Al al-Bayt University

Prince Hussein Bin Abdullah College for Information Technology

Computer Science Department

Offline Writer Identification for Arabic Handwriting Texts Based on a Scale Invariant Feature Transform (SIFT)

تحديد هوية كاتب النص العربي المكتوب بخط اليد باستخدام خوارزمية تحويل صفة الصورة الغير مرتبط بمقياس

By

Mohammad Mahmoud Ali Al-Sheyab

Supervisor

Dr. Khaled Batiha

Co-supervisor

Dr. Atallah Shatnawi

May, 2017

Dedication

This thesis is dedicated to my wonderful parents. I also dedicated it to my wife, my children, my brothers and my sisters who always support me. And for everyone who encouraged and helped me.

Acknowledgement

First, I thank my God and praise Him who gave me the ability to complete this work. And I would to thank all of people who made this thesis possible.

I would like to thank my supervisor Dr.Khaled Batiha for his valuable guidance, encouragement and for being always there when I need him. Also I would to thank Dr.atallah shatnawi for his support. And I would like to thank all of my teachers who taught and helped me.

Also I would like to thank my father, my mother, my wife and every one of my family for their continuous encouragement. Finally, I would like to thank my friends for their love and support.

List of Contents

Dedication II
Acknowledgement III
List of Contents IV
List of Tables VI
List of Figures VII
List of Abbreviations IX
Abstract X
Chapter One: Introduction 1
1.1 Motivation of Thesis 5
1-2 Problem Statements 6
1-3 Research Contributions 7
1-4 Organization of Thesis 8
Chapter Two: Background and Related Work 9
2.1 Related Work 10
2.2 Survey of SIFT Algorithm and Clustering Algorithms 15
2.2.1 SIFT Algorithm 16
2.2.2 K-Means Clustering Algorithm 22
2-2-3 The K-Nearest Neighbors (*k*-NN) Matcher 23
Chapter Three: Research Methodology 26
3.1 Theoretical Studies 27
3.2 Determine the Dataset 27
3.3 Defining the Arabic Writer Identification Issues 27
3.4 The Proposed System 27
3.5 Comparison and Discussion 28
Chapter Four: System Implementation 30
4.1 The Dataset 30
4.2 Data Pre-Processing 32
4.3 Features Extraction 32

4.4 Codebook Generation 33
4.5 Feature Matching 35
4.6 System Implementation 36
Chapter Five: Results and Discussions 45
5.1 Results 45
5.2 Discussions 53
Chapter Six: Conclusions and Future Works 58
6.1 Conclusions 58
6.2 Future Works 60
References 61
Arabic summary 72

List of Tables

Table No.	Page No.	Description
Table 1	39	RR for writer identification performance obtained with 150 centers.
Table 2	41	RR for writer identification performance obtained with 300 centers.
Table 3	43	RR for writer identification performance obtained with 600 centers.
Table 4	45	Comparison of the RR averages for writer identification performance for all writers with 150, 300 and 600 centers.
Table 5	48	Comparison between RR of our proposed system (SIFT and K- means using K-NN matcher) and (Chawki and Labiba, 2010) and other two methods for them (Combination Black GLRL, White GLRL and GLCM Features, GLCM Features).

List of Figures

Figure No.	Page No.	Description
Figure 1	2	Writer identification procedure.
Figure 2	16	Difference-of-Gaussian (Lowe, 1999).
Figure 3	17	The neighbors of the pixel (Chergui and Kef, 2015).
Figure 4	19	Part (a) shows 16x16 windows around key-point. Part (b) Shows the 128 dimensional vectors (8x4 x4) (Panchal, et al., 2013).
Figure 5	20	K-means clustering flow chart.
Figure 6	22	The process of K nearest neighbor matcher (Bazmara and Jafari, 2013).
Figure 7	23	Research methodology of our proposed system.
Figure 8	25	The proposed system framework.
Figure 9	27	Examples of handwritten samples for different writers (a, b and c).
Figure 10	29	The process of codebook generation.
Figure 11	30	The process of feature matching.
Figure 12	31	Our developed windows application Form.
Figure 13	32	The process of assigning the number of centers.
Figure 14	33	The process of collecting the descriptors.

Figure 15	34	The process of codebook generation completed.
Figure 16	35	The process of uploading image.
Figure 17	36	The process determining the threshold value.
Figure 18	37	The process of recognizing the input image and the number of matched features for all writers.
Figure 19	40	The RR with 150 centers.
Figure 20	42	The RR with 300 centers.
Figure 21	44	The RR with 600 centers.
Figure 22	45	Comparisons of RR average with 150.300 and 600 centers.
Figure 23	49	Comparison between RR of our proposed system and (Chawki and Labiba, 2010).

List of Abbreviations

Abbreviation	Meaning
SIFT	Scale Invariant Feature Transform
KNN	K-Nearest Neighbors
SD	SIFT Descriptors
DOG	Difference-of-Gaussian
RR	Recognition Ratio

Abstract

The recent and ongoing improvements in telecommunications, financial, and industrial fields create remarkable needs for reliable and easy to use authentication system. These systems are broadly used in many ubiquitous applications including: banking, information dissemination, and online and electronic trade systems.

Writer identification systems are one of the most common authentication systems currently used. In spite of the huge development in writer identification systems, Arabic writer identification has not been studied as Latin or Chinese writer identification until the last few years. Arabic Writer identification systems' development faces many challenges, including the characteristics of Arabic writing, noise effectiveness, text thinning as well as the contours or the allograph of handwriting.

In this thesis, we propose an Arabic offline text-independent writer identification system based on the Scale Invariant Feature Transform (SIFT) algorithm and the k-means clustering algorithm. The system consists of two stages: training and identification stages. In the training stage, the SIFT descriptors (SDs) are extracted from the input handwritten samples, and then the k-means clustering algorithm is applied on these SDs to produce a centers for each writer and store them in the codebook. In the identification stage, the

SDs are extracted from the test input handwritten sample and matched with the ones in the codebook for identification by using *k*-nearest neighbors matcher (K-NN).We used the IFN/ENIT database in our work.

In this thesis, a comparison between three cases was applied by changing the centers of SIFT descriptors clusters that were produced by applying k-means clustering algorithm; the first case with 150 centers, the second one with 300 centers and the third one with 600 centers.

The results showed that the best case was when using 300 centers where the recognition ratio (RR) of identifying the writer was 81% as Top 1, 86% as Top 2 and 94% as Top3, where Top 1 means that the system retrieves the correct writer for the sample as the first candidate, Top 2 means that the system retrieves the correct writer for the sample as the second candidate, Top 3 means that the system retrieves the correct writer for the sample as the third candidate. And these results were compared with (Chawki and Labiba, 2010) and achieved better recognition ratio.

Chapter One: Introduction

The development of image processing technology and its applications can be applied to solve the personal identification problem, which is considered as one of the noticeably challenged problems. Most of the traditional ways of personal identification (e.g. PIN, Keys, etc.) did not success and caused fake authentication, because they may be shared, lost or stolen (Ubul, et al., 2012). Therefore, the need to apply rapid, non-traditional and strong authentication way has increased to maintain high of confidentiality and security of our life.

Biometric systems are among of the most reliable authentication systems. There are two kinds of biometric characteristics: physiological (e.g. face, fingerprint etc.) and behavioral (e.g. handwriting, gait, voice etc.) (Zhu, et al., 2000). Among different types of biometric systems, writer identification systems are considered one of the most popular behavioral biometric systems and recently they are a very active area of research (Kanade, et al., 2005; Jain, et al., 2006).

Writer identification is the process of identifying who is the writer of the handwritten text based on his/her character-writing features. It has been studied in wide areas, such as security, financial activity, forensic, digital rights management, decision-making systems and to solve the expert problems in criminology (Louloudis. et al., 2011). As a result, it is appealing enough to each industry and academia (Said, et al., 2000; Srihari, et al., 2002; Schomaker and Bulacu, 2004; He, et al., 2008).

Writer identification systems have essential advantages over traditional authentication systems since everyone has specific features which couldn't be stolen easily, also they have the advantage of easy to access, cheap and reliable (He, et al., 2007; Ubul, et al., 2012).

The writer identification system performs a one-to-many search in the dataset with handwriting samples of known writer, and then the system should suggest a list of candidates that have samples most similar to the one in testing based on feature matching (Siddiqi and Vincent, 2009; Chawki and Labiba, 2010). Figure 1 shows the identification systems procedure.

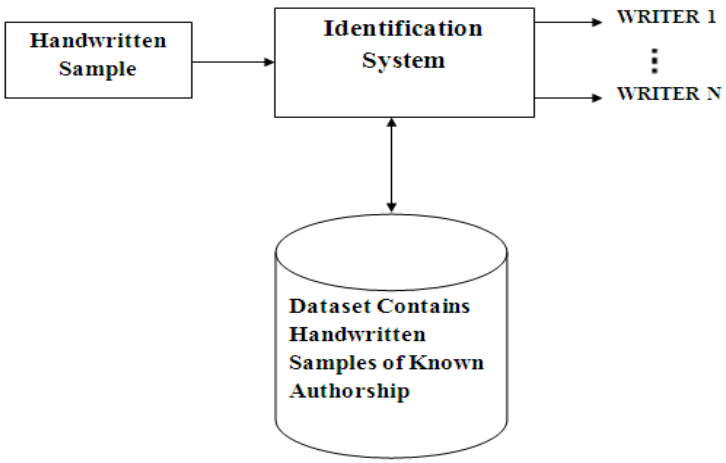


Figure1: Writer identification procedure.

Figure 1 shows the process of writer identification systems, where after the handwritten text is entered to the system to identify its writer, the system matched with database that contains all writers with their own features to retrieve the candidates writers based on the similarity between their features and the input handwritten text features.

Any writer identification system contains: data acquisition, pre-processing, feature extraction, and decision making or classification. And these are the main stages for all writer identification systems (Ahmed and Sulong, 2014).

In data acquisition stage the handwritten document is entered into the system using one of the ways that can provide this technology (e.g. scanning process). After that handwritten attributes are gained automatically to be represented as features.

In preprocessing stage the information is set up to get better accuracy of the system and performs the writer identification properly. In general, one or all of the following steps are followed for offline writer identification systems; applying filter to eliminate the noise and undesired area, converting the colored image to gray-scale image or black and white image, resizing the image which may cause a huge loss in image information.

Feature extraction stage is a critical part of writer identification systems; since in this stage the system extracts unique information for every input handwritten image and this information is called features, which can be either global or local (Palhang and Sowmya, 1999). Global features describe properties of complete handwritten image (Bouletreau, et al., 1998; Said, et al., 2000; Siddiqi and Vincent, 2008) such as (slope, density of thinned image, width to height ratio and skeweness… etc). While local features represent the distribution of the pixels of handwritten image that are deriving in the manner a

writer specifically writes characters (Bulacu, et al., 2003; Bensefia, et al., 2005) such as (slant angle, black pixels etc).

