3
State-of-the-Art : Off-line Writer Identification

This chapter reviews the state-of-the-art methods for offline writer identification. It enlists the diversity of the domain, the datasets being used, their characteristics and constraints. The organization of this chapter is as follows. Section 3.1 elaborates the problem of writer identification and its various divisions. It also presents the handwriting specific peculiarities that are exploited as features to differentiate among various writers. Section 3.2 reviews writer identification at the atomic layer of document content i.e. character or grapheme level. It reviews the literature in two broad categories of feature-based and codebook-based writer identification schemes. Section 3.3 reviews the features and models to identify writers of a complete document. Section 3.4 gives an overview of features and methods used for writer identification of handwritten words. Section 3.5 describes methods to identify the writer of a handwritten text-line. The chapter is concluded in Section 3.6 with a brief summary of the research attempts in this domain.

3.1 INTRODUCTION

Confucius philosophized, "Men's natures are alike, it is their habits that separate them." It implies that in nature, a person can be identified by its peculiar form of laughing, gesturing, walking, etc. People are predominantly creatures of habit and writing is a collection of those habits. This clarifies that writing is determined by the personality of its author, and hence, it pursues identification capability. Writing habits are neither instinctive nor hereditary but are complex processes that are developed gradually. All of these habits when considered in combination, constitute the means of discrimination among different handwritings. Writer identification reflects new insights into the evolving nature of handwriting research. It is a hard problem which attracts scientific research in the area of pattern recognition.

According to graphologists, even smaller groups of characteristics might serve to discriminate between writings. Therefore, the discriminating element of writing can be segregated into four broad categories.

1. Elements of style : Style is something which is influenced by a writer's artistic ability. It deals with the arrangement of text on paper either in sense of proportion or the instruction received while writing. The Style also defines the way two or more letters are united. The connecting stroke exemplifies the union letting it to be a distinguishable property among writing. Style also refer to the physical measurements of writing, including such terms as proportions, relative heights, size, relative sizes, and ratios. Spacing between capital letters and lowercase or small letters in the same words and between words also determines stylometry.

2. Elements of execution: It comprises abbreviations, alignment, commencements and terminations, diacritics and punctuation, embellishments, and line continuity. It basically tells the overall behavior of a writer while writing. It depends upon the personal instincts of the person.

3. Attributes of All Writing Habits: It records the consistency or natural variations in writing

multiple times in different durations. It means it captures the imprecision with which the habits of the writer is executed on repeated occasions. Significantly, it throws light on the intra and inter writing variations.

4. Combinations of Writing: It is the product of letter formation, letter sizes, and the spacing between letters and words. It ranges from contracted to expanded. It also focuses on the overall proportions of distinct words in a document. It is prominently applied for signature verification purposes.

Figure 3.1 presents the discriminating elements of writing that are habitual, individual, and of potential value in writer identification. In the domain of pattern recognition and machine learning,

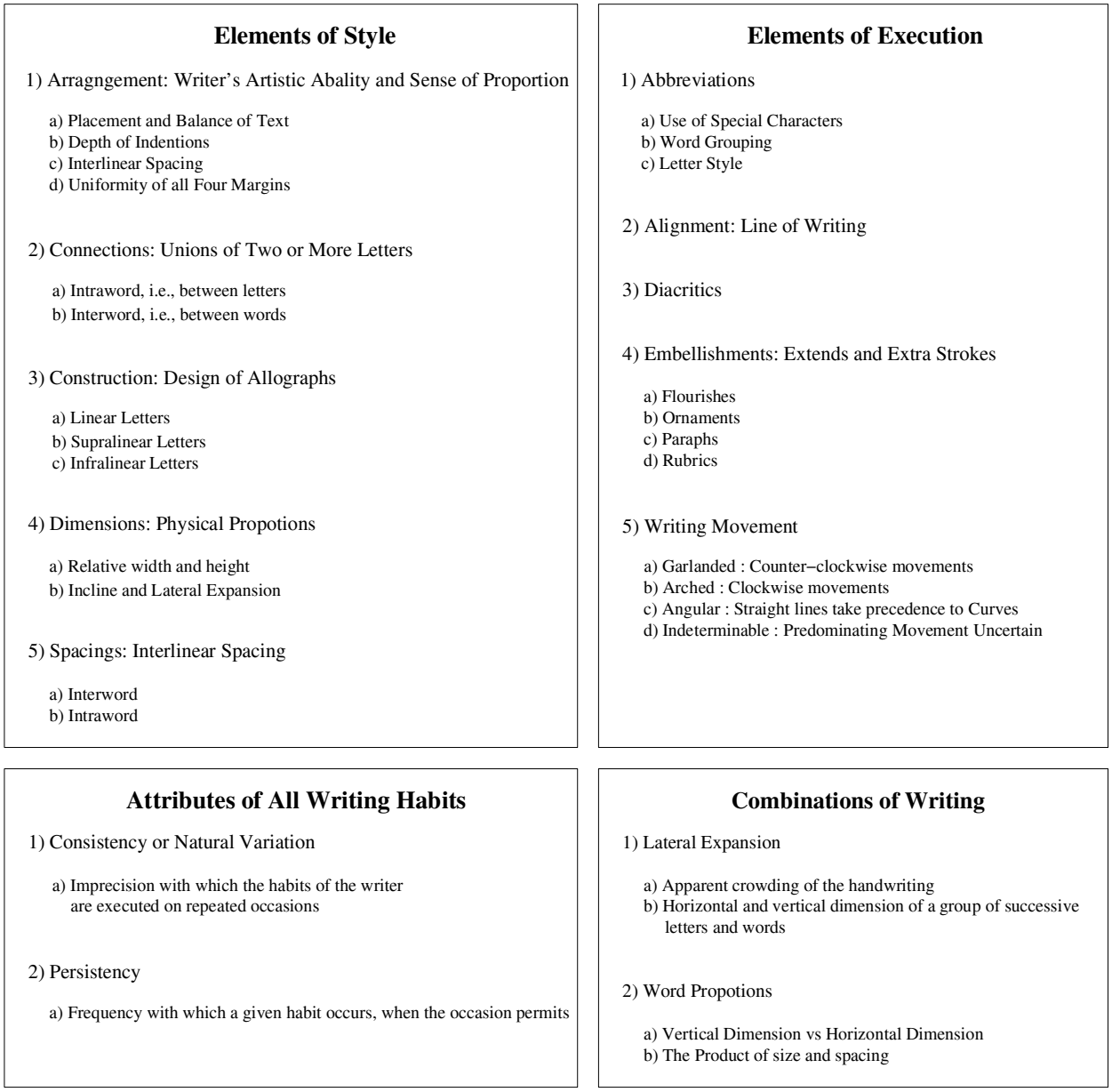

Figure 3.1 : Discriminating Elements in Handwriting.

the characteristics stated in Figure 3.1 are fully absorbed into features to distinguish the authors of the documents, lines, words, or characters.

Historically, the study of writer identification is explored within two broad categories : Text-dependent writer identification and Text-independent writer identification. In the former approach, the features encapsulate the characteristics of the writer based on the similar text content written by different writers. On the other hand, the latter approach captures the writer specific properties that are independent of any text written. Table 3.1 defines the contrast among the writer dependent and independent writer identification. It demonstrates the insight and the scope of applicability of text-dependent and independent writer identification methods.

Table 3.1 : Writer Identification: Dependent vs Independent Text.

	Writer Identification with Text Dependence	**Writer Identification with Text Independence**
1	Encapsulates the characteristics of a writer based on the similar text content.	Encapsulates the writer specific properties that is independent of any text content.
2	This approach is not extensible because of constrained text.	This approach is better suited for real world scenarios as they are scalable.
3	Their identification model, calculates global features (writing habits) to verify the writer.	Their identification model, calculates local features (peculiarities) to capture writer specific style.
4	Being dependent on same text content its potential applications are in banking and administrative sector particularly for Signature Identification and verification.	Being independent on text content its has huge potential applications are in any real world scenario particularly in forensic, academics and historic data indexing.

