# Dissimilarity Gaussian Mixture Models for Efficient Offline Handwritten Text-Independent Identification using SIFT and RootSIFT Descriptors

Faraz Ahmad Khan, Fouad Khelifi, *Member, IEEE*, Muhammad Atif Tahir, Member IEEE, and Ahmed Bouridane, *Senior Member, IEEE*

***Abstract*—Handwriting biometrics is the science of identifying the behavioural aspect of an individual's writing style and exploiting it to develop automated writer identification and verification systems. This paper presents an efficient handwriting identification system which combines Scale Invariant Feature Transform (SIFT) and RootSIFT descriptors in a set of Gaussian mixture models (GMM). In particular, a new concept of similarity and dissimilarity Gaussian mixture models (SGMM and DGMM) is introduced. While a SGMM is constructed for every writer to describe the intra-class similarity that is exhibited between the handwritten texts of the same writer, a DGMM represents the contrast or dissimilarity that exists between the writer's style on one hand and other different handwriting styles on the other hand. Furthermore, because the handwritten text is described by a number of key point descriptors where each descriptor generates a SGMM/DGMM score, a new weighted histogram method is proposed to derive the intermediate prediction score for each writer's GMM. The idea of weighted histogram exploits the fact that handwritings from the same writer should exhibit more similar textual patterns than dissimilar ones, hence, by penalizing the bad scores with a cost function, the identification rate can be significantly enhanced. Our proposed system has been extensively assessed using six different public datasets (including three English, two Arabic and one hybrid language) and the results have shown the superiority of the proposed system over state-of-the-art techniques.**

***Index Terms*—Writer identification, text independent, Gaussian Mixture Model, dissimilarity framework, weighted histograms, SIFT, RootSIFT.**

## I. INTRODUCTION

THE identification of people through biometric traits can be referred to as determining the identity of a person through his or her unique traits, which can be divided into (i) behavioural and (ii) physiological traits [1]. Physiological traits include fingerprints, face or iris i.e. using direct measurements of parts of the human body. On the other hand, behavioural traits accomplish the task of identification using data acquired from actions performed by the person.

Writer identification and verification can be carried out using an individual's handwriting. Recent research has shown handwriting to be a very reliable and useful behavioural biometric tool which can be used in various application domains including forensic analysis [2], analysis of historical documents [3], [4] and security [5].

The challenge of automatic hand writer identification can be divided into either text dependent or text independent approaches [6]. In text dependent approaches the identity of a writer is determined using a specific transcript and usually the writer is asked to reproduce a sample of text. Therefore, in text dependent systems, the availability of the writer is assumed. Signature identification or authentication is one example of a text dependent system [7]. Text independent systems, on the other hand, aim to identify the identity of a writer regardless of the written text. Generally, text dependent systems are more accurate. However, they assume the exact same text to be reproduced for accurate identification along with the availability of the writer thus making it less practical. Text independent systems, on the other hand, are more practical as they do not depend on the exact same content; however, they do require a large amount of data from every writer in order to properly train the system.

Writer recognition can be deployed in two modes: verification or identification. In writer verification (authentication) a system is presented with two samples of text and it must determine whether or not the two samples have been written by the same writer. On the other hand, in writer identification a system is presented with an unknown sample of text which is then used by the system and compared to a database of known writers. After the comparison process, the system should present a list of likely authors of the unknown sample of text. These two tasks can be performed either offline or online. In an offline mode, identification is performed using only a scanned image whereas in online recognition mode, the system has access to a set of additional dynamic features such as the slant of the pen, its speed, pressure and height [8]. Due to the availability of these extra features online writer identification systems are expected to perform better than offline identification counterparts.

This paper proposes an offline writer identification system that relies on a similarity and dissimilarity Gaussian mixture model approach using a weighted histogram of GMM scores. SIFT and RootSIFT [9] descriptors, which are used to represent handwritten text data are employed to construct a set of SGMMs and a DGMM for every writer (as will be explained in Section III-C). In this context, for every writer, two SGMMs are generated to describe the intra-class similarity that is exhibited between handwritten texts of a writer using SIFT and RootSIFT descriptors. On the other hand, the DGMM represents the dissimilarity which inherently exists between that writer's style and other different handwritings of other different hand writers styles using SIFT technique. While the SGMM/DGMM approach leads to multiple scores for a single handwritten text generated from key point-based descriptors, a new weighted histogram method is introduced to efficiently derive intermediate prediction scores for each writer's GMM. The proposed system has been evaluated using six publicly available datasets including multiple languages: three English, two Arabic and one Hybrid language. A comparative analysis against state-of-the-art systems has been carried out to validate the proposed approach.

The main contributions of this work can be summarised as: (i) Dissimilarity Gaussian Mixture Models are introduced. Along with two similarity GMMs created for every writer, a DGMM is also constructed for describing the contrast between different writers in the database. The combination of DGMM with SGMM has been shown to bring significant improvements over the overall identification date. (ii) Because a handwritten text is described by a number of key point descriptors where each descriptor has its own SGMM/DGMM contribution, a new weighted histogram method is proposed to derive the intermediate prediction score from the set of individual key point contributions. The idea of the weighted histogram relies on the fact that handwritings of the same writer should exhibit more similar textual patterns than dissimilar ones. Therefore, by penalizing the bad contributions with a cost function, the identification rate can be significantly enhanced. (iii) SIFT and RootSIFT descriptors are effectively exploited in a joined SGMM/DGMM system where a simple but efficient score fusion method has been proposed accordingly.

The remainder of this paper is organised as follows. Section II overviews the related work done in the field of writer identification. Section III describes our proposed method in detail. Section IV provides an experimental analysis of our proposed system along with a discussion of our results achieved. And finally, Section V discusses the conclusions drawn from our work.

## II. Related Work

Writer identification has been extensively researched and various approaches have been proposed. Said et al. [10] extracted features from handwritten samples using co-occurrence matrices and Gabor filters. Nejad and Rahmati [11] proposed to extract features using the output of Gabor filters using moments and non-linear transformations. Helli and Maghoddam [12] also utilized Gabor filters for the purpose of feature extraction and extended the concept by using a Longest Common Subsequence (LCS) based classifier. The same authors extended their previous work by using a Feature Relation Graph (FRG) to represent the Gabor features [13]. The FRG was constructed for each writer using a fuzzy method where the identification was performed using a graph similarity approach.

Schlapbach and Bunke [14] proposed the use of Hidden Markov Model (HMM) for the purpose of writer identification. For each writer, three global and six local (total of 9) features were extracted using a sliding pixel wide window and used to train a single HMM recognizer to determine the authorship of the written text. The same authors further improved their previous work by using a GMM approach instead of a HMM method [15]. The same principle was applied where each writer is modelled by its corresponding GMM recognizer and the authors concluded that GMM outperforms HMM.

Marti et al. [16] utilized visual identifying features such as the height, slant, skew and width of the handwriting. The authors extracted 12 such characteristics from every writer. Likewise, the same features were also extracted by Sadeghi and Moghaddam [17] but at a grapheme and gradient level. These features were then clustered using a fuzzy clustering method.

Siddiqi and Vincent [18] proposed a system which utilized both the global and local features of handwriting. They proposed a two step approach by first extracting global features by localizing the oriented segments using Gabor filters followed by a direction criterion to determine the handwriting class. Then the query document was identified using local features by searching only in the specific class that was determined previously. The same authors later proposed to extract the identifying information from the contours of handwriting at local and global levels [19]. The authors further improved their previous work in [20] by extracting information from the orientation and curvature of the chain code sequence of handwriting. This system improved their previously reported results.

Balacu et al. [21] proposed to classify writers using two sets of features: textural and allograph level characteristics. A Probability Distribution Function (PDF) and a stochastic pattern of graphemes were used to represent these features. Brink et al. [22] demonstrated the importance of width patterns in writer identification. The authors proposed the QuillHinge feature to extract identifying characteristics from the direction of ink traces as well as the width of ink strokes. Bensefia et al. [23] proposed a 'writer invariant' codebook for writer identification. The codebook generated by clustering the graphemes that were produced through segmentation. Identification was performed by comparing the writer invariant codebooks of different samples.

Wu et al. [24] proposed to perform writer identification by using two features: SIFT descriptors (SD) and their corresponding scales and orientations (SO) which were used in different ways during the identification process. At the enrollment stage a scale and orientation histogram (SOH) was also generated from the extracted SOs. The identification was performed by extracting and comparing SDS and SOH of the query images with the enrolled ones. The resulting two matching distances were then fused to obtain the final identification result. Bertolini et al. [25] considered that a handwritten script can be thought of as texture and thus can best be represented by Local Binary Patterns (LBP) and Local Phase Quantization (LPQ). They further expanded upon the idea presented by Hanusiak et al. in [26] and applied their presented idea to writer identification. Bertolini et al. also relied on a dissimilarity framework for the purpose of writer identification. They applied the dissimilarity framework by generating two classes: a positive population and a negative population. These populations were then used to train a binary SVM classifier where the identification was performed by generating dissimilarity vectors of the query image resulting from it's comparison against all the images in the dataset. These dissimilarity vectors were then classified using the trained SVM.

By observing the discriminative power of fragments in writer identification problems, Hannad et al. [27] proposed a writer identification system by extracting textural information from small fragments of text. Handwritten documents were first segmented into small fragments using a window size of 100 x 100 and then fed to three feature extractors: LBP, LTP and LPQ. The authors demonstrated that out of the three descriptors, LPQ gave the best identification results and maintained its accuracy as the number of writers was increased. The handwritten documents were compared and classified by using a simple dissimilarity measure.