Classification stage is the last stage of the system, where Handwriting images provided by every one of the candidates are used to train the classification task. When classification is accomplished, another handwriting images are used to test the accuracy of system.

Writer identification can be classified into on-line and off-line based on the input method of writing (He, et al., 2005; Halder, et al., 2016). In on-line, the writing behavior is directly captured from the writer and converted to a sequence of signals using a transducer device but in off-line the handwritten text is used for identification in the form of scanned images (Saranya and Vijaya, 2013; Al-Maadeed, et ai., 2016). The on-line problem is usually easier than the off-line problem since more information is available about the writing style of a person such as speed, angle or pressure (Schlapbach, et al, 2008).

Text-dependent and text-independent are the other classification of automated writer identification Dependent on the text content (Plamondon and Lorette, 1989; Pavelec, et al., 2008). Text-dependent only matches the same characters and the writer should write the same text (Bulacu and Schomaker, 2007; Halder, et al., 2016). In text-independent any text can be used to establish the identification of a writer (Sreeraj and Idicula, 2011). In general, Text dependent systems accomplish better performance than text independent systems with very small amount of writer data (Pavelec, et al., 2008).

Offline text-independent writer identification can be classified into two categories: texture-based methods and structure-based methods (Wu, et al., 2014). Texture-based methods take the handwritten texts as a unique texture image then detect and extract the textural features for identifying the writer where the structure-based methods usually depend on the contours or the allograph of handwriting. Recently, the most studies in writer identification focused on the structure-based methods because they are much more intuitionist, stable and notable than texture-based methods (Thasneem and Febina, 2015).

Our proposed system will be focused on Arabic offline, text independent, local features and structure-based method.

In the next sections we will illustrate the motivation of our proposed system, the cause of selecting this problem to resolve and our contribution that we've performed to resolve this problem.

1.1 Motivation of Thesis

Arabic writer identification has not been addressed as Latin or Chinese writer identification till the last few years (Maliki, 2015) although the noticeable large improvement in writer identification systems.

Arabic is spoken by Hundreds of millions people around the world and it's the language of the holy QURAN. Arabic script characters and similar characters are used by means of a much higher percentage of the world's population to write languages such as Arabic, Farsi (Persian), and Urdu.

Thus, the potential to automate the interpretation of written Arabic might have extensive advantages.

Arabic Writer identification systems development faces many challenges including the noise effectiveness, the text thinning, the text skewness and the characteristics of Arabic writing especially the contours or the allograph of handwriting; it is easily affected by the slant.

However, when the document is written, the words are usually taken as an entire and the structures of them are usually stable and have a robust discriminate for different writers. Therefore, the structures between allographs inside the same word are important for determining the individuality of the writer (Wu, et al., 2014).

To deal with these problems, we applied a scale invariant feature transform (SIFT) algorithm on Arabic handwriting samples.

1-2 Problem Statements

The primary challenge that faces researchers in writer identification research area is to detect the features of the handwritten sample. Over the years, all preceding researches have attempted to find a way to extract the important features of the handwritten samples that will enhance the performance and give a good result, because the primary purpose is to increase the accuracy of the identification approach (Al-Ma'adeed, et al., 2008).

Arabic text writing characteristics is another challenge for selecting the suitable features to determine the correct writer - especially the allograph of the handwriting or the contours - where Arabic letters have many shapes and can be written in distinctive manner depends on its position in the word (Benjelil, et al., 2009). In additional Arabic writer identification systems' enhancement faces many challenges such as noise effectiveness and skewness.

1-3 Research Contributions

In this research, we proposed an Arabic offline text-independent writer identification system based on Scale Invariant Feature Transform (SIFT) algorithm and k-means clustering algorithm using *k*-nearest neighbors matcher (*k*-NN).

SIFT is one of the most popular algorithms that used to extract the descriptor from image because it has many properties such as invariant to scale change, invariant to rotation change, invariant to illumination change, robust to substantial range of affine transformation, and highly distinctive for discrimination (Lowe, 2004).

K-means clustering algorithm is applied on SIFT descriptors (SDs) to make the number of features extracted by SIFT limited and fixed for each writer.

1-4 Organization of Thesis

This thesis consists of six chapters; **Chapter 1** presents a general introduction of writer identification system. **Chapter 2** is background and literature review chapter which gives the scientific background of writer identification systems, illustrate SIFT algorithm and k-means clustering algorithm. **Chapter 3** is research of methodology chapter that presents the design of our offline writer identification system that based on SIFT algorithm and k-means clustering algorithm using the *k*-nearest neighbors matcher *k*-NN. **Chapter 4** explains the system implementation that shows how the designed system is implemented and tested. **Chapter 5** is results and discussion chapter that shows the results and analyzed them. **Chapter 6** shows the main conclusions of this thesis and presents some of possible future works.

Chapter Two: Background and Related Work

Writer identification process of Arabic language has not been addressed as Latin or Chinese writer identification till the last few years (Bulacu, et al, 2007; Maliki, 2015). However, most of the Arabic writer identification methods are extracted the features (local or global) from the handwriting document that can be used to determine the writer (Ahmed and Sulong, 2014).

In the next section a review of related works of writer identification systems will be explained to show how these systems have improved during the last few years. After that, we present a section that explained how the descriptors are extracted from images using Scale Invariant Feature Transform (SIFT) algorithm.

The second section is about the k-means clustering algorithm which illustrates how the K-means clustering works and explains the aim of the clustering.

The third one is about the *k*-nearest neighbors matcher (*k*-NN) which explains how does it works.

2.1 Related Work

Recently, writer identification is a very active area of research. Many researchers presented and developed different systems to identify the writer.

Shahabi and Rahmati (2006), presented a strategy for offline text-independent writer identification based on Farsi/Arabic handwriting, the features were extracted from preprocessed handwriting documents depends on multi-channel Gabor filtering and co-occurrence matrix features. For evaluation their proposed method, they selected 25 persons to evaluate their strategy.

Bulacu and Schomaker (2007), proposed an effective method for text independent writer identification and verification that use probability distribution functions (PDFs) extracted from the handwriting document to determine the correct writer. Their method depends on the textural and the allographic features. The PDF represent the characteristic of the writer and is computed using a codebook obtained by clustering. The best performance of writer identification and verification accomplished was by the combination of some textural and allographic features.

Gazzah and Amara (2007), proposed a new method for Arabic writer identification. Where they extracted the handwriting texture analysis by combining Global features (2-D discrete wavelet transforms) and Local features (space among sub-words, and features extracted from dots, line height). Experiment was used 180 samples obtained from 60 different writers as dataset and a modular multilayer perceptron as classifier.

AL-Dmour and Zitar (2007), proposed a new method for feature extraction based on hybrid spectral–statistical measures (SSMs) of texture. They compared their method with multiple-channel (Gabor) filters and the grey-level co-occurrence matrix (GLCM). The maximum discriminate features were selected with a model for feature selection using hybrid support vector machine–genetic algorithm techniques. Arabic handwriting texts for 20 different persons and Four classification strategies (linear discriminate classifier (LDC), support vector machine (SVM), weighted Euclidean distance (WED), and the K nearest neighbors (K_NN) classifier) had been used within the Experiments.

Al-Ma'adeed et al. (2008), presented a new method for Arabic text-dependent writer identification. They firstly applied the normalization process to the word, then they extracted the features (heights, lengths, and areas) from the word images which were considered as edge-based directional features and it also contain three edge-direction distributions include several different size. They created a new dataset collected from 100 writers. And WED used as a classifier in the experiment. The best result of 90% was obtained when 3 words were implemented in the top-10.

Chawki and Labiba (2010), presented a new Arabic off- line Text-Independent writer identification and verification method. Where they implemented a texture classification approach primarily based on a set of new proposed features extracted from Grey Level Run Length (GLRL) Matrices. The IFN/ENIT Database was used in the experiment.

Siddiqi and Vincent (2010), presented a novel method for writer recognition, their method depends on two different ways of writing, redundant patterns in the writing and its visual attributes. They segmented the handwriting image into small fragments with a fixed window and then generated codebook based features (The codebook and contour features were combined together) to represent different writers. The Arabic handwriting IFN/ENIT database was used in the system with nearest neighbor classifier.

Lutf, et al. (2010), proposed an efficient method for Arabic writer identification. They segmented the input handwritten text into two parts. The first one is for the letters and the second one for is for the diacritics. The local binary pattern LBP histogram for every extracted diacritic from the input handwritten text had been calculated to be use as features. They used the IFN/ENIT database in the experiments.

Helli and Moghaddam (2010), presented a text independent system for Persian writer recognition. Features are extracted using Gabor and Xgabor filter. The FRG (feature relation graph) was used to represent the extracted features for every person. The success rate for its ability to determine the correct author was 98%.

Al-Ma'adeed (2012), used the same method as used in Al-Ma'adeed et al. (2008) except changing the classifier to K-nearest neighbor classifier. The success rate for the top-10 writers was more than 90%.

Djeddi et al. (2013), explained the sensitivity of codebook-based writer recognition methods of the patterns in the codebook. They firstly explained that a codebook created from a different script than those of writings under study achieved identification rates substantially approaching those of the classical codebook based methods. This method was evaluated by using a set of database in Arabic, French, English, German, Urdu and Greek.

Wu, et al. (2014), proposed an efficient text-independent writer identification system depends on scale invariant feature transform (SIFT), it contains three main stages: training, enrollment, and identification. In which two SIFT features, SIFT descriptors signature (SDS) and scale and orientation histogram (SOH) are extracted from handwritten images to determine the individuality of the writer. The hierarchical Kohonen SOM clustering algorithm and six public data sets (including three English data sets, one Chinese data set, and two hybrid-language data sets) were used in the experiment.

Newell and Griffin (2014), proposed a new technique depends on Oriented Basic Image Feature Columns (oBIF Columns). They described how (oBIF Columns) can be used for assigning the writer and how this texture-based scheme can be enhanced by encoding a writer's style. Delta encoding provides a more informative encoding than the texture-based encoding. IAM dataset was used in the experiment.

Sreerag et al. (2015), proposed a novel of offline text document authorization using multiple feature extraction method scale invariant feature transform (SIFT) and speed up robust feature (SURF). It composed of two stages enrollment and identification. In all stages SIFT descriptors are extracted the scale and orientation (SOs) of the each sentences. At the same time SURF will extract the scale and orientation (SOs) of the same word. These extracted features are stored in the code book. SVM classifier is used for measuring the accuracy of the both SIFT and SURF algorithm. Experimental results consist of different English data sets (Iam, Firemaker etc).

Thasneem and Febina (2015), proposed an efficient Offline text-independent writer identification method depends on scale invariant feature transform (SIFT). It includes three stages: training, enrollment, and identification stages. SIFT descriptors signature (SDS) and scale and orientation histogram (SOH) are extracted from handwritten images to determine the individuality of the writer. They used the K-means clustering algorithm to find the correct writer. And they evaluated their work by using six public data sets.