Identification of the writer on the basis of the writer's style is in high demand, due to immense growth in technology. Having said that the method of writer identification can be categories according to modes of documents: Scanned documents (off-line) and Temporal documents (on-line). In on-line documents, prior information in the form of strokes is available, while in off-line documents no such prior information is available. Moreover, there exists other challenges in off-line documents such as:

1. The degree of variability and variation of the script.
2. The problem of foreground/background segmentation in highly textured and smudged documents.
3. The limited amount of text in unknown samples.

Due to a large number of classes, the identification cannot be considered as a simple classification task. Therefore, a two-stage strategy must be used to come to a conclusion concerning the authenticity of an individual. The first stage is the writer identification task while the second one is defined as the writer verification task. The difference between them is stated in the Table 3.2. Over time, an extensive literature has developed and presented by [Arazi, 1977, 1983; Sreeraj.M and Idicula, 2011; Awaida and Mahmoud, 2012] on various writer identification schemes.

3.2 CHARACTER/GRAPHEME BASED WRITER IDENTIFICATION

When an individual writes, he draws similar characters using the same basic shapes. It is convenient to collect some characters of a person and then match them with the characters of the test document, provided that the segmentation should be perfect. Graphemes are constructed to resolve the manual segmentation of characters. The process of writer identification at character/grapheme level can be studied under following sub categories as:

1. Feature-based character and connected component/grapheme based writer

Table 3.2 : Distinction Between Writer Identification and Writer Verification.

	Writer Identification	Writer Verification
1	It is one-to-many search in a handwriting database that will return a likely list of candidates.	It is one-to-one comparison whether or not the two samples are written by the same person.
2	Involves executing a search in a database of documents on a basis of a small snippet of handwriting and returns a ranked list of results for the search.	Comparison between a questioned document and one or more documents from the same known writer in order to ascertain the authenticity or identity of the questioned document.

identification.

2. Codebook-based character and connected component/grapheme based writer identification.
3. Hybrid methods for writer identification.

3.2.1 Feature Based Writer Identification

3.2.1.1 Character Feature Based Writer Identification

Although character segmentation in handwritten documents is difficult, yet their results are fast for smaller problems and datasets. Table 3.3 presents an overview of the methods and features along with other information for feature based writer identification by analyzing handwritten characters.

[Wang *et al.*, 2003] presents a text sensitive method to identify the writers of Chinese characters. They investigate the usefulness of directional element features which are widely used for Chinese character recognition. In the first place, character's image is normalized by using a linear normalization method called gravity-center normalization. The normalization step aims to eliminate the difference of image size and at the same time, keep the writing styles of each person. After normalization, contour extraction algorithm is applied. Each contour pixel is assigned a 4-dimensional vector to measure the four types of directional elements specifically in horizontal, vertical, and two diagonal directions. For this, the given image is divided into two fixed sized windows and in that neighborhood, contour pixels are counted in each direction. The dimensions of the feature set are reduced using PCA. To cope up with the small sample size problem, the most discriminative features are extracted from the reduced feature space using Fisher's Linear Discriminant Analysis. The Euclidean distance is proposed for classification. The dataset consists of two sets: Set-1 having 34 Chinese characters and each was written 16 times by 25 people and Set-2 with 20 Chinese characters, each was written 16 times by 27 people. The effectiveness of the proposed method is 96.12% for Set-1 and 82.16% for Set-2.

[Leedham and Chachra, 2003a] applied a set of 11 computational features to identify authors of handwritten digits. These include aspect ratio, average height, number of junctions and endpoints in a digit, zero crossings, width and height distribution, pixel density, fixed point distance and angular measure; and loop property like degree of roundness, length, slant, fissure length. After extraction, the obtained feature vector is appropriately binarized to form a binary feature vector of constant length. Author discrimination is done using the Hamming distance measure. For this task, a writer database consisting of 15 writers was created and each writer was asked to write random strings of 0 to 9 at least 10 times.

The paper [Pervouchine and Leedham, 2007] presents a study of structural features

of handwriting extracted from three characters ``d'', ``y'', and ``f'' and a bigram ``th''. They concentrated on the extraction of the micro level features like height, width, height to width ratio, relative height of ascender, slant of ascender, final stroke angle, fissure angle, relative height of descender, descender loop completeness, descender slant, final stroke angle, slant at point, slant of t-stem, slant of h-stem, position of t-bar. Neural network is used as a classifier and genetic algorithm is applied for searching an optimal feature set for writer identification. The paper claims that the bigram possessed significantly higher discriminating power than any of the three single characters studied, which supports the opinion that a character form is affected by its adjacent characters. They reported an accuracy of 58% for 200 writers from 600 samples of the CEDAR letter dataset.

The work [Jain and Doermann, 2013] presents a novel feature that uses contour gradients to capture local shape and curvature of a character for writer identification. This novel descriptor is computed by extracting contour from the binarized image. For each point on the extracted contour, a combined gradient is calculated using a fixed size segment on either side of the point. Finally, a histogram of gradient orientations is formed by binning the gradients into eight orientations which forms the final descriptor. These features are clustered by k-means to form pseudo-alphabets for each writing sample. A unique distance measurement calculates the character similarity between two alphabets which determines the writer's similarity. This approach achieves a Top-1 identification rate of 96.5% on the benchmark IAM dataset and 98% on ICFHR 2012 dataset.

3.2.1.2 Grapheme/Connected Component Feature Based Writer Identification

Graphemes are basic units of handwriting that present allographic variations among the handwritten contents by different writers. Generally, graphemes are extracted by sliding windows of fixed or varying sizes over the upper contour of a word or connected component. [Bulacu and Schomaker, 2006] suggests a method based on contours for grapheme extraction. According to their method, grapheme is formed when a cut is made at lower contour where the distance between the upper and lower contour of the image is comparable to the ink width. Table 3.4 presents an overview of the methods and features along with other information for feature-based writer identification by analyzing handwritten grapheme/connected component.

[Bensefia *et al.*, 2005] presents a grapheme based writer identification model that instead of applying sequential clustering by k-means, incorporates SOM to cluster the similar graphemes. The writers are identified using an Information Retrieval model where each document is represented by a descriptor built using document frequency and inverse document frequency. The similarity between the query and reference document is calculated using cosine similarity function. The method can very well cope up with the unconstrained handwriting. The approach achieves 95% correct identification on the PSI DataBase and 86% on the IAM Dataset.

[Bulacu and Schomaker, 2006] worked at grapheme level for free handwriting in only lower case text. They claimed an improvement in writer identification rate by infusing both allographic and texture features. The Probability Density Function (*PDF*) was computed for features such as contour direction, contour hinge, contour co-occurrence, run length, and grapheme emission. They typically captures the stylometry of a particular writer as:

1. Contour direction PDF: It captures the handwriting slant for a writer.
2. Contour hinge PDF: It captures the orientation and the curvature of the contours.
3. Contour co-occurrence PDF: It gives the measure of the roundness of the written characters.
4. Run length PDF: It captures the regions/empty space enclosed inside the letters.
5. Grapheme emission PDF: It captures the stroke information of the character.

Table 3.3 : Overview of dataset, performance and methods applied for writer identification by means of feature based character shape analysis.