## III. Proposed System

The framework for the training phase of our proposed system is illustrated by Fig.1. It consists of three phases: segmentation of handwritten text, feature extraction and generation of identifier models. A detailed explanation of each phase is given below.

### A. Word Segmentation

For hand writer identification, the features can be extracted from either the allographs within the words, the full words or from the full page. However, for the purpose of handwriting analysis, the features extracted at the word level are much more effective than those extracted at a page or allograph level [24]. This is because the features extracted from a full page may include other features that are detected between the words and lines, as shown in Fig.2(a). These features contain no relevant information and are in fact detrimental to the identification procedure. The segmentation of words at this stage does not have to be very accurate and segmentation failures would not have a significant impact on the performance of the system as this step is performed to filter out the excessive keypoints that contain no information. Fragment and allograph level features are extracted at a sub-word level and may lead to many stable identifying features of the writer to be missed. The features extracted at a word level perform well because only strong and valid identifying features are extracted as exemplified by in Fig.2.

The extraction of words from a document is achieved by using a mask on the original grey scale image. To obtain this mask, the input image is subjected to a binarization process using Otsu's method [28]. After the binarization process, a morphological closing, which is a process of performing dilation followed by an erosion of the image, is performed on the logical image using a disk structuring element of radius 9. The structuring element is kept at a certain size to ensure that the gaps between letters of words are closed but those between words are still distinguishable. This ensures that the individual words are extracted and the white gaps between words are ignored. Once this step is done, the extraction of the words can be carried out using a bounding box. The area of each bounding box is then used to determine and remove the diacritics, commas and periods since they do not carry any information related to the writing style of the writer. The bounding box coordinates allows for the extraction of the connected components which are used as masks on the original image to extract only the sample text words while ignoring all the white spaces. This procedure is shown in Fig.3.

### B. Feature Extraction

SIFT method was first proposed by Lowe in [29] and has been successfully applied in many fields due to its capability to extract very distinctive and scale invariant features from images [30]. SIFT is usually the preferred method of feature extraction in applications such as object retrieval and object detection. In particular, a writer can generate text with varying scale, orientation and translation. Furthermore, the different scanning procedures of the documents can cause variations in the illumination for the training and query documents. Therefore, these challenges should be taken care of by the feature extraction method to ensure robustness against such variations in order to provide a reliable result. SIFT has proven to be an efficient method to address these geometric distortions as demonstrated in the field of writer identification [31], [32], [24], [33]. SIFT algorithm operates in four stages. First, an image is broken down into a Gaussian pyramid of octaves where the original image is then convolved with its corresponding octaves of the pyramid with difference of Gaussian (DoG) filters at different variances. In the next stage, referred to as key point localization, the stable key points are detected. Then the orientations, scales and locations of these key points are calculated. Finally, 128 dimension descriptors are generated to represent the image features. This is based on the histogram of oriented gradients (HoG).

In addition to SIFT we have also employed RootSIFT [9] for the purpose of feature extraction. RootSIFT and SIFT follow the same principle for the extraction of the features with the only difference being that SIFT uses an Euclidean distance for similarity measurement while RootSIFT uses the Hellinger kernel. By using the Hellinger kernel instead of the Euclidean one, significant performance improvements can be obtained [9]. This is due to the fact that the Euclidean distance is much less efficient than the Hellinger kernel for comparison of histograms.

Let us analyse the connection between the Hellinger kernel and the Euclidean distance kernel in SIFT. Let $x$ and $y$ to be two feature vectors having a unit Euclidean normalization, i.e. $\|x\|_2 = 1$. Therefore, it follows, the relationship between the Euclidean distance, $d_E(x, y)$ and their similarity kernel is given as

$$\begin{aligned} d_E(x,y)^2 &= \|x-y\|_2^2 = \|x\|_2^2 + \|y\|_2^2 - 2x^T y \\ &= 2 - 2S_E(x,y) \end{aligned} \tag{1}$$

where $S_E(x, y) = x^T y$ and $\|x\|_2^2 = \|y\|_2^2 = 1$.

To convert SIFT to RootSIFT the Euclidean kernel need be replaced with the Hellinger kernel. The Hellinger kernel is defined as [34]:

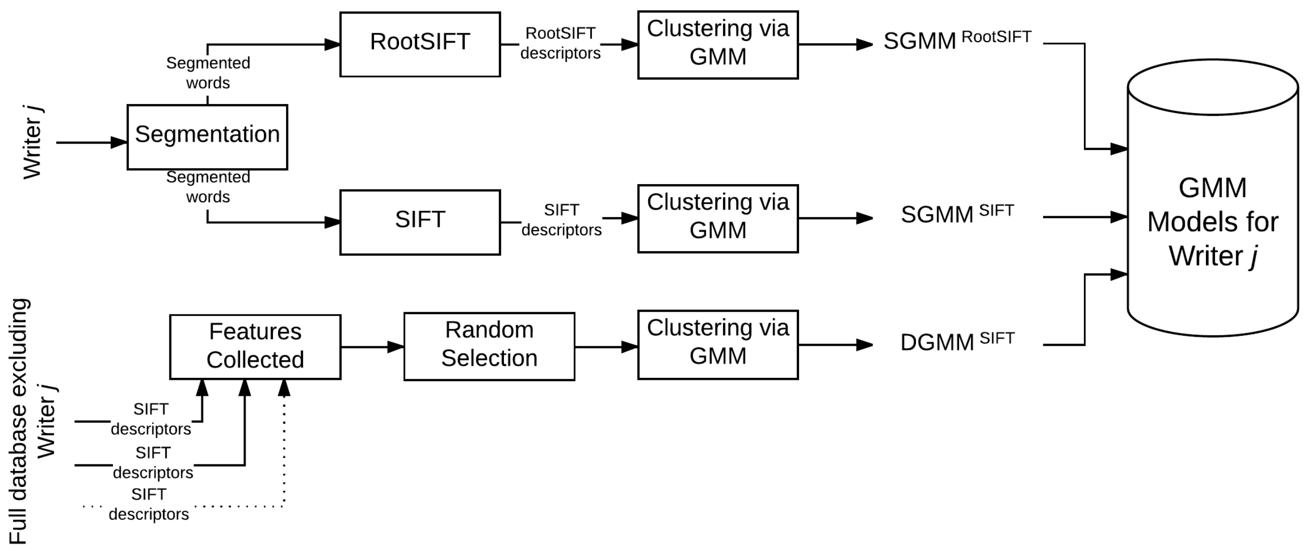
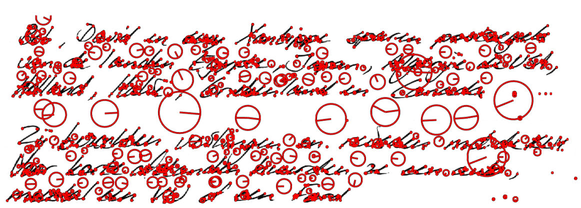


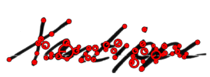

Fig. 1: SGMM and DGMM generation for writer *j*.



(a)

(b)

(c)



(d)

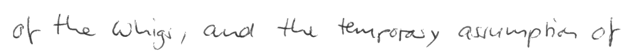
(e)

Fig. 2: (a) SIFT feature extraction on an unsegmented document from the Firemaker database showing invalid keypoints. (b) (c) (d) (e) Only valid SIFT keypoints extracted after word segmentation.

$$H(x, y) = \sum_{i=1}^{n} \sqrt{x_i y_i} \tag{2}$$

where $x$ and $y$ represent two $L_1$-normalized histograms, i.e. $\sum_i^n x_i = 1$ , $x_i \geq 0$.

Therefore, to perform a similarity measure between two SIFT descriptors using the Hellinger kernel, two algebraic




(a)

(b)

(c)

(d)

(e)

Fig. 3: Segmentation and word extraction procedure. (a) An image from the IAM data set in it's original form. (b) The same image after being subjected to binarization and morphological closing. (c) (d) (e) The extracted words.

operations must be followed, (i) perform an $L_1$ normalization of the SIFT descriptor and (ii) perform an element wise square root operation on the normalized SIFT vector. Therefore,

$$S_E(\sqrt{x}, \sqrt{y}) = \sqrt{x}^T \sqrt{y} = \sum_{i=1}^{n} \sqrt{x_i y_i} = H(x, y) \tag{3}$$

where

$$S_E(\sqrt{x}, \sqrt{x}) = \sqrt{x}^T \sqrt{x} = \sum_{i}^{n} x_i = 1 \tag{4}$$

At this stage, the SIFT descriptors have been converted to RootSIFT and as such, comparing these RootSIFT descriptors using the Euclidean distance will have the same effect as comparing original SIFT vectors via the Hellinger kernel, i.e.

$$d_E(\sqrt{x}, \sqrt{y})^2 = 2 - 2H(x, y) \tag{5}$$

By following this procedure, the benefits of using the Hellinger kernel on SIFT descriptors can be exploited without altering the original script used to generate the SIFT vectors. As a result, SIFT can be simply replaced with RootSIFT at every point of the algorithm. It is worth noting that each segmented word in the handwritten text document provides a number of SIFT and RootSIFT descriptors. Therefore, the total number of SIFT and RootSIFT descriptors extracted from each text document varies depending on the number of segmented words as well as the number of key points detected on each word.