In this section, we provided a study of related works for writer identification systems. This study demonstrated different procedures that used to evolve these systems through the last years such as Delta encoding oriented Basic Image Features (oBIF Columns), local binary pattern histogram (LBP) for each diacritic, Height area, length and Edge –direction distribution, contour-based orientation and curvature features, Gabor & Xgabor filters, etc.

In this research we propose Arabic offline text-independent writer identification system that based on SIFT algorithm and k-means clustering algorithm using *k*-NN matcher. This will be reviewed in the next sections.

2.2 Survey of SIFT Algorithm and Clustering Algorithms

Recently, different feature descriptors have been proposed. For example, the Gaussian derivatives descriptor (Florack, et al., 1994), the complex features descriptor (Baumberg, 2000), the phase based local features descriptor (Carneiro and Jepson, 2003) and the moment invariants descriptor (Mindru, et al., 2003). However, in 2004, a descriptor that outperforms the other descriptors was proposed, this descriptor is Scale Invariant Feature Transform (SIFT) (Lowe, 2004). It has a distinctive power to fend the effects of localization errors off (Mikolajczyk and Schmid, 2005) by applying the stable interest point detector in scale space. Then, it computes the histogram of the local oriented gradients around the interest point to locate the key points. Finally, rotation invariant descriptors are constructed.

Also several clustering methods have been proposed such as fuzzy c means clustering (Dunn, 1973; Bezdek, 2013) and support vector machines (Cortes and Vapnik, 1995) and K-means clustering (MacQueen, 1967) which considered the most common algorithm uses an iterative refinement technique (Jamnejad, et al., 2014).

2.2.1 SIFT Algorithm

Lowe (1999), proposed a new algorithm called Scale Invariant Feature Transform (SIFT). Where his algorithm (SIFT) analyses an image based on Gaussian scale-space and generates descriptors at minimum and maximum in the difference-of-Gaussian function of two adjacent scale space images. That was the initial implementation of the SIFT algorithm.

Lowe (2004), developed his work to extract stable features from the image which can be used to identify the object. These features are invariant to rotation and scale. Also they are robust against illumination changes and noise. That means the features can be detected in image although the object has been rotated or its distance has been changed.

SIFT algorithm consists of four main stages; scale-space extrema detection, key-point localization, orientation assignment and key-point descriptor.

In scale-space extrema detection stage, SIFT decomposed the source image into scales and octaves. SIFT takes the source image and create a new images from it with different blurring levels (each new image is less blurring than the previous one) and the set of different blurring levels of the image is called octave. Then, SIFT resize each image on the octave to half size and the set of different sizes of each image in the octave is called a scale. And SIFT keep repeating.

“Blurring” is referred to as the convolution of the Gaussian operator and the image:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y), \tag{1}$$

where L is a blurred image, G is the Gaussian Blur operator, I is an image, x and y are the location coordinates, σ is the "scale" parameter and The * is the convolution operation in x and y.

Then the SIFT uses scale-space extrema in the difference-of-Gaussian (DOG) function convolved with the image pyramid to detect stable key-point locations in scale space efficiently (Lowe, 1999). As shown in figure 2, the difference-of-Gaussian function convolved with the image, D(x,y,σ), which can be computed from the difference of two nearby scales separated by a constant multiplicative factor k:

$$\begin{aligned} \boldsymbol{D(x, y, \sigma)} &= \big(\boldsymbol{G(x, y, k\sigma)} - \boldsymbol{G(x, y, \sigma)}\big) * \boldsymbol{I(x, y)} \\ &= \mathbf{L(x, y, k\sigma)} - \mathbf{L(x, y, \sigma)}. \end{aligned} \tag{2}$$

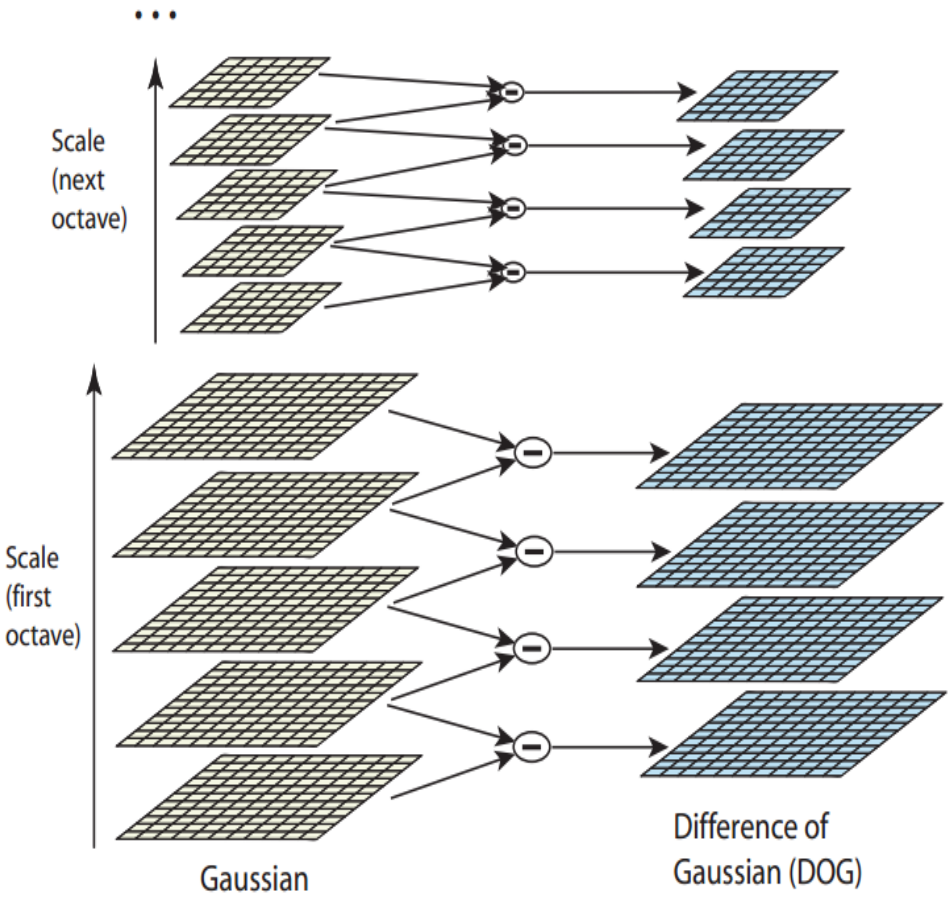


Figure 2: The Difference-of-Gaussian (Lowe, 1999).

In key-point localization step and after the DOG are calculated, each pixel in the image is compared with its 26 neighbors pixels, 8 of them in the same level, 9 pixels in the above level and 9 pixels in the below level. If the pixel has the maximum or the minimum value among all the 26 neighbors' pixels, it is considered as key-point. As shown in figure 3.

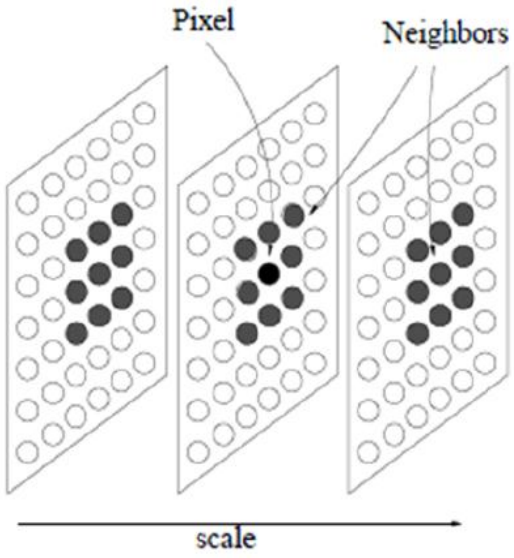


Figure 3: The neighbors of the pixel (Chergui and Kef, 2015)

The numbers of the extracted key-points are too many and some of them are not needed, therefore, SIFT uses some ways to eliminate edges and low contrast regions, it uses the quadratic Taylor expansion of the scale-space function to get more accurate location of key-point. Taylor expansion is shown in the following equation:

$$\boldsymbol{D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 y}{\partial x^2}} \quad (3)$$

where x is the offset, and D is the DOG scale space function.

Also if the value of DOG for any key-point is less than a threshold value (0.03), it is rejected (Lowe, 2004). Therefore, only the interest key-points are selected

In orientation assignment step and after deleted the unwanted key-points, the orientation is assigned to each interest key-points to get invariance to image rotation. The magnitude and orientation are calculated for all pixels around the key-point using these formulae:

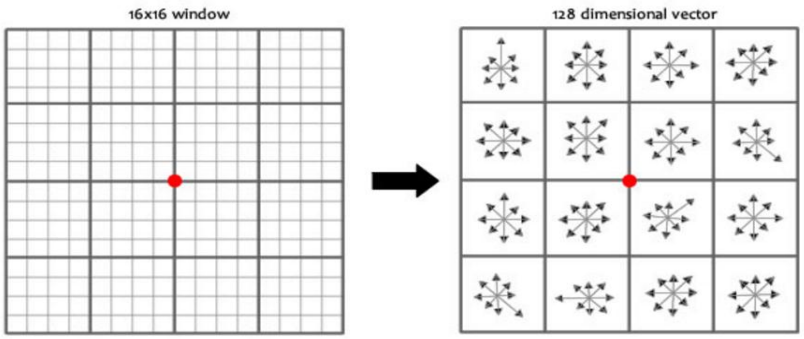
$$\boldsymbol{m(x,y)} = \sqrt{(\boldsymbol{L(x+1,y)} - \boldsymbol{L(x-1,y)})^2 + (\boldsymbol{L(x,y+1)} - \boldsymbol{L(x,y-1)})^2}$$

$$\boldsymbol{\theta(x,y)} = \mathbf{tan}^{-1}((\boldsymbol{L(x,y+1)} - \boldsymbol{L(x,y-1)}) \,/\, (\boldsymbol{L(x+1,y)} - \boldsymbol{L(x-1,y)})) \quad (5)$$

where m(x, y) is the magnitude and θ(x, y) is the orientation.

In the histogram, the 360 degrees of orientation are divided into 36 bins (one for each 10 degrees). The highest peak in the histogram is taken and any peak above 80% of it is also considered to calculate the orientation.

Finally key-point descriptor step, where in this step - after identifying the location of all interest key-points and assigned orientation for all of them in the previous step - Key-point descriptors are generated to represent the image data around the key-point (Lowe, 2004). Where A 16x16 neighbors around the key-point is taken. It is segmented into 16 sub-blocks with size 4x4; each one of them has eight bins (one for each 45 degrees). Therefore a total of 128 bin values (4x4x8) are available as shown in figure 4.



(A) (B)

Figure 4: Part (a) shows 16x16 windows around key-point. Part (b) Shows the 128 dimensional vectors (8*4*4) (Panchal, et al., 2013).

