كما يتميز النظام باستقرار أدائه وكفاءته حتى في ظل وجود تنوع أكبر في البيانات.

الكلمات المفتاحية: التعرف على الكاتب و التحقق من هويته, نمط الحواف المحلي ,أثر الحبر و شكل الحرف ,التعلم الآلي, التعرف على الأنماط, المعالجة الشرعية للوثائق, الكتابة اليدوية.

Abstract

The work presented in this manuscript addresses the problem of **offline writer recognition** based on handwritten documents. The feature extraction stage is a crucial component of any writer recognition system, as its objective is to select the most discriminative information for the classification task. In this thesis, handwritten samples are represented using several attributes, including stroke orientation, curvature, stroke thickness, and handwriting size. The proposed method incorporates an innovative feature extraction technique that improves the accuracy of both **writer identification** and **writer verification**. Two key descriptors, **LCP–IWSL$_B$** (black pixels) and **LCP–IWSL$_C$** (outer contour pixels), are employed independently to characterize the individuality of each writer. Classification is performed using the **k-Nearest Neighbors (k-NN)** classifier in conjunction with the **Chi-Square (χ^2)** distance measure. The proposed approach was evaluated on three handwritten document databases containing samples from 26, 250, and 310 writers, respectively, where promising results were obtained. The experimental results demonstrate the robustness of the proposed method in both single-script and multi-script scenarios, particularly in hybrid environments that simulate real-world multilingual handwriting conditions. The system exhibits stable and reliable performance even in the presence of greater data diversity.

Keywords: Writer identification and verification, Local contour pattern, IWSL: Ink Trace-Width and Shape Letter, Machine Learning, Forensic science, Handwriting.

Résumé

Les travaux présentés dans ce manuscrit abordent le problème de reconnaissance de scripteurs à partir de leur écriture manuscrite hors-ligne. La phase d'extraction de caractéristiques est une étape très importante pour un système de reconnaissance de scripteurs. L'objectif de cette étape est la sélection des informations les plus pertinentes pour la tâche de classification. Dans ce mémoire les échantillons d'écritures manuscrites sont représentés par des attributs comme l'orientation, la courbure, l'épaisseur des traits et la taille de l'écriture. La méthode proposée intègre une technique innovante d'extraction de caractéristiques qui améliore l'exactitude de l'identification et de la vérification. Deux descripteurs clés, LCP −IWSLB (pixels noirs) et LCP −IWSLC (pixels de contour externe), sont utilisés indépendamment pour caractériser l'individualité du scripteur. La classification est réalisée en utilisant les k plus proches voisins (k-NN) et la distance de Chi-Square (χ2). La méthode proposée a été évaluée en utilisant trois bases de documents de 26, 250 et 310 scripteurs où des résultats intéressants ont été enregistrés. Nos résultats montrent la robustesse de la méthode proposée dans les scénarios mono- et multi-script, en particulier dans les environnements hybrides qui simulent les conditions d'écriture multilingue dans le monde réel. Le système se montre stable et performant, même en présence d'une plus grande diversité de données.

Les mots clés: Identification et vérification de scripteurs, Modèle de Contour Local, IWSL: Epaisseur de trait Forme de Lettres, Apprentissage automatique, Examen légal des documents, Écriture manuscrite.