### C. Similarity and Dissimilarity Gaussian Mixture Models

GMM's have been widely used and successfully applied in the field of speech recognition [35], [36]. In this work, a GMM models the distribution of the feature vectors extracted from an individual's handwritten text by a multivariate Gaussian mixture distribution [36]. This model is then used to estimate the probability that a certain handwritten text image corresponds to that individual's handwriting style. Given a feature vector $x$ with $D$ random variables representing an individual's handwriting style, the multivariate Gaussian mixture density conditioned on a set of parameters $\lambda_j$ for the $j^{th}$ writer ($j \in \{1, 2, \cdots, N\}$) is defined as

$$p(x|\lambda_j) = \sum_{i=1}^{M} \phi_i^j \; \aleph(x|\mu_i^j, C_i^j)\,, \tag{6}$$

where $\aleph(x|\mu_i^j, C_i^j)$ stands for the multivariate Gaussian function with mean vector $\mu_i^j \in \Re^{D\times 1}$ and covariance matrix $C_i^j \in \Re^{D\times D}$. $\phi_i^j$ is the weight corresponding to the $i^{th}$ Gaussian where $\sum_{i=1}^{M} \phi_i^j = 1$. Here, the parameters of the GMM, i.e., $\lambda_j = \{\mu_i^j, C_i^j, \phi_i^j\}$, are estimated via the Expectation Maximization (EM) algorithm [37] in an iterative fashion. Once the parameters $\lambda_j$ are estimated for writer $j$, the model can be used to calculate the probability conditioned on $\lambda_j$ for a query text image where the extracted feature vectors are $X = \{x_1, x_2, \cdots, x_K\}$. Conventionally, the negative log-likelihood can be used to identify a writer by selecting the GMM that corresponds to the lowest value. This is given as

$$-\log p(X|\lambda_j) = \sum_{k=1}^{K} -\log p(x_k|\lambda_j)\,. \tag{7}$$

However, because the query feature vectors within the same document might be considerably similar or different from the features used to construct the GMMs, the summation given by (7) may not be effective practically since it assigns the same weight to the contribution of descriptors. To address this problem, we propose in this paper a weighted histogram-based method that efficiently derives the intermediate prediction score of the GMM for any given query handwritten text image as will be detailed later.

Let us first discuss the three types of GMM models, proposed for each writer, as illustrated by Fig.1. All three models are combined to determine the identity of query documents. The RootSIFT features extracted from the training documents of writer $j$ are used to generate a GMM identifier for that writer. Similarly, another GMM identifier for the same writer is generated from the corresponding SIFT features. These two GMM identifiers can be considered to be the authority on identifying the handwriting of writer $j$ and they are viewed as similarity GMMs (SGMM) because they describe the intra-class features. On the other hand, a Dissimilarity GMM (DGMM) is also trained for the same writer (writer $j$). The DGMM takes into consideration the contrast between the text written by writer $j$ and the rest of the writers from the database. The DGMM is generated from the SIFT features of the full database excluding the writer of interest, i.e., writer $j$. This ensures that the DGMM will cover a good range of the negative population if writer $j$ is taken as a reference.

Note that the DGMM utilizes SIFT features and not the RootSIFT ones. This is because it was observed that the DGMM based on RootSIFT features did not bring any improvements in terms of the identification accuracy, whereas the DGMM generated from the SIFT features greatly helped in improving performance. This can be justified by the fact that due to the process by which SIFT features are converted to RootSIFT (Section III-B) the divergence between the writers is minimized. This decrease in divergence helps in creating an identifier model for a single writer but does not help when one considers the contrast between a writer and the rest of writers in the database. To demonstrate this point, Fig.4 and Fig.5 show the intra-class and inter-class divergence for RootSIFT and SIFT respectively using samples from the AHTID/MW database.

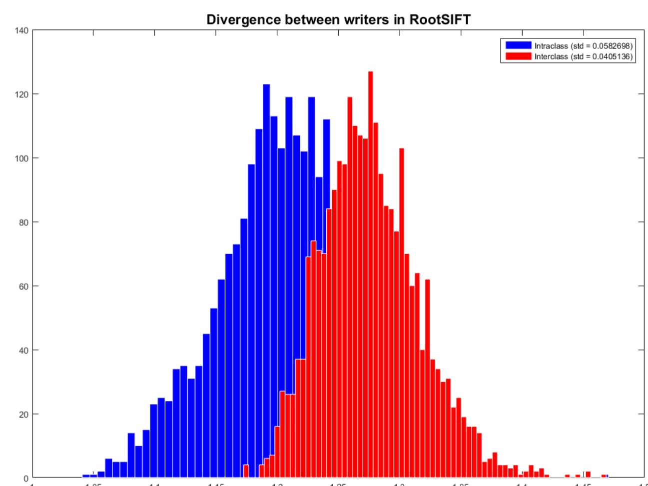



Fig. 4: Divergence between the writers of the AHTID/MW database in RootSIFT

As can be seen for RootSIFT, the divergence between the interclass documents is very small (signified by a standard deviation of 0.04). On the contrary, the divergence between the inter-class documents for SIFT is noticeably high (shown by a standard deviation of 9.69). Therefore, the DGMM appears to

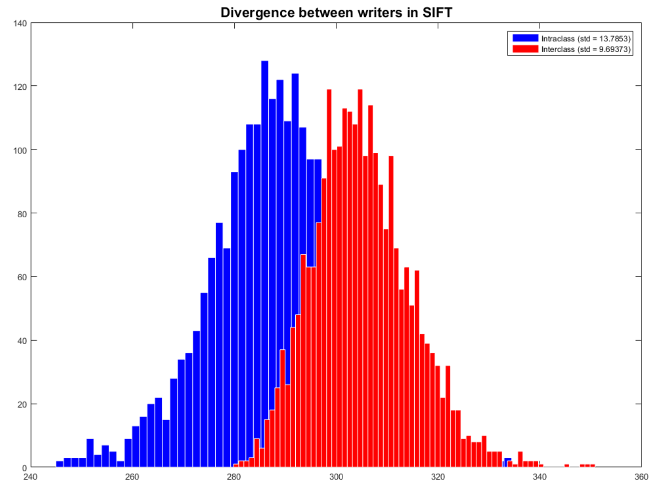



Fig. 5: Divergence between the writers of the AHTID/MW database in SIFT

contribute in identifying the query document using SIFT but fails to do so with RootSIFT.

Nevertheless, the collection of features from a full big database (excluding a single writer) can be excessive. In fact, clustering such large numbers of features may lead to problems of high dimensionality and over-fitting. In order to overcome this and to ensure consistency between the DGMM and the SGMMs of the writer, a random number of feature vectors extracted from other writer's documents equal to the number of features of that writer (i.e., writer $j$), are selected. This random selection still covers the rest of the database but without causing over-fitting.

As illustrated by Fig.6, when a query image is presented to the system, it is first segmented into words as explained in Section III-A. SIFT and RootSIFT descriptors are extracted from the segmented words of the query document. The SIFT and RootSIFT descriptors are then presented to the constructed SGMMs and DGMM of every writer. Next, instead of the conventional likelihood, a new histogram-based method is used to combine individual scores of descriptors of the same type (SIFT or RootSIFT) when a constructed GMM is applied (i.e., SGMM or DGMM). This gives an intermediate prediction score for each type of GMM. Finally, a score fusion function is used to determine the final prediction score for that writer. The lowest score among all writers will correspond to the predicted writer.

### *D. Intermediate Prediction Score with the Weighted Histogram*

The use of the negative log-likelihood as an intermediate prediction score for a query handwritten text document, as given by (7), can be viewed as a summation of individual negative log-probabilities, i.e., $-\log p(x_k|\lambda_j)$ where each represents the contribution of a descriptor to the negative log-likelihood. Note that, in this case, the contribution of each descriptor is treated equally. However, due to the nature of handwritten text data, different writing styles could share a few common patterns. Therefore, the dissimilar patterns should be penalized in their contribution to the intermediate score to enhance discriminability. Here, we propose a weighted histogram-based method to derive the intermediate score from individual contributions of the descriptors. In particular, the weights applied to the histogram bins aim to penalize the large negative log-probabilities that represent the scores of incorrect patterns whereas the scores of correct patterns are enforced with small weights.

Initially, the individual negative log-probabilities are analysed on the training data to determine a fixed minimum value $v_{min}$ and maximum value $v_{max}$ based on which histograms can be constructed. All histograms will have a constant number of bins $L$ where the first bin covers the minimum value and the last bin covers the maximum one. This is to ensure consistency for all test images throughout the dataset. Indeed, by using these fixed minimum and maximum values and a constant number of bins, all histograms will be formed on a fixed scale. Denote by $h_X^j$ the histogram corresponding to a query handwritten text image whose descriptors are $X = \{x_1, x_2, \cdots, x_K\}$ and presented to a $\mathrm{GMM}_j$ for writer $j$ (this could be a SGMM or a DGMM). It follows

$$h_X^{GMM_j}(p) = \frac{1}{K}\sum_{k=1}^{K} \delta_k^{GMM_j}(p) \ ; \ p = \{1, 2, \cdots, L\}\,, \quad (8)$$

where

$$\delta_k^j(p) = \begin{cases} 1 & v_{min} + (p-1)\,\omega \leq -\log p(x_k|\lambda_{GMM_j}) < v_{min} + p\,\omega \\ 0 & otherwise \end{cases}$$

and $\omega = \frac{v_{max} - v_{min}}{L}$ is the bin width in the histogram. $\lambda_{GMM_j}$ represents the set of parameters for $\mathrm{GMM}_j$. Theoretically, the negative log-probabilities that correspond to patterns that are similar to the ones used to generate the GMM should be very small and, therefore, the distribution of their values should fall in the first few bins of the histogram. On the other hand, the descriptors extracted from dissimilar patterns correspond to negative log-probabilities that fall in the last bins of the histogram. In the proposed method, all the bins are multiplied by an incremental value. This way, the initial bins are multiplied by a small number whereas the last bins are penalized through multiplication by a significant number in order to get clearly distinguishable. The intermediate prediction score $\Psi$ for a query image whose descriptors are $X = \{x_1, x_2, \cdots, x_K\}$ given $\mathrm{GMM}_j$ for writer $j$ is provided as

$$\Psi(X|GMM_j) = \sum_{p=1}^{L} p \times h_X^{GMM_j}(p)\,. \quad (9)$$

Obviously, if the SGMM type is considered and the query text document, $X$, is indeed written by writer $j$, the significant histogram values will be concentrated in the first few bins. As a result, the intermediate prediction score $\Psi$ will take a reasonably small value. On the other hand, if the query text document is not written by writer $j$, most of the significant histogram values will be distributed over the last bins. This will result in a large intermediate prediction score $\Psi$ due to the imposed high weighting.