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Wang *et al.* [2003]	Set-1 having 34 Chinese characters and each was written 16 times by 25 people and Set-2 with 20 Chinese characters, each was written 16 times by 27 people	Directional features in horizontal, vertical, and two diagonal direction	Euclidean distance	96.12% for Set-1 and 82.16% for Set-2	Chinese
Leedham and Chachra [2003a]	15 Writers	Computational Binarized Features	Hamming Distance	…	Numerals
Pervouchine and Leedham [2007]	200 writers, 600 of CEDAR letter dataset	Structural Micro Features	Neural Network	58%	English
Jain and Doermann [2013]	MADCAT Arabic Dataset, ICFHR 2012 Greek Dataset, IAM English Dataset	Contour Gradient Descriptor	Distance Function	98% (ICFHR 2012), 96.5% (IAM), 87.5% (MADCAT)	Arabic, English, Greek

Their method yields a performance of 87% on 900 writers at document level, which is build by combining writers from Firemaker, IAM and ImUnipen datasets.

Preliminary work of [Bulacu and Schomaker, 2007], suggested that the curvature of the ink trace is a discriminatory characteristic between different writers. Therefore using such property, [He and Schomaker, 2014] proposed a new set of a rotation-invariant feature called Delta-Hinge feature based on Hinge feature that is proposed by [Bulacu and Schomaker, 2007]. To identify the writer Nearest-neighbor classifier with a "leave-one-out" strategy is used. Their method is tested on two datasets of different scripts. It produces a Top-1 performance rate of 89.2% for Firemaker dataset and 91.6% for IAM dataset.

3.2.2 Codebook Based Writer Identification

An individual who draws a particular shape (e.g. loops) in a specific way is expected to always employ the same (similar) patterns when drawing that shape, irrespective of which

Table 3.4 : Overview of dataset, performance and methods applied for writer identification by means of feature based grapheme/connected component shape analysis.

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Bensefia *et al.* [2005]	PSI Dataset, IAM Dataset	…	Grapheme based IR model with sequential clustering	95% PSI dataset, 86% IAM dataset	English
Bulacu and Schomaker [2006]	900 writers (Firemaker, IAM and ImUnipen)	Contour-direction, Contour-hinge, horizontal and vertical Grapheme emission PDFs	k-NN	87%	English
He and Schomaker [2014]	Firemaker, IAM datasets	Delta-n Hinge feature	NN	Firemaker (89.2%), IAM (91.6%)	English, Greek

character being written. With this approach and motivation codebook methods are created that clusters the similar/repetitive patterns representing class similarity and the number of clusters representing class variability.

3.2.2.1 Character Codebook Based Writer Identification

Table 3.5 presents an overview of the methods and features along with other information for Codebook Based Writer Identification by analyzing handwritten Character.

[Leedham and Chachra, 2003b] proposed a writer identification scheme by manually segmenting off-line handwritten characters. They performed two experiments on CEDAR database for 30 writers one with local codebook while the other with the global codebook. The local codebook is composed of sub codebooks of 52 types of characters. K-means is used as a clustering scheme for creating the codebook. For each writer, the nearest codebook is searched using Euclidean distance. Therefore, for each writer one histogram (in the case of global codebook) or 52 histograms (one per character, in the case of local sub-codebooks) are obtained. A PDF is computed from each whether common or sub codebooks, by histogram binning. This PDF is used to characterize each writer. Finally, writer identification is done by a distance function. Their work concludes that: i) working with local sub-codebook results in much better performance than using a unique single codebook, and ii) when some of the sub-codebooks are combined only slight difference in performance was seen.

In a similar manner, [Hu *et al.*, 2014] presents newly devised coding strategies that outperform traditional Bag of Word encoding with hard voting for writer identification in Chinese documents. For each Chinese character, local information is extracted by SIFT [Lowe, 2004] features which are encoded by Improved Fisher Kernels (IFK) and Locality-constrained Linear Coding (LLC) [Wang *et al.*, 2010]. Ultimately, K-NN classifier is used to identify the author of a handwriting image. Their method produces Top-1 accuracy of 95.42% when used LLC without

max-Pool and 96.25% when applied IFK as encoding schemes for 240 writers.

Table 3.5 : Overview of dataset, performance and methods applied for writer identification by means of codebook based character shape analysis.

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Leedham and Chachra [2003b]	CEDER database, 30 Writers	…	k-means, local and global codebook generation	…	English
Hu *et al.* [2014]	CASIA Offline DB 2.1	SIFT, Improved Fisher Kernels and Locality-constrained Linear Coding	k-NN Classifier	LLC(max-Pool) = 60% (top1), LLC = 95.42% (top1) ,IFK = 96.25% (top1)	Chinese

3.2.2.2 Grapheme/Connected Component Codebook Based Writer Identification

Graphemes are basic units of handwriting that characterize a peculiar style of a writer. We can expect to obtain the writer's particularities only when large samples of his handwriting can be collected. They are also addressed as fraglets or fragments. Table 3.6 presents an overview of the methods and features along with other information for codebook-based writer identification by analyzing handwritten grapheme/connected component.

[Schomaker *et al.*, 2004] used Moore's neighborhood algorithm to produce a family of character fragments called fraglets. They proposed a Fragmented COnnected-COmponent COntour (FCO^3) which is a fixed size feature vector for writer identification. Its computation involves an application of Moore's algorithm to drew contour from the connected components. It then deduces fraglets by applying heuristics. Over these deduced fraglets Moore's algorithm is reapplied and fixed size (FCO^3) is computed. Subsequently, a codebook is constructed by Kohonen self-organizing feature map (SOFM) [Kohonen, 1989] for the extracted fraglets. The writer specific features from the codebook for a test writer is evaluated for writer identification by hamming distance. Their method achieved an accuracy of 97% for same character set and 70% for mixed Character set for a dataset of 200 writers. They extended their work by incorporating edge based features along with the contour based features to identify writes for only upper case Western script [Schomaker and Bulacu, 2004]. With the inclusion of the edge feature, their method shows promising results. They achieved an accuracy of 87% for Top-1 and 98% for Top-10 retrievals on a dataset of 150 writers.

[Siddiqi and Vincent, 2010] questions the ability of writer-specific and universal codebooks for writer identification. They exploited contour-based orientation and curvature feature at different levels of observation to find the writer of the document. To employ the proposed segmentation scheme, small writing fragments from the text are extracted by sliding a window. The feature vector of every fragment comprises the horizontal and vertical histograms, upper and lower profiles and a set of well-known shape descriptors like orientation, eccentricity, rectangularity, elongation, perimeter, and solidity. The codebook is generated by grouping the redundant patterns using clustering methods like k-means and fuzzy means. For comparison

between two handwritten documents, a number of distance measures including: Minkowski, χ^2 Distance, Bhattacharyya, (Non-)Intersection and Hamming distance are used. On the bases of experiments, it is observed that universal codebook performs better than the writer specific codebooks. The rates achieved with writer-specific codebook is 81% and with universal codebook is 84% on IAM dataset. It also achieves a rate of 69% with writer-specific codebook and 74% with universal codebook on RIMES dataset.

[Jain and Doermann, 2011] uses K-adjacent segment [Ferrari *et al.*, 2008] features in a bag-of-features framework to model a user's handwriting. The K-adjacent segment (KAS) feature represents the relationship between sets of neighboring edges. In order to extract KAS features from a connected component, a set of edges must be found using a Canny edge detector. The codebook is generated using a clustering technique known as affinity propagation [He *et al.*, 2008]. The experiments were performed on IAM dataset with 301 writers and on MADCAT dataset with 325 writers. The conclusion drawn from the experiments are in three folds. First, for k=3 value the KAS features performs the best with an identification rate of 93.3% on IAM dataset and 90% on MADCAT dataset. Second, increasing the number of training samples is beneficial for the identification performance. Third, by the applicability of the KAS features the codebook generated is robust.

In handwriting, the stroke thickness varies from pen to pen, ink to ink and paper to paper. In view of it, [Paraskevas *et al.*, 2014] suggests reducing the stroke thickness down to one pixel by skeletonizing the text. Using a fixed squared window the graphemes are extracted and the codebook is built by Kohonen SOFM [Kohonen, 1989]. The main contribution of the system lies in introducing an improvement in edge directional features, which results in achieving an accuracy of 95.6% with Manhattan distance.