2.2.2 K-Means Clustering Algorithm

Clustering is the process of dividing objects into groups called clusters, and the objects inside the same cluster are more similar to each other than other objects from other clusters (Minaei-Bidgoli, et al., 2014; Parvin, et al., 2011). K-means clustering aims to group the n objects based on attributes/features into K number of groups.

The □-means clustering algorithm uses iterative refinement to create a final result. The algorithm consists of two main inputs factors; they are the number of clusters □ and the all data set. The data set is a collection of features for each data entry. The algorithm initially choices the □ centers, which can either, be randomly generated or randomly selected from the data set. Figure 5 shows the flowchart of K-means clustering.

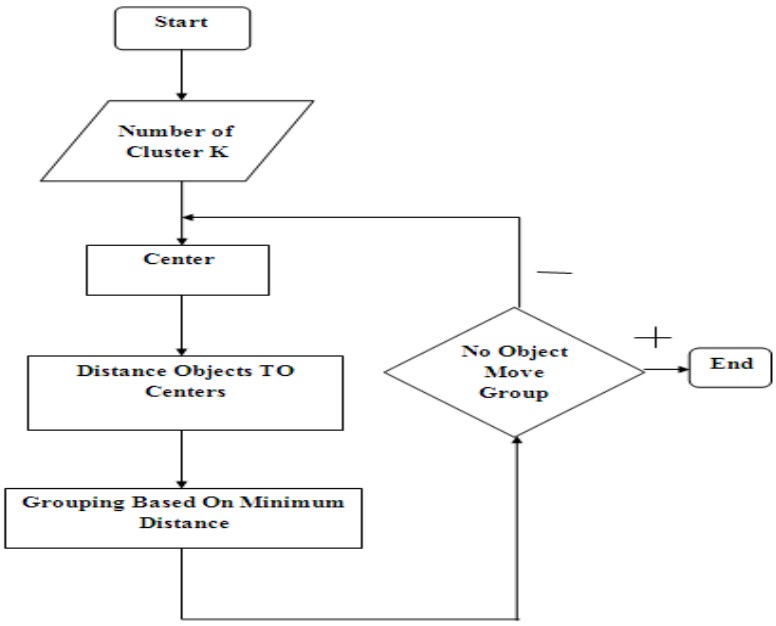


Figure 5: K-means clustering flow chart

The algorithm then iterates between two steps: Data assignment step and Center update step.

In data assignment step: Each cluster has one center. Where each data entry is compared to each center in all clusters, based on the squared Euclidean distance. Then the data entry moves to cluster that has the nearest center to the data entry.

In center update step the centers are recomputed. This is done by taking the mean of all data points assigned to that center's cluster.

The algorithm iterates between the above steps until a stopping criteria is met (i.e., no data points change clusters, the sum of the distances is minimized, or some maximum number of iterations is reached) (Jamnejad, et al., 2014).

2-2-3 The K-Nearest Neighbors (*k*-NN) Matcher

The *K*-nearest neighbors (*k*-NN) algorithm is a method to classify objects depended on the closest training examples in the feature space. It is considered one of the simplest matching and classifying algorithms. Even with its simplicity, it can achieve highly results. *k*-NN is a kind of instance-based totally learning or lazy to learn where the function only is approximated domestically and the entire computation is deferred till classification. It stores all available cases and classifies the new cases depends on the similarity measure (Mirkes, 2011).

k-NN matching classifies data into a training set and a testing set. For every row of the testing set, the K nearest training set objects (in Euclidean distance) are found, and the classification is assigned by way of majority vote with ties broken at random. If there are ties for the K-th nearest vector, all candidates are included within the vote (Bazmara and Jafari, 2013).

The training examples are considered as vectors in the multidimensional feature space, each one with a class label. The training step of the algorithm contains of storing the feature vectors only and class labels for the training samples.

In the classification step, k is considered as a user-defined constant, and an unlabelled vector (test point) is classified by way of assigning the label which is maximum frequent among the k training samples nearest to that test point. Figure 6 shows the process of K-nearest neighbors (*k*-NN) algorithm.

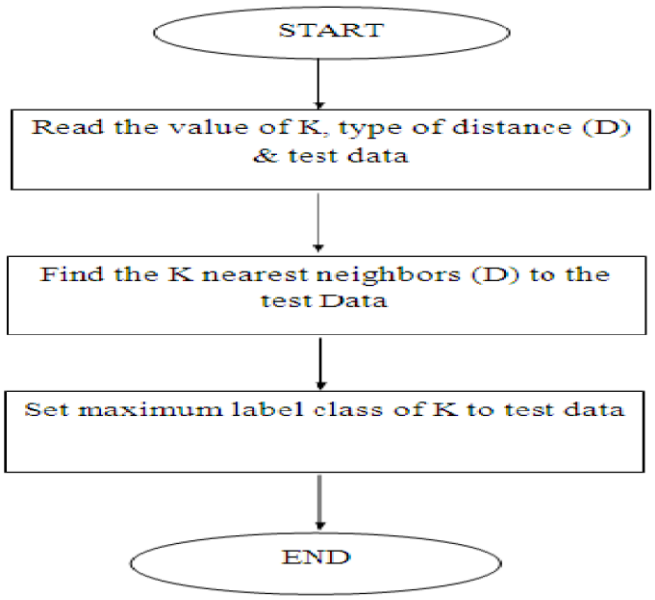


Figure 6: The process of K nearest neighbor matcher (Bazmara and Jafari, 2013).

Chapter Three: Research Methodology

The main goal of this research is to develop a full writer identification system for Arabic handwritten text to avoid the challenges those affect the Arabic writer identification systems. In this thesis, we proposed an Arabic offline text-independent writer identification system based on SIFT algorithm and k-means clustering algorithm using the *k*-NN matcher.

The research methodology is developed in five main stages to maintain a high recognition ratio and a good performance, including (Theoretical study, Arabic handwritten dataset creation, identifying the issues that needed to be considered in designing Arabic writer identification system, the development system frame work and comparison and analysis). Figure 7 shows the methodology followed in this research.

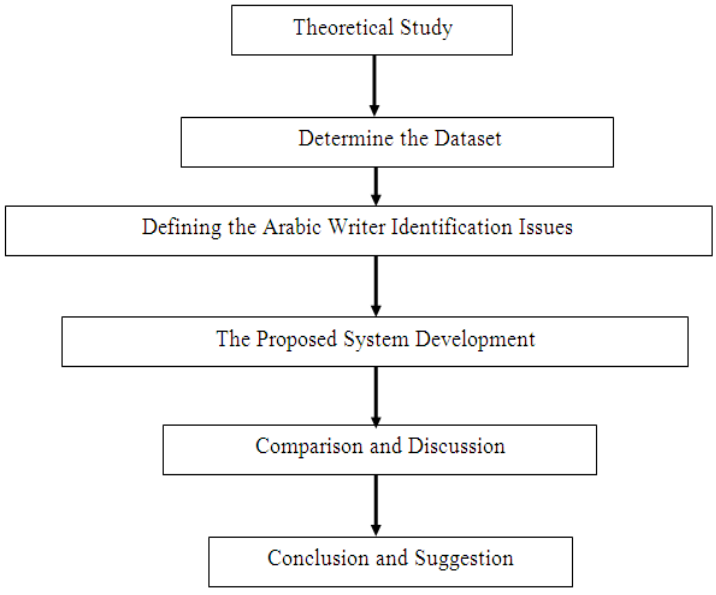


Figure 7: Research methodology of our proposed system.

3.1 Theoretical Studies

Theoretical studies that precede the development which include researching for the latest background of the research problems and identifying the objectives and scopes have been done. The theoretical studies are done by referring to literatures in published papers and journals.

3.2 Determine the Dataset

In our proposed system we used 569 samples for 10 writers from Arabic handwriting IFN/ENIT dataset. Most of these samples will be entered to the system in the training stage and the rest of them will be used to test the system in the identification stage.

3.3 Defining the Arabic Writer Identification Issues

While defining the Arabic Writer Identification Issues the challenges that must be taken into consideration when designing a full Arabic offline text-independent writer identification system are completely studied.

3.4 The Proposed System

In this thesis we proposed an Arabic offline text-independent writer identification system based on Scale Invariant Feature Transform (SIFT) algorithm and k-means clustering algorithm using the *k*-NN matcher. The system consists of two stages: training and identification stages. In the training stage, the SIFT descriptors (SD's) are extracted from the input handwriting sample,

then the k-means clustering algorithm are applied on these SD's to produce a centers for each writer which had been stored in the codebook. In the identification stage, the SD's are extracted from the test input handwriting sample and matched with the ones in the codebook for identification using *k*-NN matcher. Figure 8 shows the proposed system framework.

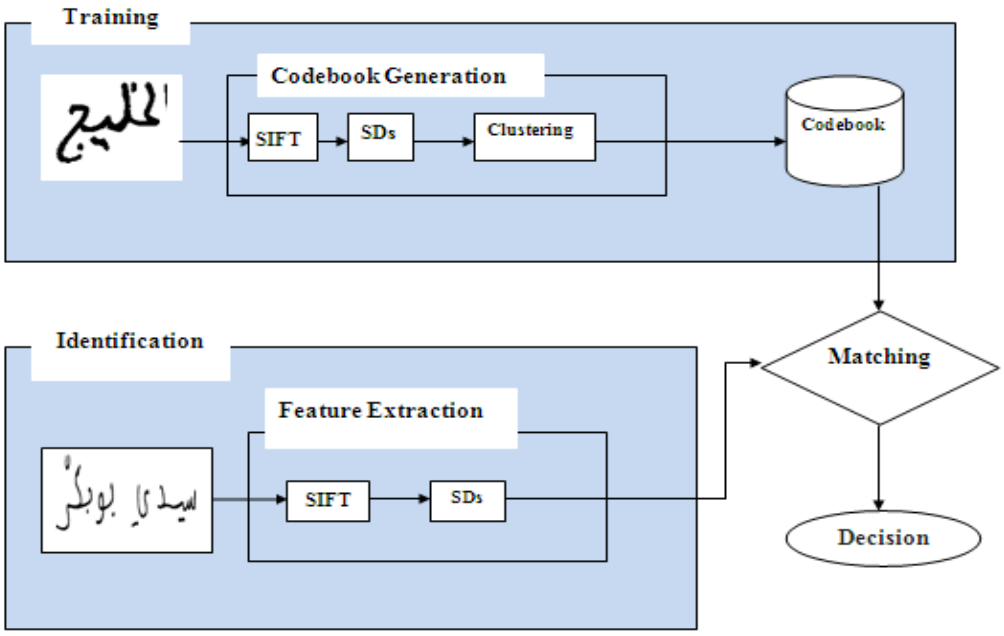


Figure 8: The proposed system framework.

3.5 Comparison and Discussion

For validation, the proposed system is compared with the one proposed by (Chawki and Labiba, 2010) based on the Arabic writing identification challenges. Both of two systems were evaluated by using the IFN/ENIT database.