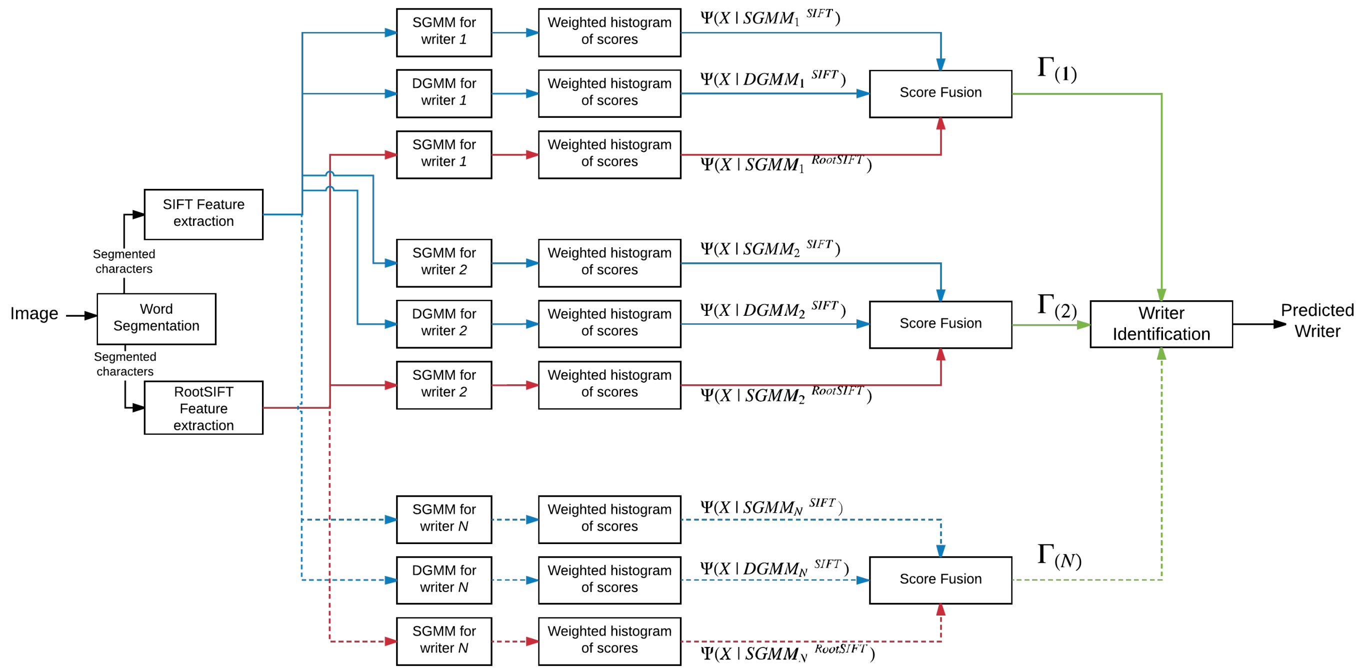



Fig. 6: Identification of an unknown query document using the proposed system

## E. Score Fusion

Given a query handwritten text image, the intermediate prediction scores are calculated as explained earlier for each writer $j \in \{1, 2, \cdots, N\}$ by considering its corresponding GMMs, i.e., SGMM$^{RootSIFT}$, SGMM$^{SIFT}$, and DGMM$^{SIFT}$. Thus, a query image will have three scores against every writer. These intermediate scores are then fused to obtain the final prediction score $\Gamma$ for each writer $j$ as

$$\begin{aligned}\Gamma(j) \quad &= \quad \Psi(X|SGMM_j^{RootSIFT}) + \alpha\ \Psi(X|SGMM_j^{SIFT}) \\ &\quad -\beta\ \Psi(X|DGMM_j^{SIFT}), \qquad (10)\end{aligned}$$

where $\alpha$ and $\beta$ are positive real numbers that act as scaling factors. This scaling is required because of the fusion of two different types of feature and GMM scores (i.e., RootSIFT and SIFT; DGMM and SGMM). The scaling parameters are determined from a validation subset of the training data. Using this arrangement of the training set, SGMM$^{RootSIFT}$, SGMM$^{SIFT}$ and DGMM$^{SIFT}$ for every writer are determined from the estimation subset. These models are then used to identify the 'known' samples from the validation subset. Using the known labels of the validation data, the selected values for $\alpha$ and $\beta$ should correspond to the highest identification rate.

Once the final prediction score is calculated for each writer, the candidate writer $j^*$ is predicted by the system as follows

$$j^* = \arg\min_j \Gamma(j) \qquad (11)$$

# IV. Experimental Results and Analysis

The proposed system was applied to six publicly available datasets to evaluate it's performance. Of the six datasets, three are English: IAM [38], Firemaker [39] and CVL [40], two are Arabic: AHTID/MW [41] and IFN/ENIT [42] and one is of hybrid-language i.e ICDAR2011 dataset [43]. The details of the datasets used are described below.

### IAM Dataset

The IAM dataset is the most widely used English database in the field of writer identification. Handwritten samples from 657 writers have been collected in this database, all of which are scanned at 300 Dots Per Inch (DPI) and saved in greyscale. Out of the 657 writers 301 have contributed two or more than two handwritten documents while the rest of the 356 writers have contributed only a single handwritten document. We have arranged this dataset as was done in [21], [24], [44], where every writer is limited to a maximum of two samples; one for training and the other for testing. For writers that have contributed multiple documents, only two are retained whereas the writers that contributed only a single document, have that document spilt roughly in half. One half is used for training while the other half is used for testing. Using this arrangement, 650 writers with useable data are left.

### IFN/ENIT Dataset

The IFN/ENIT can be considered to be the most widely used Arabic database for the purpose of hand writer identification. It consists of 411 writers who have contributed a total of 26,000 handwritten samples of different Tunisian village names. All samples are scanned at 300 DPI and are saved in binary format. We have arranged the database as explained in [27] and [44] where a significant smaller number of samples per writer are used in order to simulate conditions of the real-world scenarios. Under this setting, for every writer, 30 and

20 randomly selected words are used for training and testing purposes, respectively.

*AHTID/MW Dataset*

AHTID/MW dataset consists of handwritten samples from 53 native Arabic writers. These writers have contributed a total of 3,710 text lines without any restriction on the type of pen being used. All samples are scanned at 300 DPI and saved in greyscale format. All of the text samples have been divided into 4 sets. As done in [44], 3 sets have been used for training and the last set has been used for testing. Identification results are obtained after four fold cross validation.

*CVL Dataset*

The CVL dataset contains handwritten documents from 311 writers. Of these 311 writers, 27 writers produced 7 documents of text while the remaining 284 writers produced 5 documents of text. Each writer produced one sample in cursive German and the rest in cursive English. All documents are scanned at 300 DPI and saved in a colour depth of 24 bit. In our experiments, we have only utilized the English documents. As was done in [44] three documents per writer are used for training, while the fourth is used for testing.

*Firemaker Dataset*

The Firemaker dataset contains handwritten samples from 250 writers where each writer has contributed 4 pages of handwritten text. In page 1 the subjects were asked to copy five short paragraphs using their own normal handwriting. In page 2 the subjects were asked to do the same for two paragraphs but only using uppercase handwriting. In page 3 the subjects were encouraged to write in a "forged" text where they were asked to write in a style that is not their own. On page 4 the subjects were asked to explain a given cartoon in their own words. As was done in [21] we have only used page 1 and page 4 for our experiments.

*ICDAR2011 Dataset*

The ICDAR2011 dataset contains pages of handwritten text in four different languages i.e. English, French, Greek and German. 26 writers have contributed to this dataset, writing 2 full pages of text for each language, thereby giving 8 pages of text per writer. A variation of the ICDAR2011 dataset is known as the ICDAR2011 cropped dataset which is made by cropping only the first two lines from every image. This significantly reduces the available text for each writer. We will be using the cropped variation of the dataset in our experiments. For every writer 5 images are used for training while 3 images are used for testing.

*Experimental Setup*

For determining the optimal parameters of the system, the training data of each dataset was divided into two subsets: an estimation subset (˜ 70% of the training data) and a validation subset (˜ 30% of the training data). Using such an arrangement of the training data, the writer models were generated using the estimation subsets which were then used to identify the 'known' samples from the validation subsets. The results reported in Tables I - VIII have been achieved using the validation subset of the training data.