[Al-Maadeed *et al.*, 2014] presents a new method to extract graphemes. The skeletonized text is segmented at its junction pixels into elementary graphic units called graphemes. The codebook is generated by clustering the graphemes according to their distributions by matching to a set of predefined grids or templates. Chi-square distance is used to match the graphemes with the appropriate grid. Once the codebook is generated, a feature vector is created for each document image. To identify the writers the generated document descriptors are compared using Euclidean distance. The overall Top-1 accuracy achieved is 90.86% for ICDAR 2011 writer identification contest dataset.

3.2.3 Hybrid Approach for Writer Identification

The texture-level features and allograph-level features yields substantial results on different dataset when combined together as single descriptor. Their combination tends to increase the performance rate for about 4% to 5% for various datasets. Table 3.7 presents an overview of the hybrid approaches and features for Writer Identification.

The idea reported in [Bulacu and Schomaker, 2007; Bulacu *et al.*, 2007] states that a combination of texture-level features and allograph-level features would reveal the true distinguishing properties in writer identification task. The set of texture features includes contour probability density functions of hinge, direction and direction co-occurrence along with run-Length PDFs and autocorrelation. These features are concatenated with allographic-based writer specific features. A shape codebook is generated by grapheme clustering. In order to identify the writer k-NN classifier is used. A comparative study is established among three different clustering techniques for codebook generation: k-means, Kohonen SOM 1D, and 2D. They point out that same performance is achieved by all three clustering methods and that performance is stable over a large range of codebook sizes. The experiments are performed on three data sets: Firemaker, IAM, and ImUnipen. A Top-1 accuracy of 92% is achieved for the

Table 3.6 : Overview of dataset, performance and methods applied for writer identification by means of codebook based grapheme/connected Component shape analysis.

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Schomaker *et al.* [2004]	(215 writers) Firemaker Dataset	PDF of fragmented connected components	Codebook by Kohonen maps + Hamming Distance	97% (same character set), 70% (mixed Character set)	English (uppercase)
Schomaker and Bulacu [2004]	Firemaker Dataset (100 (training) + 150 (testing) writers)	PDF contour based features, edge based features	Codebook by Kohonen maps + Hamming Distance	87% (Top-1), 98% (Top-10)	English (uppercase)
[Siddiqi and Vincent, 2010]	IAM, RIMES	horizontal and vertical histograms, upper and lower profiles, shape descriptors (orientation, eccentricity, rectangularity, elongation, perimeter and solidity)	χ^2	81% (IAM) and 69% (RIMES) with writer-specific codebook, 84% (IAM) and 74% (RIMES) with universal codebook	English
Jain and Doermann [2011]	IAM dataset, MADCAT dataset	K-adjacent segment	BoW	93% (Top-1,IAM), 90.3% (Top-1,MADCAT)	English, Arabic
Paraskevas *et al.* [2014]	Firemaker Dataset (250 writer)	edge-direction distribution, edge hinge distribution, edge hinge combinations and skeleton hinge distribution	k-NN with Manhattan, Euclidian, chi square distance	95.6%	English
Al-Maadeed *et al.* [2014]	ICDAR 2011 Writer Identification Contest	Grid/ Template based Distributions	Euclidean Distance	90.86% (Top-1 all documents)	Greek, English, French, German

feature set combination of Contour-Hinge PDF and Run-Length PDFs. Similarly, [Bulacu *et al.*, 2007] investigates the same approach for on the IFN/ENIT dataset of Arabic handwriting with 350 writers, 5 samples/writer. They yielded a performance of 88% as Top-1 identification rates for the

feature combination fusing directional, grapheme and run-length features.

Likewise, [Jain and Doermann, 2014] demonstrates the advantages of utilizing multiple features for capturing complimentary attributes of handwriting. This combination entails features produced from segmentation-free methods such as SURF, that extract features from interest points; Edge-based KAS features extracted from character contours, and Contour Gradient features from allograph methods that aim to capture a character's shape and style. These local features are combined and pooled using Fisher Vector distances [Perronnin *et al.*, 2010]. It is observed that feature pooling using the Fisher Vector consistently outperform individual features. They claimed that feature pooling by Fisher Vectors outperforms the codebook method and reported significant results on four different types of datasets as: IAM with 356 writers and accuracy achieved 97.4% for English; ICDAR 2013 writer identification contest with 250 writers and accuracy achieved 97.4% for English; CVL dataset with 309 writers and accuracy achieved 98.3% for English; and MADCAT dataset with 325 writers and accuracy achieved 98.5% for Arabic.

Table 3.7 : Overview of dataset, performance and methods applied for writer identification by means hybrid approaches.

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Bulacu and Schomaker [2007]	Firemaker, IAM, and ImUnipen dataset	Contour PDFs of hinge, direction and direction co-occurrence along with run-Length PDFs and autocorrelation	k-NN	92%	English
Bulacu *et al.* [2007]	IFN/ENIT dataset (350 writers, 5 samples/writer)	Contour PDFs of hinge, direction and direction co-occurrence along with run-Length PDFs and autocorrelation	k-NN	88%	Arabic
Jain and Doermann [2014]	IAM (356 writers), ICDAR 2013 (250 writers), CVL dataset (309 writers), MADCAT dataset (325 writers)	SURF, KAS features, Contour Gradient Features	Code book generation on Fisher Vectors	IAM 97.4%, ICDAR 2013 97.4%, CVL dataset 98.3%, MADCAT 98.5%	English, Arabic, Greek

3.3 DOCUMENT BASED WRITER IDENTIFICATION

Table 3.8 presents an overview of the methods and features along with other information for Document Based Writer Identification.

[Said *et al.*, 1998, 2000] proposed a global text independent approach based on texture analysis for writer identification. According to their approach after preprocessing and image normalization, texture features are obtained. These include multi-channel Gabor Filtering (MGF) [Tan, 1992] and the grey-scale co-occurrence matrix (GSCM) [Haralick, 1979]. Weighted Euclidean distance (WED) and the k-NN are used as classification scheme and it is observed that with WED classifier the performance rate increases by 10%. Results of 96.0% accuracy on the classification of 1000 test documents from 40 writers are very promising. Their research showed that multichannel filtering outperforms the GSCM technique but it is computationally expensive. Their work inspired [Shahabi and Rahmati, 2006] to find writers for Farsi handwritten documents. They apply multi-channel Gabor filtering and co-occurrence matrix features with Weighted Euclidean distance as a classification scheme. The correct identification is achieved at 88% Top-1 for a dataset of 25 images.

Handwriting holds a fundamental property of inter-writer invariants, between patterns representing distinct letters. The work [Bensefia *et al.*, 2002] exploits this property by forming clusters of similar graphemes. The graphemes are extracted by applying template matching using the correlation measure. They are clustered together using a sequential search. A similarity score is then computed between the clusters of the writer's sample document and the clusters of test document. The writer producing the maximum similarity score is selected as the writer of the given test document. For experiment 88 writers were allowed to write three documents consisting 107 and 98 words. They showed that writer identification can reach a correct identification rate of 92.9% using only samples of 50 graphemes of each writing.

[Zhang *et al.*, 2003] presents a unique method involving character segmentation and its matching to identify the document writer. According to their method, 62 characters were segmented from each handwritten document. The micro- features comprising of 512 bits were extracted that corresponds to gradient and structural (each 192 bits), and concavity (128 bits). Thus for each writer, they construct a 512 dimension feature vector. To measure the similarity between two binary vectors, the Correlation measure is used. The method produced an accuracy of 97.83% when the training set includes 2206 documents and the testing set consists of 875 written by 875.