From these five stages above the conclusion of the proposed system and the suggestion (future work) are provided at the end of this research.

Chapter Four: System Implementation

This chapter presents the implementation of our proposed system that detects the features for input handwriting samples using SIFT algorithm, k-means clustering and the *k*-NN as a matcher. Arabic handwriting IFN/ENIT dataset (Pechwitz, et al., 2002) was used in the system.

The framework of the system consists of several steps, which are determining the dataset step, data pre-processing step, features extraction step, codebook generation step and data matching step. The implementation of each step is explained in the following sections.

4.1 The Dataset

In this research we used the Arabic handwriting IFN/ENIT dataset (Pechwitz, et al., 2002). It considers one of the popular available Arabic handwriting dataset. This dataset was generated for training and testing (validating) the recognition systems for Arabic handwritten words and was used for the ICDAR 2005 Arabic OCR competition (Märgner, et al., 2005) .

IFN/ENIT dataset contains many images of Arabic handwriting sample of Tunisian towns/villages names. It was collected from 411 writers each one of them has nearly 50 handwritten samples filled in nearly 5 forms, in totally it contains nearly 26000 binary word images (Chawki and Labiba, 2010).

IFN/ENIT dataset images have a resolution of 300 dpi and they are in monochrome BMP format. The file size should be less than 18 KB and the image dimensions are also restricted and have an upper and lower bound, both for height and width. The height ranges are 50 and 161 pixels at the same time as the width ranges are 90 to 976 pixels. As an additional restriction, the town name at maximum includes three words and the word may have any number of sub-words, symbols or alphabets. Writers have been asked to fill in paperwork without a restrictions and without writing traces or boxes (Pechwitz, et al., 2002).

In this research we used 569 samples for 10 writers from IFN/ENIT dataset; each writer has from 50 to 60 different samples. Some of these samples are written more than one time for the same writer and some of the same samples are written from different writers as shown in figure 9. Most of these samples were entered to the system in the training stage and the rest of them were used to test the system in the identification stage. Figure 9 shows five handwritten samples for three different writers (a, b, c).

Figure 9: Examples of handwritten samples for different writers (a, b, c).

4.2 Data Pre-Processing

In this step, all the input handwritten samples are converted to gray-scaled images before the system extracted their features.

4.3 Features Extraction

The proposed system consists of two stages: training and identification stages. Feature extraction process is performed in both stages using SIFT algorithm.

In the training stage, the SIFT descriptors (SD's) are extracted from all input handwriting samples (one by one) for each writer. Writer SD's are stored in an array to be used in codebook generation step.

In the identification stage (testing stage), the SD's are also extracted from the test input handwriting sample to be used for identifying in matching stage.

4.4 Codebook Generation

Since SIFT algorithm is used to detect a number of key points and extract their descriptors, a large amount of key-points from different handwriting samples may be introduced, and it is not easy to keep all of these SD's for writer identification. K-means clustering algorithm is applied on the SD's which are extracted from the training stage in the previous step into N categories to make the number of the features limited and fixed and represent each category with its center, is called a code. All of the N codes form a SD codebook with size N. and N is determined as 300. Figure 10 shows the process of codebook generation

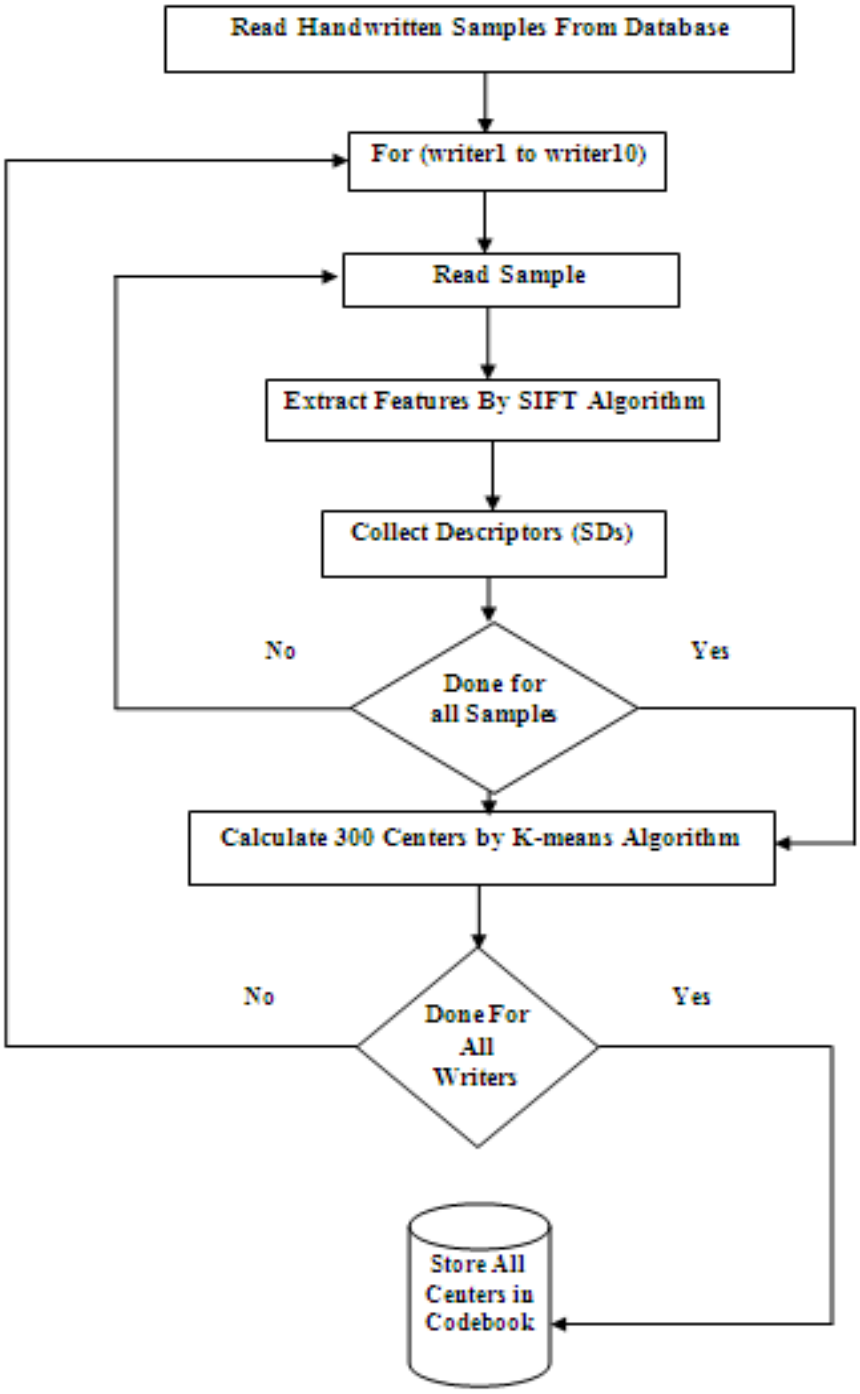


Figure 10: The process of codebook generation.

Figure 10 shows the process of codebook generation where the first input handwritten text for the first writer is entered to the system, and then the system will extract the features (SD's) from it using SIFT algorithm. This is will be done for all handwritten texts for the first writer to collect all his features, after that K-means clustering is applied for the features extracted by SIFT for the first writer to calculate his centers. The system will do this procedure for all writers to generate the codebook.

4.5 Feature Matching

In this step, the SD's extracted from the input handwriting sample in the identification stage are compared with 3000 centers (300 centers for each writer) which were stored in the codebook to identify the writer. This matching process is done by the *k*-NN matcher. Figure 11 shows the feature matching process.

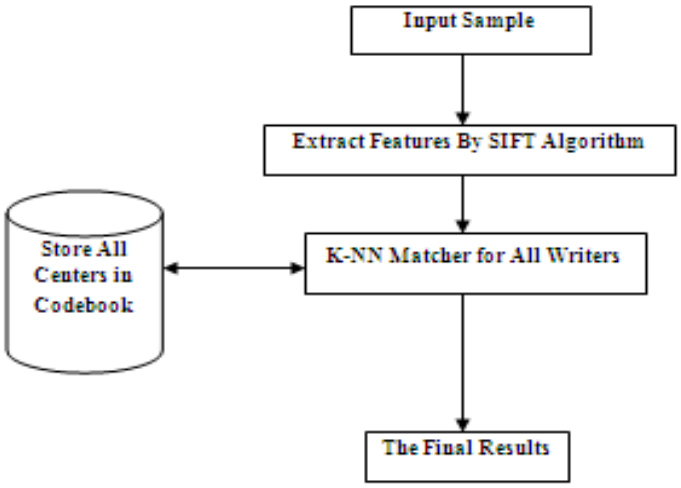


Figure 11: The process of feature matching.

Figure 11 shows the process of feature matching where this process is started after the training stage finished and codebook generation process completed. In feature matching process the handwritten text is entered to the system. Then the system will extract the features from it by SIFT algorithm and matched them with the ones are available in the codebook using the *k*-NN matcher to identify the writer based on the similarity between the features. And finally see the performance of the system and its accuracy.

4.6 System Implementation

Our proposed system was written in python language using open CV 3.0 libraries and was run using jetBrains PyCharm community edition 2017.1.1 program. The used computer is Lenovo laptop with windows 10 pro edition as an operating system; it has intel(R) core(TM) i3 CPU at 2.53 GHz and 4 GB installed memory.

A simple windows application is created to represent user framework. Figure 12 shows form application window:

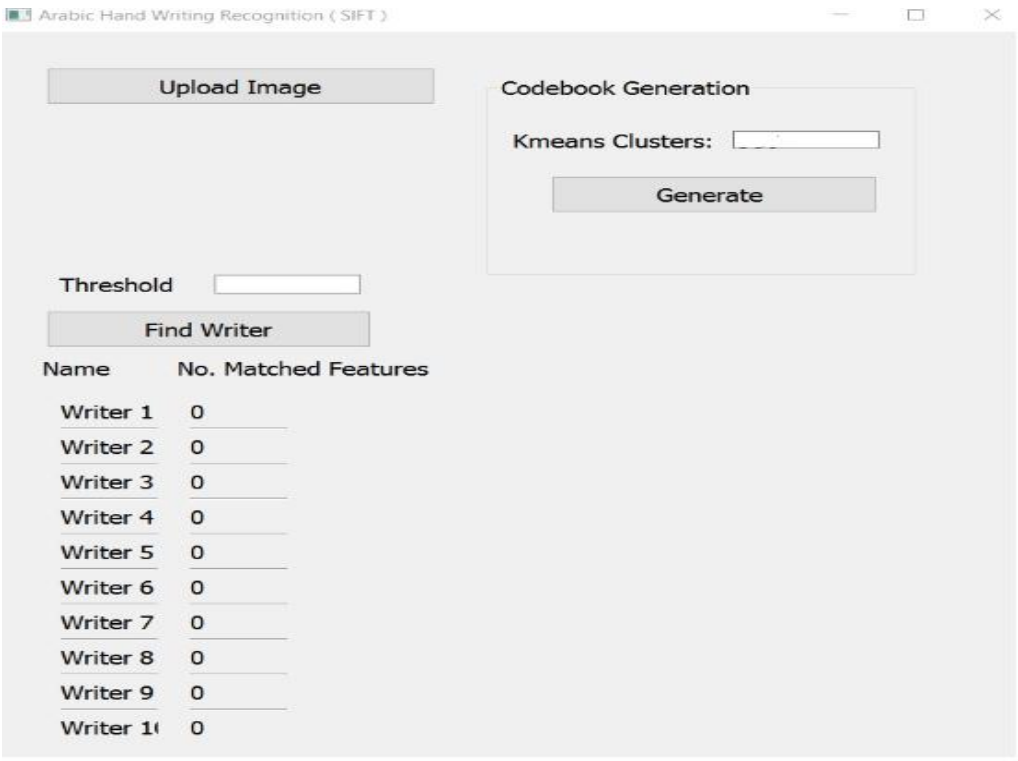


Figure 12: Our developed windows application Form.