TABLE I: Comparison of Top 1 accuracy achieved (in percentage) using all models at varying Gaussians.

| | Writer model used | Gaussians used | | | | | |
|---|---|---|---|---|---|---|---|
| | | **16** | **32** | **64** | **128** | **256** | **512** |
| **IAM** | $\mathbf{SGMM^{RootSIFT}}$ | 54.23 | 61.54 | 66.62 | 71.11 | 70.15 | - |
| | $\mathbf{SGMM^{SIFT}}$ | 21.08 | 15.77 | 13.23 | 11.00 | 10.62 | - |
| | $\mathbf{(SGMM^{SIFT}-DGMM^{SIFT})}$ | 32.62 | 39.62 | 33.85 | 17.46 | 8.11 | - |
| **IFN/ENIT** | $\mathbf{SGMM^{RootSIFT}}$ | 19.75 | 41.73 | 60.00 | 62.72 | 66.91 | 70.41 |
| | $\mathbf{SGMM^{SIFT}}$ | 6.67 | 6.77 | 5.68 | 4.44 | 3.70 | 2.18 |
| | $\mathbf{(SGMM^{SIFT}-DGMM^{SIFT})}$ | 2.96 | 9.63 | 10.86 | 8.89 | 4.71 | 1.74 |
| **AHTID/MW** | $\mathbf{SGMM^{RootSIFT}}$ | 35.22 | 49.37 | 64.47 | 75.47 | 80.82 | 84.59 |
| | $\mathbf{SGMM^{SIFT}}$ | 42.45 | 44.03 | 53.77 | 54.41 | 43.71 | 28.30 |
| | $\mathbf{(SGMM^{SIFT}-DGMM^{SIFT})}$ | 27.04 | 34.28 | 60.38 | 72.33 | 76.10 | 64.78 |
| **CVL** | $\mathbf{SGMM^{RootSIFT}}$ | 76.05 | 87.70 | 93.85 | 96.76 | 97.73 | 98.06 |
| | $\mathbf{SGMM^{SIFT}}$ | 83.17 | 79.94 | 69.26 | 56.31 | 39.48 | 27.26 |
| | $\mathbf{(SGMM^{SIFT}-DGMM^{SIFT})}$ | 66.02 | 89.63 | 94.15 | 94.42 | 80.22 | 71.17 |
| **Firemaker** | $\mathbf{SGMM^{RootSIFT}}$ | 58.93 | 61.88 | 68.37 | 71.41 | 73.50 | 73.86 |
| | $\mathbf{SGMM^{SIFT}}$ | 46.32 | 42.00 | 31.54 | 22.42 | 10.23 | 6.99 |
| | $\mathbf{(SGMM^{SIFT}-DGMM^{SIFT})}$ | 51.74 | 54.39 | 60.45 | 61.99 | 23.26 | 18.37 |
| **ICDAR2011** | $\mathbf{SGMM^{RootSIFT}}$ | 83.15 | 86.08 | 90.62 | 90.62 | 91.00 | - |
| | $\mathbf{SGMM^{SIFT}}$ | 71.23 | 73.62 | 68.23 | 64.48 | 61.31 | - |
| | $\mathbf{(SGMM^{SIFT}-DGMM^{SIFT})}$ | 75.08 | 75.69 | 78.15 | 72.23 | 67.85 | - |

TABLE II: Diversity measure of the RootSIFT ($SGMM^{RootSIFT}$) and SIFT ($SGMM^{SIFT}-DGMM^{SIFT}$) systems using the correlation coefficient.

| | Dataset Used | | | | | |
|---|---|---|---|---|---|---|
| **Diversity Measure** | **IAM** | **IFN/ENIT** | **AHTID/MW** | **CVL** | **Firemaker** | **ICDAR2011** |
| **Correlation coefficient** | 0.258 | 0.114 | 0.233 | 0.317 | 0.334 | 0.109 |

## A. Complementarity of SIFT and RootSIFT

For individual systems to perform well in a combined setting, a diversity measure is used to determine the complementarity of the two systems. Without determining this measure, it is possible that the combination of systems would perform no better than the individual systems alone [45]. The correlation coefficient [46] has been used as the pairwise diversity measure here and has been reported in Table II for all datasets (using the optimal number of Gaussians for each dataset).

Since RootSIFT is derived from SIFT, it was expected that the two systems exhibit some correlation as demonstrated in Table II. Ideally, two completely independent systems should correspond to a correlation coefficient of 0, however, it is worth mentioning that a low correlation coefficient indicates a good complementarity of the two systems.

### *B. Sensitivity to Model Parameters*

The Top 1 accuracy achieved using $\text{SGMM}^{\text{RootSIFT}}$, $\text{SGMM}^{\text{SIFT}}$ and $\text{DGMM}^{\text{SIFT}}$ for varying number of Gaussians for all the datasets was recorded. These results are depicted in Table I. As can be seen, $\text{SGMM}^{\text{RootSIFT}}$ improves in terms of performance as the number of Gaussians increase, whereas the performance of $\text{SGMM}^{\text{SIFT}}$ deteriorates. Therefore, it is sensible to use a different number of Gaussians for each descriptor type at the score fusion level (Section III-E). However, it can also be observed that $\text{SGMM}^{\text{SIFT}}$has performed better than $\text{SGMM}^{\text{RootSIFT}}$at 16 Gaussians for the AHTID/MW and CVL datasets. This can be justified by the fact that when generating the writer models at such a low number of Gaussians, there are no sufficient parameters to model such a complex distribution of multivariate statistical samples. However, as the number of parameters (Gaussians) increases, the system starts to learn the data and reaches its optimal performance at a certain number of Gaussians.

Theoretically speaking, data can be efficiently modelled by a GMM if the number of features are significantly higher than the number of GMM parameters. However, from an implementation point of view, this was not possible for some of the datasets used in this work. Indeed, the IAM dataset, for example, has a huge variation in the number of samples per writer, and for many writers the GMM could not be built with 512 Gaussians. As a result, performance with 512 Gaussians is not shown in Table I for the IAM and ICDAR2011 datasets, respectively.

For the IAM dataset, Table III represents the Top 1 accuracy achieved (in percentage) as a function of varying number of Gaussians used at the score fusion stage using either SIFT or RootSIFT features. The results displayed in Table III validates our observation in the sense that SIFT features are modelled more effectively with a lower number of Gaussians whereas RootSIFT features are described more effectively with a higher number of Gaussians. Thus, a judicious selection of the number of Gaussians for RootSIFT and SIFT prior to the score fusion stage can lead to high performance. As can be seen from Table III the IAM dataset performs best using 256 Gaussians for RootSIFT combined with 32 Gaussians for SIFT.

It is worth noting that the best performing number of Gaussians for SIFT and RootSIFT is in perfect agreement with that of the validation subsets used to estimate the scaling parameters (See Section III-E). Therefore, from a practical perspective, the optimal number of Gaussians can be determined at the training stage on a validation subset. In the rest of the paper, the number of Gaussians used for SIFT and RootSIFT corresponds to the best performing combination in Tables III - VIII, respectively.

TABLE III: Top 1 accuracy achieved (in percentage) on the IAM dataset with varying number of Gaussians for the SIFT and RootSIFT features.

| SIFT \ RootSIFT | **16** | **32** | **64** | **128** | **256** | **512** |
|---|---|---|---|---|---|---|
| **16** | 64.85 | 67.15 | 69.54 | 74.00 | 78.23 | - |
| **32** | 59.54 | 74.77 | 74.92 | 77.31 | **81.77** | - |
| **64** | 71.92 | 73.31 | 75.77 | 76.00 | 70.15 | - |
| **128** | 64.69 | 67.54 | 67.92 | 61.69 | 59.85 | - |
| **256** | 6.23 | 9.15 | 14.31 | 23.69 | 31.92 | - |
| **512** | - | - | - | - | - | - |

TABLE IV: Top 1 accuracy achieved (in percentage) on the IFN/ENIT dataset with varying number of Gaussians for the SIFT and RootSIFT features.

| SIFT \ RootSIFT | **16** | **32** | **64** | **128** | **256** | **512** |
|---|---|---|---|---|---|---|
| **16** | 26.91 | 43.46 | 60.49 | 66.91 | 70.12 | 71.49 |
| **32** | 34.32 | 47.41 | 61.23 | 67.41 | 76.05 | 72.10 |
| **64** | 54.32 | 62.72 | 72.10 | 73.33 | 76.05 | 70.33 |
| **128** | 49.88 | 56.54 | 66.67 | 64.44 | **80.93** | 74.41 |
| **256** | 32.98 | 42.41 | 48.72 | 51.67 | 58.32 | 63.14 |
| **512** | 2.47 | 4.20 | 9.14 | 21.80 | 29.13 | 34.54 |

TABLE V: Top 1 accuracy achieved (in percentage) on the AHTID/MW dataset with varying number of Gaussians for the SIFT and RootSIFT features.

| SIFT \ RootSIFT | **16** | **32** | **64** | **128** | **256** | **512** |
|---|---|---|---|---|---|---|
| **16** | 44.03 | 55.66 | 65.41 | 73.27 | 77.04 | 83.96 |
| **32** | 51.89 | 65.41 | 75.79 | 77.99 | 82.70 | 87.42 |
| **64** | 63.21 | 68.55 | 75.47 | 79.87 | 83.96 | 90.57 |
| **128** | 71.70 | 77.36 | 80.50 | 83.33 | 87.11 | 89.31 |
| **256** | 74.84 | 77.36 | 83.33 | 86.16 | 89.31 | **90.88** |
| **512** | 79.56 | 82.39 | 83.33 | 85.85 | 87.42 | 89.94 |

TABLE VI: Top 1 accuracy achieved (in percentage) on the CVL dataset with varying number of Gaussians for the SIFT and RootSIFT features.

| SIFT \ RootSIFT | **16** | **32** | **64** | **128** | **256** | **512** |
|---|---|---|---|---|---|---|
| **16** | 89.00 | 93.53 | 95.79 | 97.73 | 98.38 | 98.71 |
| **32** | 93.85 | 95.79 | 97.73 | 98.06 | 98.38 | 98.71 |
| **64** | 95.47 | 97.41 | 98.38 | 98.06 | 98.71 | 99.03 |
| **128** | 97.09 | 98.06 | 98.06 | 98.71 | 99.03 | **99.35** |
| **256** | 95.47 | 97.09 | 98.38 | 98.38 | 98.71 | 99.03 |
| **512** | 92.31 | 96.73 | 97.09 | 97.41 | 97.73 | 98.71 |