Reduction in computational cost plays an effective role in any model generation. With this view, [He *et al.*, 2005] presents a novel wavelet-based Generalized Gaussian Density (GGD) method and replaced the traditional 2-D Gabor filters to identify document's author. In GGD method, the handwriting image is first decomposed by wavelet transform at 3 sub-band levels of different resolution instead of frequencies as in Gabor filters. Then GGD is applied on each wavelet decomposition sub-band for feature generation. The experiments are performed on 20 Chinese documents authored by 10 persons each writing 64 characters. Only one document is used for training and testing. A test image sub-band features are matched with all training handwriting images sub-band features using Kullback-Leibler Distance. The results obtained for each writer are sorted in an ascending order to produce a list. It produces an accuracy of 80% with an elapsed average time of 19.38 sec in comparison of Gabor filter with an accuracy of 70% with a elapsed average time of 124.812 sec. This work is extended by replacing GGD with Hidden Markov Tree (HMT) model to generate characteristic features for sub-band images [He *et al.*, 2008]. Kullback-Leibler Distance used as a similarity measure for test and reference images. Their work produced an identification rate of 97.83% for top-300 matches on 1000 Chinese handwritings provided by 500 persons.

[Abdi *et al.*, 2009] presented a novel approach for text- independent Arabic writer identification. They proposed a novel stroke-based feature set that uses Borda count ranking algorithm as a classification scheme. Their method incorporates stroke segmentation for thinned image. Their feature set comprises length, height/width ratio; PDF of stroke length/ratio and their horizontal and vertical cross- correlation, stroke length/curvature and their horizontal and vertical cross- correlation, and stroke length/curvature and length/ratio cross-correlation. Their method presented an identification rate of 92.5% for Top-1 on 40 writers from the IFN/ENIT database.

[Helli and Moghaddam, 2009] proposed a text independent method to identify a writer by examining the sorted order of the feature values. According to the method, two Gabor and XGabor filters are applied in different directions and a feature set is extracted. The features in the feature vector are sorted by magnitudes to make SOF (Sorted Order of Features). The final writer is identified by Longest Common Subsequence algorithm that selects the writer which have the longest sub sequence common in the input vectors. The method was tested on two different databases. With 100 documents in Persian dataset, the performance rate achieved is 77% and with 30 documents of IAM English dataset 80% performance rate is achieved. Later, they used Gradient features and applied MLP classifier to find the writers for Persian dataset in [ram and Moghaddam, 2009]. This scheme produced a rate of 94% for 250 documents penned by 50 writers with a ratio of 5 pages/writer.

In order to speed up the retrieval of writer for a particular document, [Djeddi *et al.*, 2012] aimed is to reduce the search space for the same. For this, they proposed to use retrieval mechanism for writer identification on multi-script documents. This is achieved by using two sets of texture features and city-block distance to measure the similarity among the feature vectors. Accordingly, run-length features are extracted from the document to identify a fixed top nearest documents to the query document. Among the top nearest retrieved documents, edge hinge features are extracted and a top nearest document is retrieved as the corresponding writer. Their work reports an improvement in identification rates form 83.5% to 92.4% for the IFN/ENIT database, from 97.6% to 99.5% for the GRDS database and from 85.3% to 93.3% for the mixed database.

[Chanda *et al.*, 2012] proposes a writer identification system for Oriya script. They used curvature feature which is computed by using bi-quadratic interpolation method as described in [Shi *et al.*, 2002]. While processing, from a document, characters/graphemes are extracted and their curvature features are trained over SVM. Finally, on the application of voting scheme, the given document is classified to a particular writer to which it belongs. Promising results of 94.00% at Top-1 and 99% Top-3 accuracy are achieved on Oriya script using the above methodology.

The work [Chen and Lopresti, 2012] presents a novel method to address the writer identification problem for noisy handwritten documents that are written on a substrate of pre-printed ruling lines. Instead of attempting to remove rulings and to recover broken strokes, they incorporate rulings to help with the identification task. For this, a new displacement feature descriptor is devised based on the upper and lower profiles in each word. This feature is concatenated with contour-hinge feature descriptor and is inputted in SVM classifier for writer discrimination. The experiments are performed on Arabic dataset involving 61 writers and 4,890 handwritten text lines. Their method showed an effective improvement of 10% performance gain over the baseline system which attempts to remove ruling lines.

For Indic script as in Bangla, [Biswas and Das, 2012] proposed an approach that can identify the writer by using lesser amount of information from the handwritten samples. According to their method, CC are extracted from document images and are divided into two sets, on the account of a threshold based on height. Each set will possess its own set of Radon transform projection profile features for each of the reference documents. Euclidean distance is applied for matching a test

feature set from both the reference sets. Finally, a score is computed against each writer and on its maximization, the writer is identified. It achieved an accuracy of 83.63% Top-1 retrieval.

In one of the recent works, [Fecker *et al.*, 2014a] presents writer identification in historical Arabic manuscripts. Their work mainly aimed to presents a comparative contrast on the effects and performance rate of contour, textual, and key-point features upon various classification schemes. Their feature set includes orientation, contour, and key-point based feature descriptors. They used three different classification schemes: voting, aver- aging, and weighted voting. On experimentation, it is observed that key point descriptors yield a perfect identification in combination with a weighted voting scheme.

In any trainable writer identification system, to any query which not a part of training data, ideally no writer should be assigned. But this does not happen and always most similar writer is assigned to the query irrespective of its inclusion in training set. For such circumstances, [Fecker *et al.*, 2014b] contributed to learning based rejection strategy which means to reject the decision i.e., no appropriate writer candidate is found. Their feature set includes orientation, contour, and key-point based features and an SVM classifier is applied to select the appropriate writer. The decision of rejection is finalized if the distance between first two writes selected by SVM is below a certain threshold. This threshold is set heuristically or can be automatically computed with a learning-based approach stated in [Aksela *et al.*, 2001]. Their work presents an acceptance and rejection accuracy of 80.2% on a multi-script document and 84.1% on single script document.

In one of the recent works, [Fiel and Sablatnig, 2013] identified the writer of the document using Fisher Vectors. According to their approach, a vocabulary is created by clustering features using a Gaussian Mixture Model. The features are SIFT local descriptors. For each writer a Fisher Vector is generated using the vocabulary. Finally on distance measurement similarity among the test and the reference vector writer's decision is made. The proposed method is evaluated on two datasets, namely the ICDAR 2011 Writer Identification where an accuracy of 99.3% is reported and the CVL dataset that reports an accuracy of 95.6%. In a subsequent work [Fiel and Sablatnig, 2015], suggested to use CNN to form descriptors for each writer. These descriptors are generated by chopping off the second last fully connected layer. The nearest neighbor classifier is used for classification. The authors claimed a soft criteria performance of 88.5% on ICDAR-2013 dataset, 99.5% on ICDAR-2011 dataset, and 98.9% CVL dataset.

The work reported in [Christlein *et al.*, 2014] represents an individual writer by GMM supervector encoding method. According to their method, a Universal Background Model (UBM) is constructed from a sift variant RootSIFT [Arandjelović and Zisserman, 2012]. It clusters the RootSIFT features from training documents. Descriptors for independent writers are encoded which are GMM supervectors. These vectors are matched using a distance function to identify the writers. On close observation, it is found that GMM supervectors descriptor outperforms the other encoding schemes, namely Fisher vectors and Vectors of Locally Aggregated Descriptors with a rate of about 2%. They reported an accuracy of 97.1% for ICDAR-13 dataset and 99.2% for CVL dataset. In their subsequent work [Christlein *et al.*, 2015], they proposed to use Zernike moments that are evaluated at the contours, as a local feature descriptor for writer identification. These local feature descriptors are encoded into Vectors of Locally Aggregated Descriptors (VLAD) global descriptors which represent a writer. The similarity among the global descriptor of the writers is measured by cosine similarity. This local/global descriptor combination yields a mAP of 0.880 on the ICDAR 2013 benchmark database and 0.671 on CVL dataset.