As shown in figure 12, windows application form contains one picture box (where the input image is uploaded) and it contains also two text boxes; one for k-means clusters to determine the number of centers for SIFT descriptors clusters and the other one to determine the threshold value.

It also contains three buttons; generate codebook (calculate the values of centers for each writer), upload image (allow the user to select the new input image) and find writer (to display the number of matched features for all writers)

When the user clicks the generate button after he assigned the number of centers in the k-means clusters text box, the system will calculate the values of centers for each writer and saved them in the codebook as shown in figure 13:

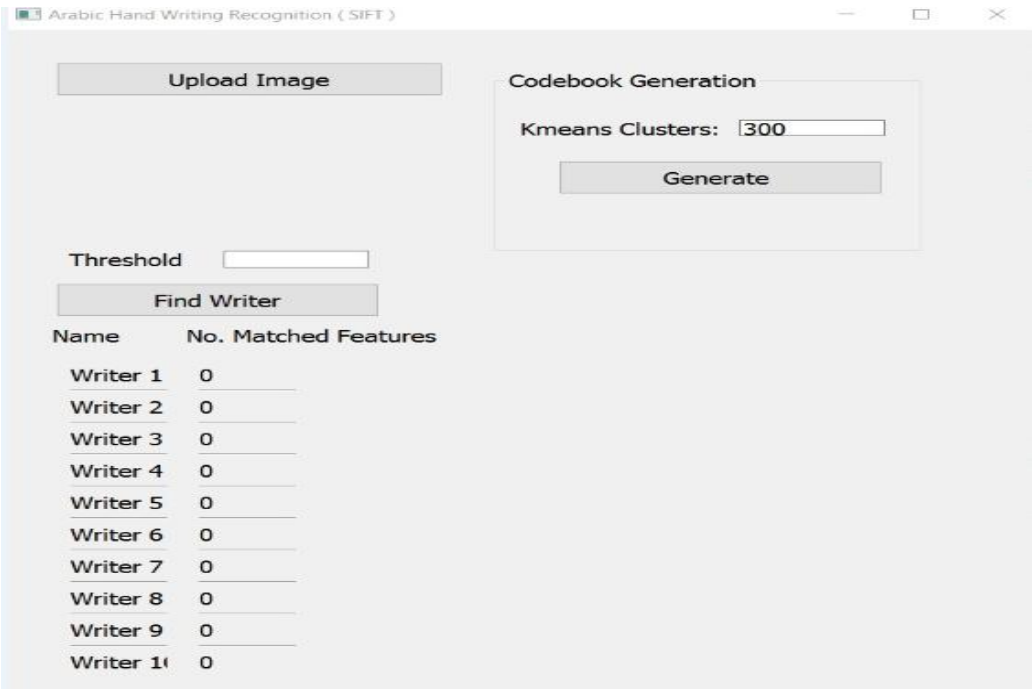


Figure 13: The process of assigning the number of centers.

When the system calculated the centers, it extracted the descriptors from each sample for all writers. Figure 14 shows the process of calculating the descriptors from the sample.

```
(679, 128)
# kps: 107, descriptors: (107, 128)
ai46_006.bmp
writer9
(786, 128)
# kps: 167, descriptors: (167, 128)
ai46_007.bmp
writer9
(953, 128)
# kps: 91, descriptors: (91, 128)
ai46_008.bmp
writer9
(1044, 128)
# kps: 123, descriptors: (123, 128)
ai46_009.bmp
writer9
(1167, 128)
# kps: 85, descriptors: (85, 128)
ai46_010.bmp
writer9
(1252, 128)
# kps: 79, descriptors: (79, 128)
ai46_011.bmp
writer9
(1331, 128)
# kps: 66, descriptors: (66, 128)
ai46_012.bmp
writer9
(1397, 128)
# kps: 108, descriptors: (108, 128)
```

Figure 14: The process of collecting the descriptors.

After the system finished calculating the number of centers process and saved them in the codebook successfully a box message will be shown to tell the user that the process of codebook generation is finished, as shown in figure 15:

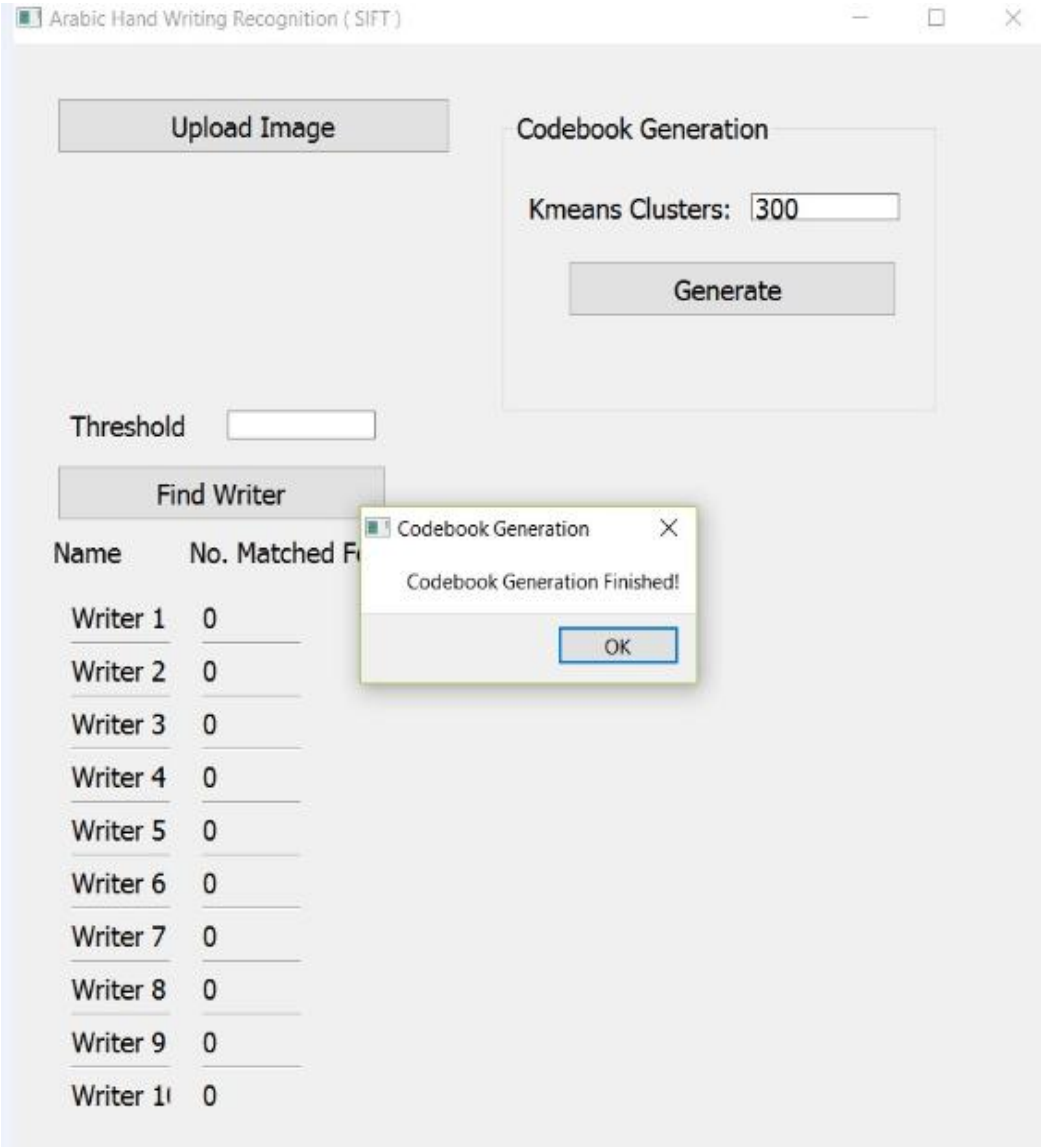


Figure 15: The process of codebook generation completed.

Then, the user can upload image for matching process by clicking on upload image button as shown in figure 16:

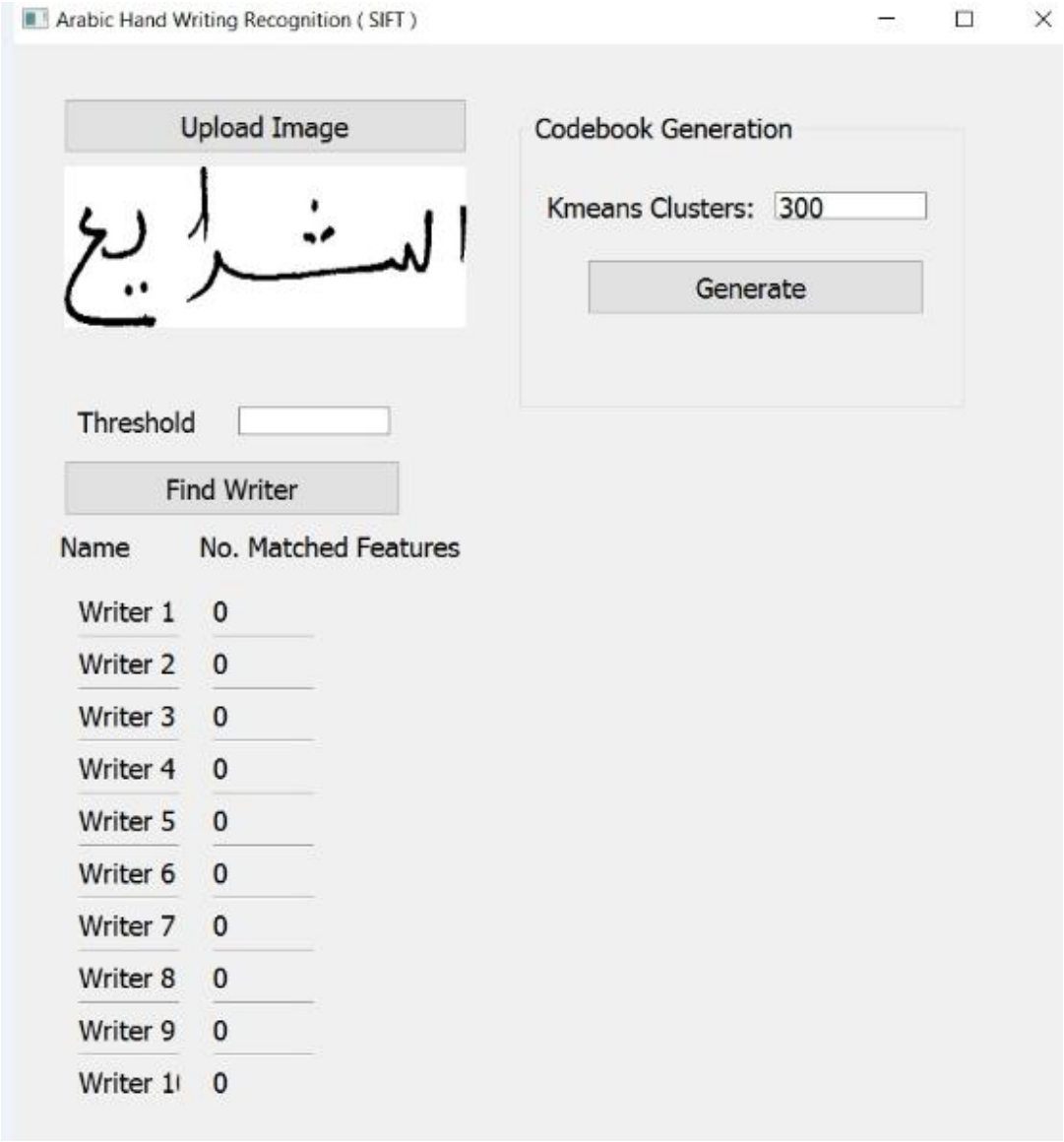


Figure 16: The process of uploading image.

Then the user should be determined the threshold value and this value is fixed and equal to 0.8 as (Lowe, 2004) as shown below in figure 17:

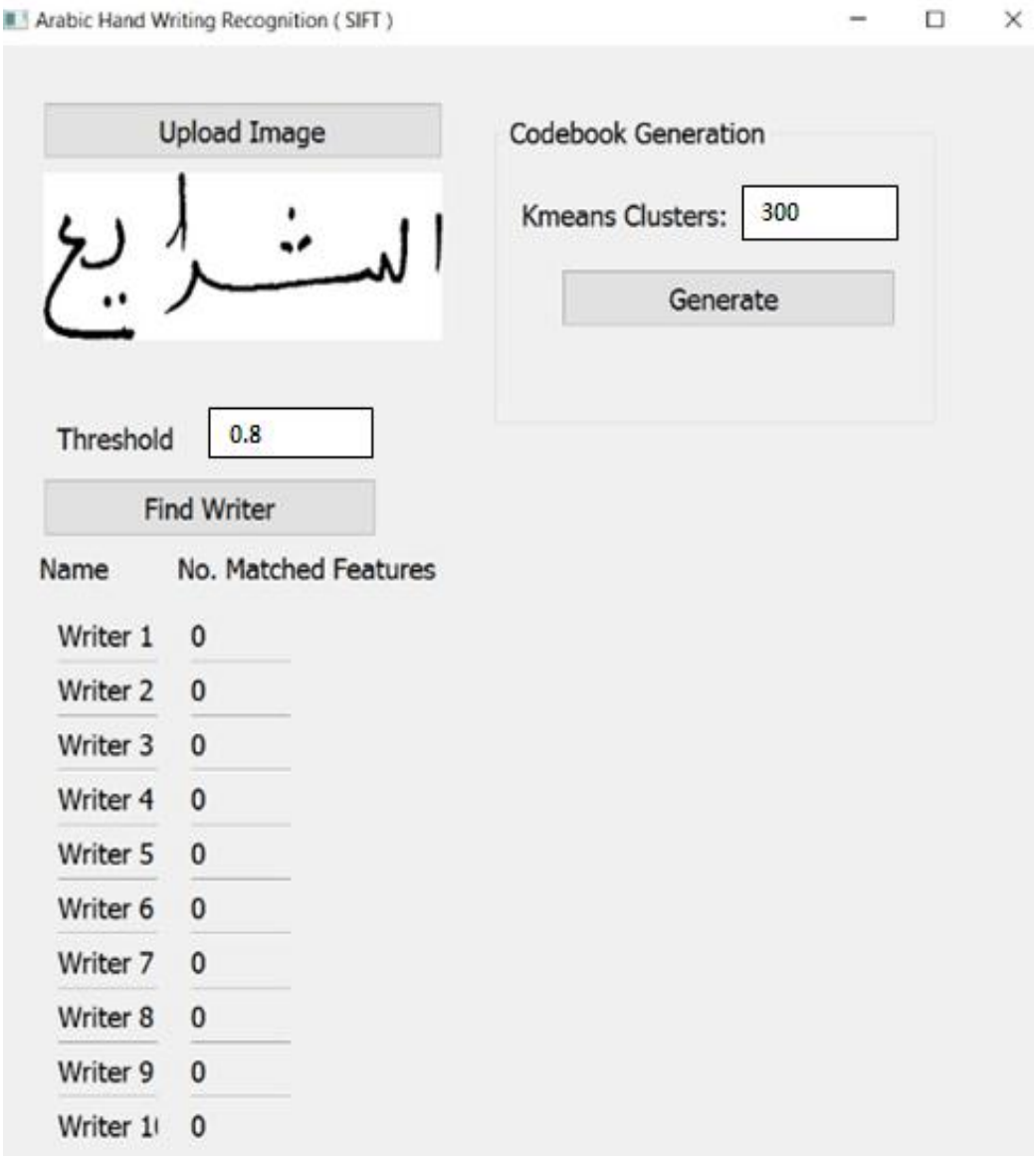


Figure 17: The process of determining the threshold value.

Once the user clicks find writer button, the system extracts the features of the new uploaded image and matches them with the ones that stored in codebook.

Then, the system retrieves the number of features for all writers that matched with the input image as shown in figure 18:

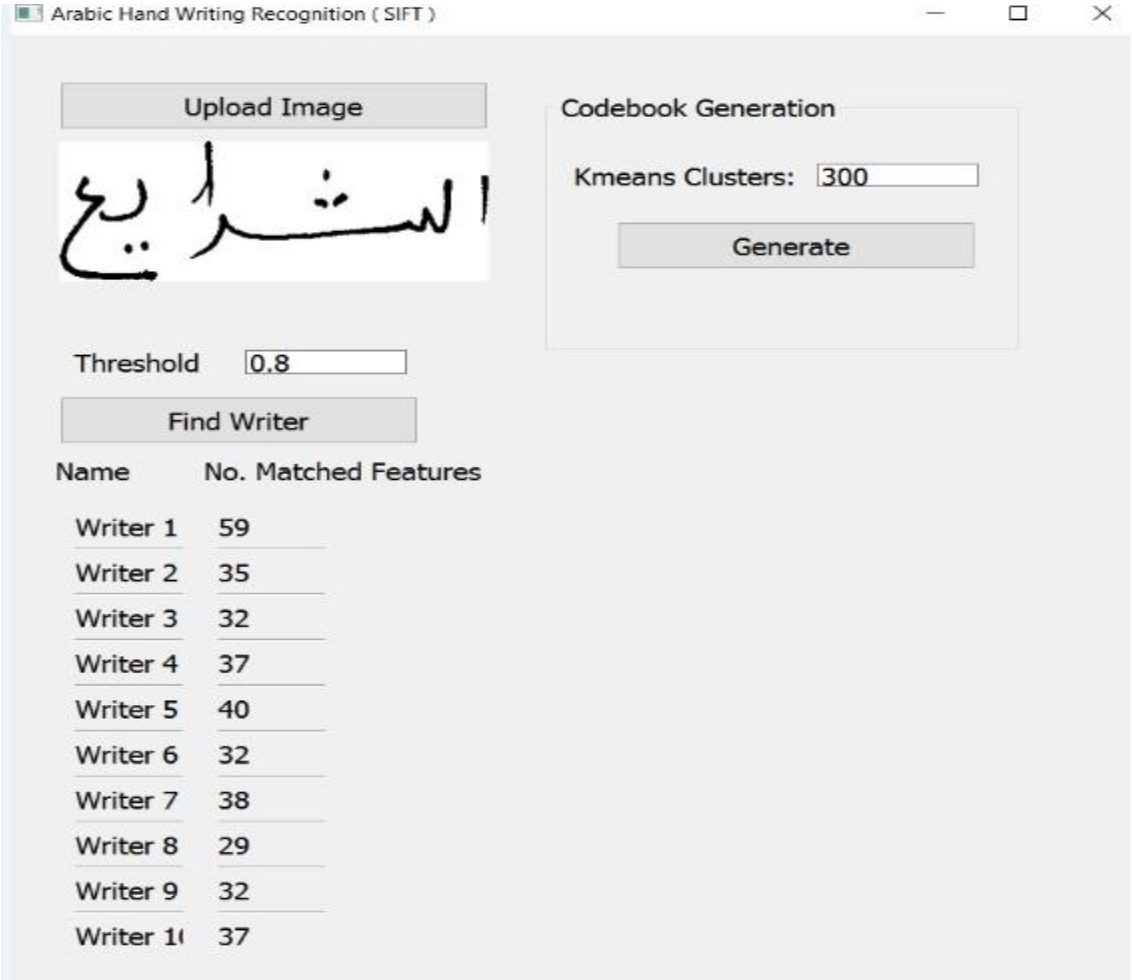


Figure 18: The process of recognizing the input image and the number of matched features for all writers.

As shown in figure 18 beside each writer there is the number of his matched features. And the one who has the highest value, he considered as Top 1. And the second one is considered as Top 2 and so on. In the above figure the input image is for writer 1, and the result of the number of his matched features is the highest. So we can say that writer 1 is Top 1 for that image.

Chapter Five: Results and Discussions

In this chapter, the proposed system is tested using 100 samples, 10 samples for each writer and the recognition ratio (RR) is measured for all writers. This parameter is indicator of system's robustness. Whenever, the performance of the system is increased if the RR is increased. Therefore, to design a robust system, we should take into consideration RR parameter.