TABLE VII: Top 1 accuracy achieved (in percentage) on the Firemaker dataset with varying number of Gaussians for the SIFT and RootSIFT features.

| SIFT \ RootSIFT | **16** | **32** | **64** | **128** | **256** | **512** |
|---|---|---|---|---|---|---|
| **16** | 64.09 | 64.73 | 66.38 | 69.03 | 73.71 | 74.41 |
| **32** | 67.17 | 68.51 | 72.88 | 74.93 | 76.47 | 77.69 |
| **64** | 70.55 | 70.74 | 73.17 | 75.66 | 77.91 | **81.52** |
| **128** | 71.62 | 73.06 | 76.09 | 78.73 | 77.71 | 75.71 |
| **256** | 65.57 | 66.17 | 69.98 | 69.93 | 71.45 | 71.93 |
| **512** | 1.12 | 1.98 | 7.33 | 10.52 | 13.90 | 19.74 |

TABLE VIII: Top 1 accuracy achieved (in percentage) on the ICDAR2011 dataset with varying number of Gaussians for the SIFT and RootSIFT features.

| SIFT \ RootSIFT | **16** | **32** | **64** | **128** | **256** | **512** |
|---|---|---|---|---|---|---|
| **16** | 86.08 | 87.69 | 88.62 | 88.62 | 89.69 | - |
| **32** | 81.00 | 83.15 | 82.08 | 83.15 | 85.23 | - |
| **64** | 82.15 | 84.23 | 89.62 | 91.08 | **91.85** | - |
| **128** | 81.08 | 83.69 | 84.15 | 84.00 | 84.00 | - |
| **256** | 81.08 | 82.00 | 85.15 | 85.15 | 88.08 | - |
| **512** | - | - | - | - | - | - |

### C. *Evaluation of the Score Fusion Method*

In this set of experiments, the capability of the proposed fusion method, given by (10), of efficiently exploiting SIFT, RootSIFT, DGMM, and SGMM is demonstrated. To this end, we have considered the performance of separate descriptors $SGMM^{RootSIFT}$, $SGMM^{SIFT}$ as well as the combination of GMMs ($SGMM^{SIFT}$ - $DGMM^{SIFT}$). Results are illustrated by Table IX. As can be seen, the accuracy achieved with $SGMM^{SIFT}$ for each dataset has been significantly improved when $DGMM^{SIFT}$ is taken into account. Furthermore, the combination of the intermediate scores using a simple yet efficient linear function, as described by (10), offers a significantly higher performance. This shows that the SIFT and RootSIFT descriptors can be complementary tools for handwritten text identification.

TABLE IX: Comparison of Top 1 accuracy achieved (in percentage) using SGMM/DGMM models and the proposed score fusion approach.

| **Dataset Used** | **Approach used** | | | |
|---|---|---|---|---|
| | **$SGMM^{RootSIFT}$** | **$SGMM^{SIFT}$** | **($SGMM^{SIFT}$ – $DGMM^{SIFT}$)** | **Score Fusion (10)** |
| **IAM** | 91.23 | 30.46 | 50.46 | 97.85 |
| **IFN/ENIT** | 87.65 | 10.12 | 41.01 | 97.28 |
| **AHTID/MW** | 92.14 | 63.21 | 82.70 | 95.60 |
| **CVL** | 98.71 | 88.35 | 97.41 | 99.03 |
| **Firemaker** | 95.55 | 65.99 | 86.23 | 97.98 |
| **ICDAR2011** | 98.08 | 84.62 | 96.15 | 100.0 |

### D. *Evaluation of the Weighted Histogram*

As discussed in subsection III-D, a query handwritten text document is represented by a number of key point descriptors where each descriptor produces its own SGMM/DGMM score. The use of the negative log-likelihood as an intermediate prediction score in the conventional approach, as given by (7), can be thought of as a summation of individual negative log-probabilities, where each represents the individual contribution of a descriptor to the negative log-likelihood. That is, the contribution of each descriptor is treated equally in the overall summation. Our proposed weighted histogram technique, however, is based on the fact that handwritings from the same writer should exhibit more similar textual patterns than dissimilar ones and, thus, by representing these scores in a histogram and furthermore, by penalizing the bad scores via a cost function (see (9)), a prominent contrast between the dissimilar handwritings can be achieved. A comparison between our proposed weighted histogram technique and the conventional negative log-likelihood one is made in Table X. As can be seen, the proposed technique brings significant improvements on all datasets.

TABLE X: Top 1 accuracy achieved (in percentage) on all datasets using proposed weighted histogram-based approach versus the averaging of scores approach.

| **Dataset Used** | **Approach used** | |
|---|---|---|
| | **Proposed weighted histogram technique (see (9))** | **Negative Log-likelihood (see (7))** |
| **IAM** | 97.85 | 86.00 |
| **IFN/ENIT** | 97.28 | 87.41 |
| **AHTID/MW** | 95.60 | 88.05 |
| **CVL** | 99.03 | 98.38 |
| **Firemaker** | 97.98 | 91.90 |
| **ICDAR2011** | 100.0 | 98.08 |

### E. *Addition of New Writers to the System*

Since negative samples are required for the generation of each DGMM, it is worth assessing the sensitivity of the system to the addition of new writers in the dataset without retraining the existing DGMMs. In this experiment, all the DGMMs were obtained from a subset of the full dataset. All dissimilarity writer models that correspond to new writers outside this subset were also created from the sample documents of the subset. For example, with the IAM dataset the DGMM for every writer was created using only the first 400 writers (˜ 60% of the dataset), therefore the remaining 250 writers (˜ 40% of the dataset) also trained their respective DGMM's using the features from the first 400 writers, this emulated addition of new writers to the system. Similar settings were used for all datasets. Table XI reports the comparison of Top 1 Accuracy achieved when only a portion of writers are used for DGMM training vs when all writers are used. It can be observed that the results are very slightly affected.

TABLE XI: Top 1 Accuracy achieved for DGMM generated from a fixed portion of writers vs DGMM generated from all the writers.

| DGMM training | Dataset Used | | | | | |
|---|---|---|---|---|---|---|
| | IAM | IFN/ENIT | AHTID | CVL | Firemaker | ICDAR |
| 60% of writers | 97.82% | 97.13% | 95.41% | 98.71% | 97.57% | 100% |
| 100% of writers | 97.85% | 97.28% | 95.60% | 99.03% | 97.98% | 100% |

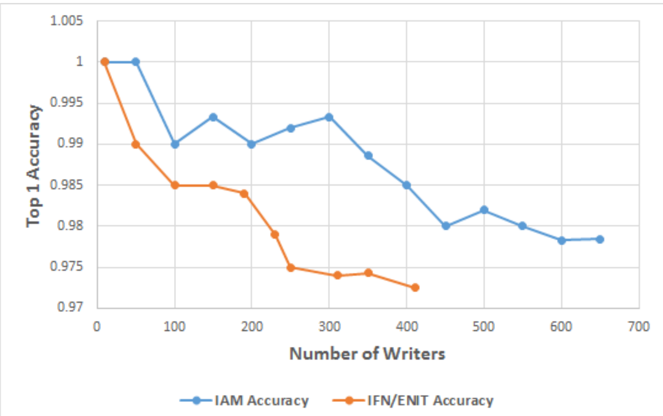



Fig. 7: Top 1 Accuracy achieved on the IAM and IFN/ENIT datasets as a function of number of writers.

### F. Sensitivity to Number of Writers

To demonstrate the performance of the system with respect to number of writers, a series of experiments were performed by varying the number of writers from 10 to all the the writers in the dataset. Fig. 7 shows the Top 1 Accuracy achieved on the two large datasets: IAM and IFN/ENIT. As expected, accuracy decreases gradually as the number of writers are increased.

### G. Comparison with Existing Works

An experimental study of the proposed system was carried out using all of the datasets described previously and the results obtained were compared with the state of the art techniques already published in the field of writer identification for their respective datasets. Comparison of our proposed system with the state of the art systems using the IAM, IFN/ENIT, AHTID/MW, CVL, Firemaker and ICDAR2011 datasets are shown in Tables XII, XIII, XIV, XV, XVI and XVII respectively.

Using our proposed dissimilarity based approached we have achieved a Top 1 accuracy of 97.85% on the IAM dataset, which although comparable to state of the art systems was only slightly outperformed by the system presented by Wu et al., [24]. Using the IFN/ENIT Arabic dataset a Top 1 accuracy of 97.28% was achieved by our proposed system, this outperforms the nearest best performing system of Hannad et al., [27] by about 2.4%. For the AHTID/MW Arabic dataset we achieved a Top 1 accuracy of 95.60%, this result outperforms the nearest best performing system of Khan et al., [47] by a margin of 8.10%. For the CVL dataset, a 99.03% of Top 1 accuracy was achieved, which was marginally outperformed by the system of Khan et al., [44] by a margin of 0.57%. For the Firemaker dataset our proposed system achieved a state of the art Top 1 accuracy of 97.89%, outperforming the nearest best performing system of Wu et al., [24] by 5.49%. Finally, for the cropped version of the ICDAR2011 multiple language dataset we were able to achieved a Top 1 accuracy of 100%. This dataset, although having comparatively smaller number of writers is challenging because of the multiple languages used and because of the cropped versions significantly reducing the data available per writer. However, if the number of writers were increased and made comparable to the other larger datasets we believe we may not achieve a 100% accuracy rate, but are confident that our proposed system would still fare better than the previously published systems.