Drawing inspiration from [Djeddi *et al.*, 2012], in one of the recent works, [Xiong *et al.*, 2015] applies a two-stage retrieval scheme with SIFT key-point descriptors. According to their method, in the first stage, a codebook of SIFT keypoint descriptors is built using k-means clustering scheme.

Chi-square distance measure is used to compare the histogram of encoded features to generate a sorted candidate list of most probable writers. Then in the second stage, contour directional feature is applied to select the appropriate writer from the candidate list. The method presents an accuracy of 94% for ICFHR-12 English dataset and 96.2% for ICDAR-13 Greek dataset.

[Fréry *et al.*, 2015] formalized writer identification problem as a supervised clustering problem and proposed three kinds of clustering to solve it. These are DCM (Dissimilarity counter method), DCM clustering, DCM voting. The feature space is represented by document specific features like characters, words and n-gram sequencing, punctuation marks, number of words in a sentence and term-document frequency of words. The experiments are performed on PAN CLEF 2013, 2014 dataset with 50 and 796 documents. Among the three clustering used DCM voting turns out to be the best with an F-score of 76.7% on 2013 dataset and 90% on 2014 Dutch documents.

[Nicolaou *et al.*, 2015] applies Sparse Radial Sampling Local Binary Patterns to identify the writer of documents. NN classifier is used as a classification scheme. The experiments are performed on CVL and ICDAR 2013 datasets and produce an overall accuracy of 97.4% on ICDAR-2013 dataset and 99.4% on CVL dataset.

Table 3.8 : Overview of dataset, performance and methods applied for writer identification of a document by means of local and global features.

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Said *et al.* [1998]	150 documents,10 writers	Grey Scale Co-occurrence Matrices, Multi-channel Gabor Filtering	k-NN, weighted Euclidean classifier	96.0% with WED, 82.2% with k-NN	English
Said *et al.* [2000]	1000 Documents, 40 Writers	Multi-channel Gabor Filtering features and the grey-scale co-occurrence matrix	Weighted Euclidean distance, k-NN	96% Top-1	English
[Shahabi and Rahmati, 2006]	25 Documents, 25 Writers	Gabor features and Gray level co-occurrence feature	Weighted Euclidean distance	88% Top-1	Farsi
Bensefia *et al.* [2002]	88 Writers	Graphemes, sequential clustering	Template matching/ similarity score	92.9%	English
Zhang *et al.* [2003]	3000 documents, 1000 writers	Micro-features	Correlation measure	97.83%	English
He *et al.* [2005]	20 documents,10 writers each writing 64 characters	wavelet-based GGD	Kullback-Leibler Distance	80%	Chinese
[He *et al.*, 2008]	1000 documents(500 persons)	Hidden Markov Tree	Kullback-Leibler Distance	97.83%	Chinese

Continued on next page

Table 3.8 -- *Continued from previous page*

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Abdi *et al.* [2009]	IFN/ENIT database (40 writers)	length, height/width ratio; PDF of stroke length/ratio and their horizontal and vertical cross- correlation, stroke length/curvature and their horizontal and vertical cross-correlation, and stroke length/curvature and length/ratio cross-correlation	Borda count ranking algorithm	92.5% Top-1	Arabic
Helli and Moghaddam [2009]	100 Writers (Persia), 30 writers (English)	Gabor, XGabor Filter	LCS-based classifier	77%(Persian), 80%(English)	Persian, English
[ram and Moghaddam, 2009]	250 documents, 50 writers, 5 page/writer	Gradient Features	MLP classifier	94%	Persian
[Djeddi *et al.*, 2012]	1583 writing samples, GRDS datasets, IFN/ENIT datasets	Probability distributions of run-lengths, edge-hinges features	Manhattan Distance	83.5% to 92.4% (IFN/ENIT database), 97.6% to 99.5% (GRDS database), 85.3% to 93.3% (mixed database)	Arabic, German, English, French, Greek
Chanda *et al.* [2012]	Self-made dataset, 100 writers, 80 words	Curvature feature by bi-quadratic interpolation method	SVM	94.00% (Top-1), 99% (Top-3)	Oriya
Chen and Lopresti [2012]	61 Writers (4,890 handwritten text lines)	Contour Hinge Feature	SVM	67.67%	Arabic

Continued on next page

Table 3.8 -- *Continued from previous page*

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Biswas and Das [2012]	BESUS database (55 Writers)	Radon transform projection profile	Euclidean Distance	83.63% (Top-1)	Bangla
Fiel and Sablatnig [2013]	CVL dataset, (1539 documents, 309 writers), ICDAR 2011 Writer Identification (208 documents, 26 writers)	SIFT	GMM for codebook generation and Fischer vector	99.5% for soft criteria, 96.2% for hard criteria	English, Greek, German, French
Fecker *et al.* [2014a]	Islamic heritage dataset (4595 pages)	Modified Contour-Based Features, Oriented Basic Image Features, SIFT	k-NN weighted voting scheme	80.2% (multi-script document), 84.1% (single script document)	Arabic manuscripts
Fecker *et al.* [2014b]	Islamic heritage dataset	Modified Contour-Based Features, Oriented Basic Image Features, SIFT	SVM	80.2% (multi-script document), 84.1% (single script document)	Arabic manuscripts
Christlein *et al.* [2014]	CVL,ICDAR-13	RootSIFT	GMM supervector encoding method, Distance function	97.1% (ICDAR-13 dataset), 99.2% (CVL dataset)	English
Christlein *et al.* [2015]	ICDAR-2013 dataset, CVL dataset	Zernike moments on contours and encoding them to Vectors of Locally Aggregated Descriptors (VLAD)	Cosine Distance	.880 mAP (ICDAR-13), 0.671 mAP (CVL)	English
Xiong *et al.* [2015]	ICFHR-2012, ICDAR-2013	SIFT, Contour Directional Feature	Chi-square distance	94% (ICFHR-12, English), 96.2% (ICDAR-13, Greek)	English, Greek

Continued on next page

Table 3.8 -- *Continued from previous page*

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Fréry *et al.* [2015]	PAN CLEF 2013,2014 dataset with 50 and 796 documents	document specific features like characters, words and n-gram sequencing, punctuation marks, number of words in a sentence and term-document frequency of words	DCM, DCM clustering, DCM voting	F-score 76.7% 92013 dataset), 90% 2014 Dutch documents	English, Greek, Spanish, Dutch
Nicolaou *et al.* [2015]	ICDAR-2013 dataset, CVL dataset	Sparse Radial Sampling, LBP	k-NN	97.4% (ICDAR-13), 99.4% (CVL)	Greek, English
Fiel and Sablatnig [2015]	ICDAR-2011 dataset, ICDAR-2013 dataset, CVL dataset	CNN features	k-NN	88.5% (ICDAR-2013), 99.5% ICDAR-2011, 98.9% CVL dataset	Greek, English

3.4 WORD BASED WRITER IDENTIFICATION

There exists a few set of documents that contains sparse handwritten text like signatures in bank cheques, annotation in official documents and content in form processing. There the handwritten text is usually drawn out as words. For such situations word based writer identification is required. Table 3.9 presents an overview of the methods and features along with other information for Word Based Writer Identification.

In the work presented by [Zois and Anastassopoulos, 2000], writers are identified for words using novel morphological directional features. These features represent the distribution of the pixels of a word along the direction of projection. To compute the distribution in a particular direction a set of fixed size structuring elements are matched. They applied trapezoidal window on the word image to find the feature vectors. A Neural Network is trained which achieves an accuracy of 96.5% in English and 97% in Greek.