TABLE XII: Accuracy comparison of the proposed system with the state of the art systems in writer identification for the IAM database.

| System | Feature source | Number of writers | Top 1 Accuracy |
|---|---|---|---|
| Bulacu and Schomaker, [21] | Contours | 650 | 89.00% |
| Siddiqi and Vincent, [20] | Contours | 650 | 89.00% |
| Siddiqi and Vincent, [48] | Contours | 650 | 91.00% |
| Kumar et al., [49] | Fraglets | 650 | 88.40% |
| Ghiasi and Safabakhsh, [50] | Graphemes | 650 | 93.70% |
| Bertolini et al. , [25] | Full page | 650 | 96.70% |
| Khalifa et al., [51] | Graphemes | 650 | 92.00% |
| Jain and Doermann, [52] | Connected components | 657 | 94.70% |
| Hannad et al., [27] | Fragments | 657 | 89.50% |
| Brink et al., [22] | Ink trace width | 657 | 97.00% |
| Schomaker and Bulacu, [53] | Contours | 657 | 82.50 % |
| Khan et al., [44] | Overlapping blocks | 650 | 97.20% |
| He et al., [54] | Junclets | 650 | 91.10% |
| **Wu et al., [24]** | **Words** | **650** | **98.50%** |
| Proposed system | Words | 650 | 97.85% |

TABLE XIII: Accuracy comparison of the proposed system with the state of the art systems in writer identification for the IFN/ENIT database.

| System | Feature source | Number of writers | Top 1 Accuracy |
|---|---|---|---|
| Bulacu and Schomaker, [55] | Graphemes | 350 | 88.00% |
| Chawki et al., [56] | Full page | 130 | 82.00% |
| Djeddi et al., [57] | Full page | 130 | 84.23% |
| Abdi and Khemakhem, [58] | Contours | 100 | 85.00% |
| Abdi and Khemakhem, [59] | Graphemes | 411 | 90.02% |
| Hannad et al., [27] | Fragments | 411 | 94.89% |
| Khan et al., [44] | Overlapping blocks | 411 | 76.00% |
| **Proposed system** | **Words** | **411** | **97.28%** |

TABLE XIV: Accuracy comparison of the proposed system with the state of the art systems in writer identification for the AHTID/MW database.

| System | Feature source | Number of writers | Top 1 Accuracy |
|---|---|---|---|
| Slimane and Margner, [60] | Sliding window | 53 | 69.40% |
| Schomaker and Bulacu, [53] | Contours | 53 | 66.40% |
| Our implementation of Hannad et al., [27] | Fragments | 53 | 77.30% |
| Khan et al., [47] | Full page | 53 | 87.50% |
| Khan et al., [44] | Overlapping blocks | 53 | 71.60% |
| **Proposed system** | **Words** | **53** | **95.60%** |

TABLE XV: Accuracy comparison of the proposed system with the state of the art systems in writer identification for the CVL database.

| System | Feature source | Number of writers | Top 1 Accuracy |
|---|---|---|---|
| Fiel and Sablatnig, [32] | Full page | 309 | 97.80% |
| Jain and Doermann, [52] | Connected components | 310 | 99.40% |
| Christlein et al., [61] | Full page | 310 | 99.20% |
| Fiel and Sablatnig, [62] | Words | 309 | 98.90% |
| Schomaker and Bulacu, [53] | Contours | 310 | 81.80% |
| Our implementation of Hannad et al., [27] | Fragments | 310 | 96.20% |
| **Khan et al., [44]** | **Overlapping blocks** | **310** | **99.60%** |
| Proposed system | Words | 310 | 99.03% |

TABLE XVI: Accuracy comparison of the proposed system with the state of the art systems in writer identification for the Firemaker database.

| System | Feature source | Number of writers | Top 1 Accuracy |
|---|---|---|---|
| Bulacu and Schomaker, [21] | Graphemes | 250 | 83.00% |
| Li and Ding, [63] | Edge pixels | 250 | 78.00% |
| Brink et al., [22] | Ink trace width | 250 | 86.00% |
| Ghiasi et al., [50] | Graphemes | 250 | 91.80% |
| He et al., [54] | Junclets | 250 | 89.80% |
| Wu et al., [24] | Words | 250 | 92.40% |
| Khan et al., [44] | Overlapping blocks | 250 | 89.47% |
| **Proposed system** | **Words** | **250** | **97.98%** |

TABLE XVII: Accuracy comparison of the proposed system with the state of the art systems in writer identification for the ICDAR2011 cropped database.

| System | Feature source | Number of writers | Top 1 Accuracy |
|---|---|---|---|
| ECNU method, [43] | Contours | 26 | 65.90% |
| QUQA-a method, [43] | Graphemes | 26 | 74.00% |
| QUQA-b method, [43] | Graphemes | 26 | 67.30% |
| TSINGHUA method, [43] | Sliding window | 26 | 90.90% |
| GWU method, [43] | Unspecified | 26 | 74.00% |
| CS-UMD method, [43] | Adjacent segments | 26 | 66.80% |
| TEBESSA method, [43] | Full page | 26 | 87.50% |
| MCS-NUST method, [43] | Contours | 26 | 82.20% |
| Wu et al., [24] | Words | 26 | 95.20% |
| Khan et al., [44] | Overlapping blocks | 26 | 82.69% |
| **Proposed system** | **Words** | **26** | **100.0%** |

### *H. Discussion*

A significant effort was made to make the comparisons made in Section IV-G transparent by clearly stating the structure and arrangement of the datasets used and by arranging the datasets in the same manner as was previously done in literature. Although some of the systems against which comparisons were made have clearly stated their dataset arrangements, many authors do not share this information. This information is valuable as changes to the dataset structure and arrangement have an impact on the performance of the system. Keeping this in view we provide the following arguments regarding our system, along with the one to one comparison.

In almost all of our comparisons, the datasets with a large number of writers have been considered. This is necessary as in real world writer identification scenarios the number of writers is a determined factor for the evaluation of the performance of any system. This was also observed by Hannad et al., that for a system of writer identification, a natural and gradual decrease in performance accuracy occurs as the number of writers are increased [27]. Our proposed system has performed at a more than acceptable level on all datasets having large number of writers. However, the major limitation of our system is that since it relies on GMM clustering to generate a writer model, it requires a fairly large number of features to generate a strong predictor model of any writer. This explains the incorrect predictions achieved when using the IAM dataset as the writers having limited training data were unable to provide a strong and correct prediction.

Furthermore, in the case of the Arabic datasets, it was observed by Balacu et al., [55] and Khan et al. [44] that systems that tend to perform better on scripts such as Roman and Latin fail to perform acceptably when applied to an Arabic dataset. It was concluded that due to the Arabic writing style, identification of Arabic handwriting is a more challenging task than identification of writers in other scripts. Keeping in view this observation, our proposed system was able to perform well irrespective of the script used.

## V. Conclusion

In this paper, an offline handwritten text identification system has been proposed. The concept of similarity and dissimilarity GMMs has been introduced and incorporated in the proposed system. Furthermore, SIFT and RootSIFT descriptors have been extracted from handwritten text images. These descriptors are then used to generate similarity and dissimilarity GMMs for each writer. Interestingly, the use of both SIFT and RootSIFT descriptors combined together in a single system has proven to be efficient on handwritten text data which suggests that the two features are complementary rather than redundant for handwritten text identification. Given a query text image, a GMM produces an intermediate prediction score via a new weighted histogram-based method for each writer. This has been shown to perform significantly better than the conventional averaging of the negative log-likelihood scores, because the contribution of irrelevant descriptors is penalized by the weighting process. Intermediate prediction scores are then efficiently fused using a linear function to obtain the final prediction score. Assessed on a number of handwritten text datasets through intensive experiments, the proposed system has been shown to operate remarkably well with different handwritten languages. Experiments have also shown the superiority of the proposed system over state-of-the-art techniques.

## Acknowledgement

This publication was made possible by NPRP grant number NPRP NPRP7 - 442 - 1 - 082 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

## References

[1] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Transactions on circuits and systems for video technology*, vol. 14, no. 1, pp. 4–20, 2004.

[2] M. Tapiador, J. Gómez, and J. A. Sigüenza, "Writer identification forensic system based on support vector machines with connected components," in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. Springer, 2004, pp. 625–632.

[3] A. Fornés, J. Lladós, G. Sánchez, and H. Bunke, "Writer identification in old handwritten music scores," in *Document Analysis Systems, 2008. DAS'08. The Eighth IAPR International Workshop on*. IEEE, 2008, pp. 347–353.

[4] A. Fornes, J. Llados, G. Sanchez, and H. Bunke, "On the use of textural features for writer identification in old handwritten music scores," in *2009 10th International Conference on Document Analysis and Recognition*. IEEE, 2009, pp. 996–1000.

[5] L. Ballard, D. Lopresti, and F. Monrose, "Evaluating the security of handwriting biometrics," in *Tenth International Workshop on Frontiers in Handwriting Recognition*. Suvisoft, 2006, pp. –.

[6] A. Namboodiri and S. Gupta, "Text independent writer identification from online handwriting," in *Tenth International Workshop on Frontiers in Handwriting Recognition*. Suvisoft, 2006, pp. –.

[7] A. K. Jain, F. D. Griess, and S. D. Connell, "On-line signature verification," *Pattern recognition*, vol. 35, no. 12, pp. 2963–2972, 2002.

[8] W. Yang, L. Jin, and M. Liu, "Deepwriterid: An end-to-end online text-independent writer identification system," *IEEE Intelligent Systems*, vol. 31, no. 2, pp. 45–53, 2016.

[9] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2911–2918.

[10] H. E. Said, T. N. Tan, and K. D. Baker, "Personal identification based on handwriting," *Pattern Recognition*, vol. 33, no. 1, pp. 149–160, 2000.

[11] F. Nejad and M. Rahmati, "A new method for writer identification and verification based on farsi/arabic handwritten texts," in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, vol. 2. IEEE, 2007, pp. 829–833.

[12] B. Helli and M. E. Moghaddam, "A text-independent persian writer identification system using lcs based classifier," in *2008 IEEE International Symposium on Signal Processing and Information Technology*. IEEE, 2008, pp. 203–206.

[13] B. Helli and M. E. Moghaddam, "A text-independent persian writer identification based on feature relation graph (frg)," *Pattern Recognition*, vol. 43, no. 6, pp. 2199–2209, 2010.

[14] A. Schlapbach and H. Bunke, "Using hmm based recognizers for writer identification and verification," in *Frontiers in Handwriting Recognition, 2004. IWFHR-9 2004. Ninth International Workshop on*. IEEE, 2004, pp. 167–172.

[15] A. Schlapbach and Bunke, "Off-linewriter identification using gaussian mixture models," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 3. IEEE, 2006, pp. 992–995.

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

[17] M. E. Moghaddam, "Text-independent persian writer identification using fuzzy clustering approach," in *Information Management and Engineering, 2009. ICIME'09. International Conference on*. IEEE, 2009, pp. 728–731.

[18] I. Siddiqi and N. Vincent, "Combining global and local features for writer identification," in *Proceedings of the 11. Int. Conference on Frontiers in Handwriting Recognition, Montreal*, 2008, pp. –.

[19] I. Siddiqi and N. Vincent, "A set of chain code based features for writer recognition," in *2009 10th International Conference on Document Analysis and Recognition*. IEEE, 2009, pp. 981–985.

[20] I. Siddiqi and N. Vincent, "Combining contour based orientation and curvature features for writer recognition," in *International Conference on Computer Analysis of Images and Patterns*. Springer, 2009, pp. 245–252.

[21] M. Bulacu and L. Schomaker, "Text-independent writer identification and verification using textural and allographic features," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 4, pp. 701–717, 2007.

[22] A. Brink, J. Smit, M. Bulacu, and L. Schomaker, "Writer identification using directional ink-trace width measurements," *Pattern Recognition*, vol. 45, no. 1, pp. 162–171, 2012.

[23] A. Bensefia, T. Paquet, and L. Heutte, "A writer identification and verification system," *Pattern Recognition Letters*, vol. 26, no. 13, pp. 2080–2092, 2005.

[24] X. Wu, Y. Tang, and W. Bu, "Offline text-independent writer identification based on scale invariant feature transform," *Information Forensics and Security, IEEE Transactions on*, vol. 9, no. 3, pp. 526–536, 2014.

[25] D. Bertolini, L. S. Oliveira, E. Justino, and R. Sabourin, "Texture-based descriptors for writer identification and verification," *Expert Systems with Applications*, vol. 40, no. 6, pp. 2069–2080, 2013.

[26] R. K. Hanusiak, L. S. Oliveira, E. Justino, and R. Sabourin, "Writer verification using texture-based features," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 15, no. 3, pp. 213–226, 2012.

[27] Y. Hannad, I. Siddiqi, and M. E. Y. El Kettani, "Writer identification using texture descriptors of handwritten fragments," *Expert Systems with Applications*, vol. 47, pp. 14–22, 2016.

[28] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, no. 285-296, pp. 23–27, 1975.

[29] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[30] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 2161–2168.

[31] S. Fiel and R. Sablatnig, "Writer retrieval and writer identification using local features," in *Document Analysis Systems (DAS), 2012 10th IAPR International Workshop on*. IEEE, 2012, pp. 145–149.

[32] S. Fiel and R. Sablatnig, "Writer identification and writer retrieval using

the fisher vector on visual vocabularies," in *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*. IEEE, 2013, pp. 545–549.

[33] Y.-J. Xiong, Y. Wen, P. S. Wang, and Y. Lu, "Text-independent writer identification using sift descriptor and contour-directional feature," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2015, pp. 91–95.

[34] M. S. Nikulin, "Hellinger distance," *Encyclopedia of mathematics*, vol. 151, pp. –, 2001.

[35] D. A. Reynolds, "Speaker identification and verification using gaussian mixture speaker models," *Speech communication*, vol. 17, no. 1, pp. 91–108, 1995.

[36] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital signal processing*, vol. 10, no. 1, pp. 19–41, 2000.

[37] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[38] U.-V. Marti and H. Bunke, "The iam-database: an english sentence database for offline handwriting recognition," *International Journal on Document Analysis and Recognition*, vol. 5, no. 1, pp. 39–46, 2002.

[39] L. Schomaker and L. Vuurpijl, "Forensic writer identification: A benchmark data set and a comparison of two systems [internal report for the Netherlands Forensic Institute]," pp. –, 2000.

[40] F. Kleber, S. Fiel, M. Diem, and R. Sablatnig, "Cvl-database: An off-line database for writer retrieval, writer identification and word spotting," in *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*. IEEE, 2013, pp. 560–564.

[41] A. Mezghani, S. Kanoun, M. Khemakhem, and H. E. Abed, "A database for arabic handwritten text image recognition and writer identification," in *Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on*. IEEE, 2012, pp. 399–402.

[42] M. Pechwitz, S. S. Maddouri, V. Märgner, N. Ellouze, H. Amiri *et al.*, "Ifn/enit-database of handwritten arabic words," in *Proc. of CIFED*, vol. 2. Citeseer, 2002, pp. 127–136.

[43] G. Louloudis, N. Stamatopoulos, and B. Gatos, "Icdar 2011 writer identification contest," in *2011 International Conference on Document Analysis and Recognition*. IEEE, 2011, pp. 1475–1479.

[44] 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*, vol. 71, pp. 404–415, 2017.

[45] B. Hadjadji, Y. Chibani, and H. Nemmour, "An efficient open system for offline handwritten signature identification based on curvelet transform and one-class principal component analysis," *Neurocomputing*, vol. 265, pp. 66–77, 2017.

[46] L. I. Kuncheva and C. J. Whitaker, "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy," *Machine learning*, vol. 51, no. 2, pp. 181–207, 2003.

[47] F. A. Khan, M. A. Tahir, F. Khelifi, and A. Bouridane, "Offline text independent writer identification using ensemble of multi-scale local ternary pattern histograms," in *Visual Information Processing (EUVIP), 2016 6th European Workshop on*. IEEE, 2016, pp. 1–6.

[48] I. Siddiqi and N. Vincent, "Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features," *Pattern Recognition*, vol. 43, no. 11, pp. 3853–3865, 2010.

[49] R. Kumar, B. Chanda, and J. Sharma, "A novel sparse model based forensic writer identification," *Pattern Recognition Letters*, vol. 35, pp. 105–112, 2014.

[50] G. Ghiasi and R. Safabakhsh, "Offline text-independent writer identification using codebook and efficient code extraction methods," *Image and Vision Computing*, vol. 31, no. 5, pp. 379–391, 2013.

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

[52] R. Jain and D. Doermann, "Combining local features for offline writer identification," in *2014 14th International Conference on Frontiers in Handwriting Recognition (ICFHR). IEEE*, 2014, pp. –.

[53] L. Schomaker and M. Bulacu, "Automatic writer identification using connected-component contours and edge-based features of uppercase western script," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 6, pp. 787–798, 2004.

[54] S. He, M. Wiering, and L. Schomaker, "Junction detection in handwritten documents and its application to writer identification," *Pattern Recognition*, vol. 48, no. 12, pp. 4036–4048, 2015.

[55] M. Bulacu, L. Schomaker, and A. Brink, "Text-independent writer identification and verification on offline arabic handwriting," in *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, vol. 2. IEEE, 2007, pp. 769–773.

[56] D. Chawki and S.-M. Labiba, "A texture based approach for arabic writer identification and verification," in *Machine and Web Intelligence (ICMWI), 2010 International Conference on*. IEEE, 2010, pp. 115–120.

[57] C. Djeddi and L. Souici-Meslati, "Artificial immune recognition system for arabic writer identification," in *Innovation in Information & Communication Technology (ISIICT), 2011 Fourth International Symposium on*. IEEE, 2011, pp. 159–165.

[58] M. N. Abdi and M. Khemakhem, "Arabic writer identification and verification using template matching analysis of texture," in *Computer and Information Technology (CIT), 2012 IEEE 12th International Conference on*. IEEE, 2012, pp. 592–597.

[59] M. N. Abdi and M. Khemakhem, "A model-based approach to offline text-independent arabic writer identification and verification," *Pattern Recognition*, vol. 48, no. 5, pp. 1890–1903, 2015.

[60] F. Slimane and V. Margner, "A new text-independent gmm writer identification system applied to arabic handwriting," in *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on*. IEEE, 2014, pp. 708–713.

[61] V. Christlein, D. Bernecker, F. Hönig, A. Maier, and E. Angelopoulou, "Writer identification using gmm supervectors and exemplar-svms," *Pattern Recognition*, vol. 63, pp. 258–267, 2017.

[62] S. Fiel and R. Sablatnig, "Writer identification and retrieval using a convolutional neural network," in *Computer Analysis of Images and Patterns*. Springer, 2015, pp. 26–37.

[63] X. Li and X. Ding, "Writer identification of chinese handwriting using grid microstructure feature," in *International Conference on Biometrics*. Springer, 2009, pp. 1230–1239.