To find a set of individual words which best characterize a person's handwriting style, [Long Zuo, 2002] proposes to use PCA for writer identification. The feature vector is formed over a gray-scale handwritten word image by combining the rows together. PCA is applied to a set of gray-scale features to find the characteristic feature of a particular writer. Finally, the unknown writer vector is matched with the reference vectors by vector distance mechanism to identify the writer. Their method obtained a performance of 97.5% on 400 pages containing 16000 Chinese words written by 40 different writers.

[Zhang and Srihari, 2003] used GSC (Gradient, Structural and Concavity) features on words and performed writer discrimination and verification. Their study majorly focused only on four characteristic words, ``been'', ``Cohen'', ``Medical'', and``referred''. Their dataset comprises 1000

writers who wrote only these four words on three documents and reported an accuracy of 83%.

[Siddiqi and Vincent, 2007] identified the writer with only grapheme features and reported an identification rate of 94% at the document level. To extract graphemes, the handwritten text is divided into a large number of small windows of fixed size. Then a correlation similarity measure is used to cluster similar graphemes. Each document is then modeled as a Gaussian Mixture of grapheme codewords. Bayes decision theory is employed for document classification.

[Al-Máadeed *et al.*, 2008] presents a novel method to identify the writer of an Arabic word. They used edge based directional probabilities and image statistics like moments, area, length, height, length from baseline to upper and lower edge as features to identify the writers. The classification decision is made by k-NN classifier. The paper reports an accuracy of 93.8% for long words and 53.4% for short words for a dataset comprising 100 writers.

For Indian languages especially for Telugu script, [Pulak Purkait and Chanda, 2010] drew inspiration from [Zois and Anastassopoulos, 2000] and used morphological directional features for writer identification. They built a dataset of 22 writers where each writer writes 10 words. Every word is described by three directional features of opening, closing, and erosion. Along with the directional features, curvature feature are also concatenated to the obtained feature vector. A k-NN classifier with Euclidean distance is trained with 4 documents. The results revealed that the directional opening feature outperforms the other features in identification with an accuracy of 71.73%. In combination with all the features, the reported accuracy for five words is 90%.

[Chaabouni *et al.*, 2011] performed word level writer identification by combining the on-line and off-line features. For the same word, the same writer will have two images one offline and the other online in the test dataset. Using both word images high density of information points are identified and then they are surrounded by a box. The points identified are called fractal and the features extracted from them are termed multi-fractal features. The experiments are performed on 100 writers of ADAB database on 25 words and following conclusions were drawn:

1. The on-line fractal features (84.6%) outperform the off-line fractal features (80.9%).
2. The combination of on-line and off-line fractal produces higher accuracies (93.2%).
3. To characterize the styles of writings on-line are better because they are more informative.

~~As a person can be identified by its peculiar habits, similarly writing is determined by the personality of its author. Based on this idea~~ [Vásquez *et al.*, 2013], presents a new approach for writer identification using graphometrical and forensic features. These features include computation of angle and height of ascenders and descenders, degree of unity, distance between strokes, roundness of writing, and micro features. The method is trained on the LS-SVM classifier with RBF kernel. The performance reported by them reaches a success rate of 99.1% for a dataset penned by 100 writers with 10 samples of 34 words per each one.

[Slimane and Margner, 2014] claimed to achieve an accuracy of 23.03% at word level and 69.48% at line level for writer identification. They used sliding window approach to extract graphemes and avoided manual segmentation. A GMM is built for each writer of the AHTID dataset with 53 writers. Their work also states the comparison of using GMMs instead of HMMs for writer identification and also states that writer identification by a single word is more complex than by a single text line.

Table 3.9 : Overview of dataset, performance and methods applied for writer identification of a document by means of word property analysis.

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Zois and Anastassopoulos [2000]	…	Morphological Feature	NN	96.5% (English), 97% (Greek)	Greek, English
Long Zuo [2002]	400 Documents,16000 words, 40 Writers	PCA based Image gray value feature	Vector Distance	97.5%	Chinese
Zhang and Srihari [2003]	12,000 words, 1000 writers (CEDAR 4 words,	GSC features	k-NN	83%	English
Siddiqi and Vincent [2007]	50 documents, IAM	block based features, Grapheme based	Bayesian classifier	94%	English
Al-Máadeed *et al.* [2008]	100 Writers	Edge based directional features	kNN classifier	Short words = 53.4%, Long words = 93.8%	Arabic
Pulak Purkait and Chanda [2010]	22 Writers, writes 10 words each, 4 document/writer (Training), 1 document/writer (testing)	Directional opening, directional closing, direction erosion and k-curvature	NN classifier	82.70%	Telugu
Chaabouni *et al.* [2011]	ADAB dataset 100 writers	Multi-Fractal (Online, offline) Features	k-NN	93.2% (Top-1)	Arabic
Vásquez *et al.* [2013]	100 writers (10 samples with 34 words/writer)	Graphometrical and Forensic features	LS-SVM classifier with RBF kernel	99.1%	Spanish

Continued on next page

Table 3.9 -- *Continued from previous page*

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Slimane and Margner [2014]	AHTID database, 53 writers, 3710 text lines, 22,896 words	Mean and Standard deviation of vertical and horizontal projection, Derivate of horizontal projection vector profile, Mean and Standard deviation of vertical and horizontal runs, Typological features	GMM	69.48% (Top-1) for 4096 Gaussian,	Arabic

3.5 LINE BASED WRITER IDENTIFICATION

In most of the applications like postal addresses, form processing, etc. the handwritten text is segmented in the form of text lines. Therefore, for such systems, text-line writer identification methods are required. Table 3.10 presents an overview of the methods and features along with other information for Line Based Writer Identification.

[Marti *et al.*, 2001] proposed the application of statistical features for the same. They used a set of twelve features including width, the slant and the height of ascender, core and descender zones. The results are produced by training two classifiers: k-NN and feed-forward Neural Network for 100 pages of text written by 20 different writers. An average recognition rate of 87.8% for the k-NN and 90.7% for the neural network is measured.

Text Normalization is a pre-processing step that removes intra and inter variability in handwriting. It is beneficial for many recognition and segmentation problems. To explore its effect on the domain of writer identification, [Schlapbach and Bunke, 2005] confronted the subject of text-line writer identification. For each writer in the considered population, an individual HMM based handwriting recognition system is trained using only data from that writer. They all have the same architecture, but their parameters, i.e., transition and output probabilities, are different because they are trained on different data each. Intuitively, each HMM can be understood as an expert specialized in recognizing the handwriting of one particular person. In order to extract features, a window is slided over the text-line and nine geometrical features, three global and six local features are extracted. The global features are the fraction of black pixels in the window, the center of gravity and the second order moment. The local features represent the position of the upper and the lower-most pixel, the number of black-to-white transitions in the window, and the fraction of black pixels between the upper and lower-most black pixel. Using these features, an input text line is converted into a sequence of 9-dimensional feature vectors. They compared three normalization techniques: slant correction, width normalization, and vertical scaling. It is observed that the best writer identification rate of 97.70% is obtained when the slant correction and width normalization are not applied. Later, the same idea is extended in the work [Schlapbach and Bunke, 2007] where writer verification along with identification is incorporated. [Schlapbach and Bunke, 2008] replaced HMM with GMM to represent the distribution of features extracted

from the text lines of a writer. Using a similar feature set GMM is built and trained on text lines of each writer. With log-likelihood score the writers are identified for the feature set consists of three global and six local features. Their method produced an identification rate of 97.88% on IAM text-line dataset that is comparable with HMM.

[Cao *et al.*, 2010] applies speaker selectivity of speech recognition to writer identification. They applied a two-fold process to implement this adaptability. First, handwriting of the most likely writer is selected from the training set using a writer identification algorithm based on k-NN and voting method. Then MAP adaptation technique is applied to decode the document using the writer dependent HMM models of the identified writer class. A set of style features and profile features are extracted which produces an accuracy of 57.4% on Arabic dataset for 37.6K documents written by 259 writers as the training set.

Many real-world problems are complex in nature and can be divided into sub-tasks. For such mechanism structural learning offers common optimal structure to all auxiliary tasks to a given problem. Based on this, [Porwal *et al.*, 2012] focused the identification of a writer typically by structural learning. According to it, the writer identification problem is broken to binary classification problems equal to a number of writers. SVM is used as the classification scheme. The method produced an accuracy of 81.34% on IAM 93 writer dataset.

[Daniels and Baird, 2013] used text line features to identify the writer of the text line or a complete passage. The features slant and slant energy, skew, pixel distribution, curvature, and entropy is classified to a respective writer by a k-NN classifier. The experiments are performed on combined IAM and ICDAR 2011 datasets comprising 50 and 100 writers respectively. The paper reports accuracy in hard and soft parameters. In soft identification rate, an error is said to have occurred when a document by the same writer does not appear. In hard identification rate, an error is said to have occurred when a document by a different writer appears. For soft results, the top-1 identification rate is 97.1% and for hard results, the top-1 rate is 92.8%.

Writer Identification is seen as a multi-class learning problem. One of the fundamental approaches to solve a multi-class problem is by breaking it into binary classification tasks. With this inspiration [Porwal *et al.*, 2014] proposes a generic approach for multi-class classification using an ensemble of binary classifiers. In the ensembles binary classifiers, each classifier predicts one bit of the codeword. Hence, a bitwise output from each classifier will form a codeword for a particular writer. Thus for each writer gets assigned to a codeword of length equal to the number of classifiers used. The GSC features are extracted from each text-line. To encode the descriptor probabilistic error correcting code method is applied. Further MRF with belief propagation is used for decoding the output of the classifiers on the test image. Finally, hamming distance is used to find the appropriate nearest writer. The experiments are conducted on the publicly available IBM-UB-1 dataset with 41 writers and 3714 pages and produces an average performance of 66.87%.

In one of the recent works, [Alaei and Roy, 2014] models the handwriting style through an individual set of histograms. According to the method, after pre-processing, for each extracted text-line, a set of 92 features are computed based on analysis of connected component, enclosed region, lower and upper contours, fractal code, and Curvelet. A histogram is created for all feature values of individual writer using a histogram-valued symbolic data algorithm. Distance measure is used to find similarity/dissimilarity among the feature set. The writer of the document is predicted by majority voting over the labels of the text-lines comprising the document. To evaluate the proposed scheme, two different handwritten datasets written in two different scripts. The first dataset contains 228 pages written in Kannada by 57 people. The other one is the dataset used in SigWiComp2013 composed of 330 document pages written in English by 55 individuals. Concerning the Kannada dataset, an F- measure of 92.79% was obtained when 114 documents

used as training and testing each while for the SigWiComp2013 dataset an F-measure of 26.67% was obtained.

In the most recent work, [Xing and Qiao, 2016] presents DeepWriter, a multi-stream CNN architecture to extract writer- sensitive features. DeepWriter takes multiple image patches of fixed size as input and train by fine-tuning its parameters. For the identification of writer, a text image is broken down into patches and inputted to DeepWriter. A vector of dimension equivalent to the number of writers is generated where each feature value represents a score as a probability distribution over all writers. On averaging this score for each patch and maxim value will signify the writer of text-line or a character. The proposed method is evaluated on IAM dataset and HWDB1.1 dataset. They achieved a performance rate of 97.3% on 657 writers from the IAM dataset and 93.85% on 300 writers from HWDB1.1 dataset.

Table 3.10 : Overview of dataset, performance and methods applied for writer identification of a document by means of text-line property analysis.

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Marti *et al.* [2001]	IAM dataset	12 statistical features	k-NN, NN	87.8% k-NN, 90.7% NN	English
Schlapbach and Bunke [2005]	4,307 text lines, 100 writers from IAM dataset	geometrical features, global and local features	HMM	97.70%	English
Schlapbach and Bunke [2008]	IAM Dataset (54 text lines, 1500 pages)	Three global features: distribution of the pixels in the column, the center of gravity and the second order moment. Six local features: position and the orientation of the upper and the lower-most pixel, the number of black-to-white transitions in the window, and the fraction of black pixels between the upper-and the lower-most black pixel	GMM	97.88%	English

Continued on next page

Table 3.10 -- *Continued from previous page*

System	Dataset	Feature Set Used	Identification Scheme	Performance	Language
Cao *et al.* [2010]	37.6K training documents with 259 writers, 767 testing documents with 35 writers	Differential features, contour features, pen pressure, connected component-related statistics, contour slopes	k-NN, Mahalabonis Distance, HMM	57.4%	Arabic
Porwal *et al.* [2012]	IAM dataset 4075 lines, 93 writers	GSC, contour angle	SVM with RBF kernel	81.34%	English
[Daniels and Baird, 2013]	IAM, ICDAR 2011 datasets	slant and slant energy, skew, pixel distribution, curvature, entropy	NN	97.1% for soft criteria, 92.8% for hard criteria (for entire dataset)	English, Greek, French, German
Porwal *et al.* [2014]	IBM-UB-1 (41 writers)	GSC	Error correcting codes and belief propagation, k-NN	66.87%	English
Alaei and Roy [2014]	288 pages Kannada dataset (57 writers), 330 pages of SigWiComp2013 (55 writers)	connected component, Enclosed region, Lower and upper contours, Fractal code, Basic information, Curvelet	Histogram valued symbolic Modeling	F-score= 92.78% (Kannada), F-score= 26.67% (English)	Kannada, English
Xing and Qiao [2016]	IAM dataset, HWDB1.1 dataset	Deep CNN features	Average Scoring	97.3% (IAM), 93.85% (HWDB1.1)	Englsh, Chinese

3.6 CONCLUSION

Writer identification is an important step in the interpretation and analysis of annotated documents in forensic, biometric, academic and omni-writer applications. This chapter presented a summary of the state-of-the-art methods for every aspect of offline writer identification problem. We presented a brief categorization of the discriminating elements of handwriting in Figure 3.1. It is observed that due to prevalent idiosyncrasies in handwriting, a system must be coupled with two qualities:

1. A robust consistent set of prototypes to represent the basic shape of the handwriting of a writer.
2. Measures to subside the inherent variants and intensify singularity in handwriting.

Writer identification at character-level is mostly used for text-dependent systems. These approaches are likely to be adopted for smaller datasets or most specific texts like in banks for signature verification, administrative documents like numerals in tender fill documents. Due to the difficulty in segmentation of characters, this is not practiced as a top-notch identification scheme, and we are inclined to apply graphemes based analysis. Codebook methods are created to cluster the similar/repetitive patterns representing class similarity and the number of clusters representing class variability. They are intended to increase the adaptability of the system for inter and intra writer's character variabilities. Although grapheme analysis adds independence and produces substantial performance over character yet it is not popular and not many works exist in this domain. There are a few hybrid methods that combine features extracted from characters and graphemes to identify the writers for identification. Recent studies collectively outline the methods to identify the writers of the documents, words, and text-lines based on the application requirements. Another factor that enhances the execution of a writer identification system is search space reduction. It is been also observed that most of the methods do not have any reject option for an unknown and unreferenced writer. This gap has been identified by the [Fecker *et al.*, 2014b], yet more to be explored in this direction. In Chapter 8 we present application of graphemes and characters to identify the writer of a handwritten word. We employ sliding window analysis and voting mechanism to compare the performance between grapheme and character-based identification. The voting scheme identifies the writer and it also offers a reject option.

In the next chapter we describe our proposed method based on spectral partitioning for handwritten and printed text classification.

…