Recognition ratio is defined as the ratio between the numbers of times when the system retrieves the correct samples and the number of all tests (the number of all used samples in system testing). The following equation calculate recognition ratio (RR):

$$RR\,(\%) = \frac{\text{No. of Retrieved Correct Samples}}{\text{No. of All Tested Samples in Dataset}} * 100\% \quad (6)$$

5.1 Results

The experimental study was implemented on the handwritten samples from IFN/ENIT dataset where the system was tested using 100 samples, each 10 samples for one writer. We tested our system more than one time and in each time we changed the number of centers of SIFT descriptor clusters. We used 150, 300 and 600 centers. And we considered the RR for Top 1, Top 2 and Top 3. Table 1 shows the RR of (Top1, Top2 and Top3) for each writer with 150 centers.

Table 1: RR for writer identification performance obtained with 150 centers.

RR (%) / Writer No	RR (%) For Top 1	RR (%) For Top 2	RR (%) For Top 3
Writer 1	90%	90%	100%
Writer 2	60%	60%	80%
Writer 3	70%	80%	90%
Writer 4	100%	100%	100%
Writer 5	50%	60%	80%
Writer 6	100%	100%	100%
Writer 7	80%	80%	90%
Writer 8	80%	90%	90%
Writer 9	60%	60%	70%
Writer 10	80%	100%	100%

Table 1 shows the RRs for SIFT algorithm with k-means clustering using *k*NN matcher for all writers. As we described before we used many Arabic handwritten texts from IFN/ENIT dataset to train the system (collecting features for all writers and store it in the codebook) and then we tested the system with 10 handwritten texts for each writer with different centers of SIFT descriptor clusters. In this case we tested the system with 150 centers of SIFT descriptor

clusters. And the results have shown that the RR is deferent from writer to writer.

Figure 19 shows the RR of the system when the number of centers of SIFT descriptor equal to 150 centers.

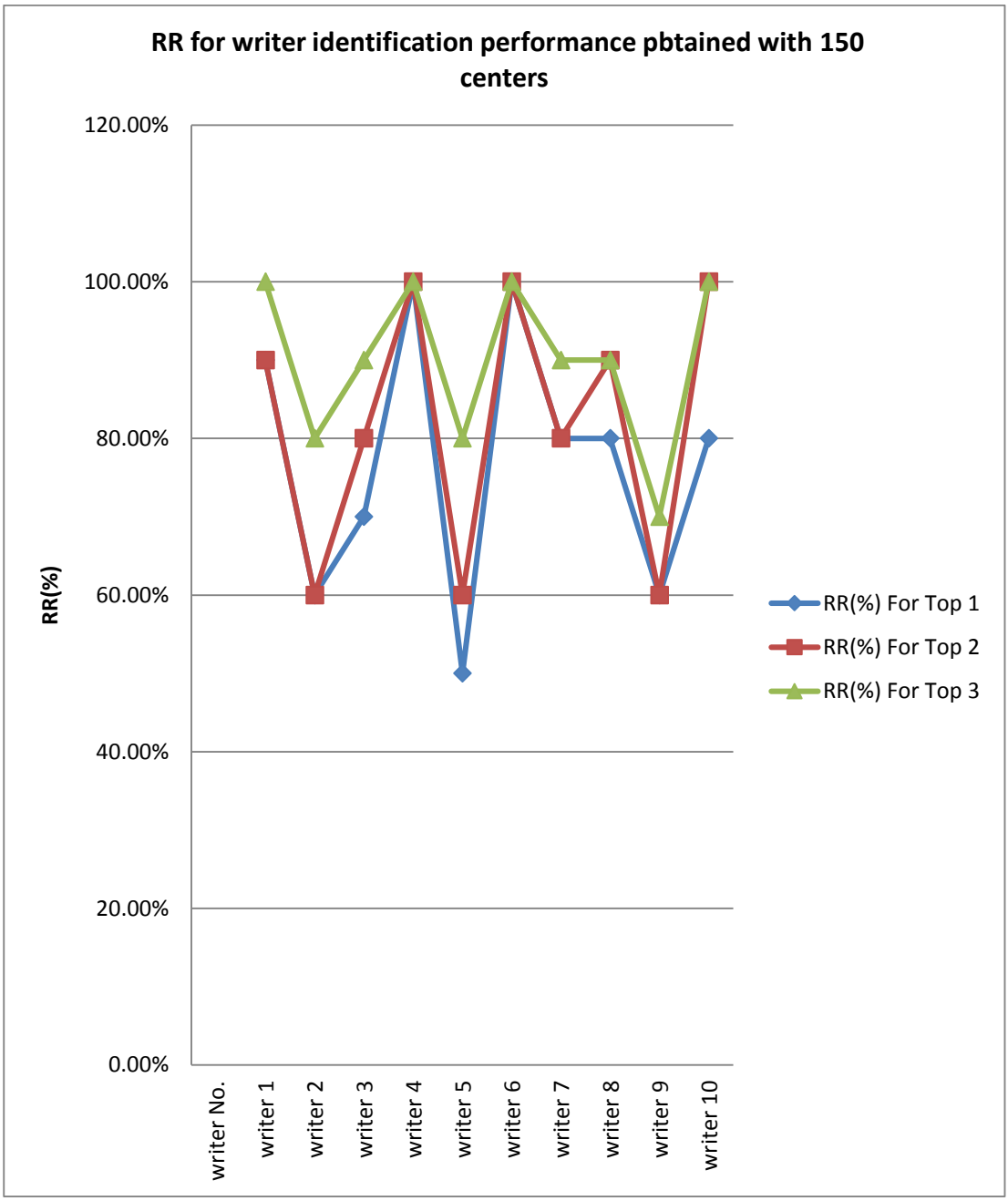


Figure 19: The RR with 150 centers.

Then we changed the number of centers to 300 centers to see how the performance of system will be with this changing as shown In Table 2.

Table 2: RR for writer identification performance obtained with 300 centers.

RR(%) / Writer No	RR(%) For Top 1	RR(%) For Top 2	RR(%) For Top 3
Writer 1	90%	90%	100%
Writer 2	70%	70%	90%
Writer 3	80%	80%	90%
Writer 4	100%	100%	100%
Writer 5	70%	80%	90%
Writer 6	100%	100%	100%
Writer 7	80%	90%	100%
Writer 8	80%	90%	100%
Writer 9	50%	60%	70%
Writer 10	90%	100%	100%

Table 2 shows RRs for the system for all writers when the number of centers of SIFT descriptor clusters are changed to 300 centers.

Figure 20 shows the RR of the system when the number of centers of SIFT descriptor equal to 300 centers.

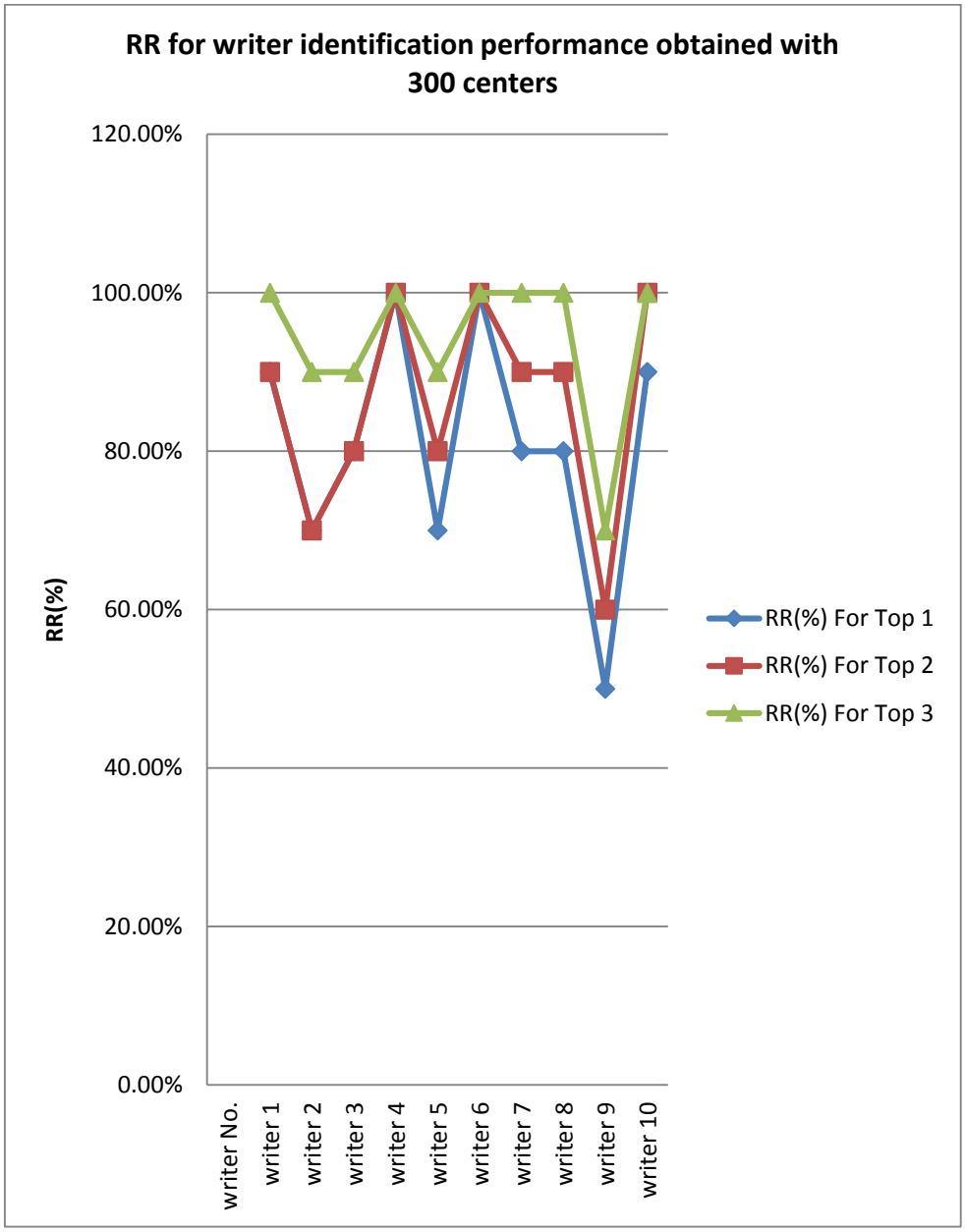


Figure 20: The RR with 300 centers.

Then we changed the number of centers to 600 centers to see how the performance of system will be with this changing as shown In Table 3.

Table 3: RR for writer identification performance obtained with 600 centers.

RR (%) / Writer No	RR (%) For Top 1	RR (%) For Top 2	RR (%) For Top 3
Writer 1	80%	90%	100%
Writer 2	60%	70%	80%
Writer 3	80%	80%	90%
Writer 4	100%	100%	100%
Writer 5	60%	70%	90%
Writer 6	100%	100%	100%
Writer 7	80%	90%	90%
Writer 8	70%	90%	90%
Writer 9	50%	60%	70%
Writer 10	80%	80%	90%

Table 3 shows RRs for the system for all writers when the number of centers of SIFT descriptor clusters are changed to 600 centers.

Figure 21 shows the RR of the system when the number of centers of SIFT descriptor equal to 600 centers.

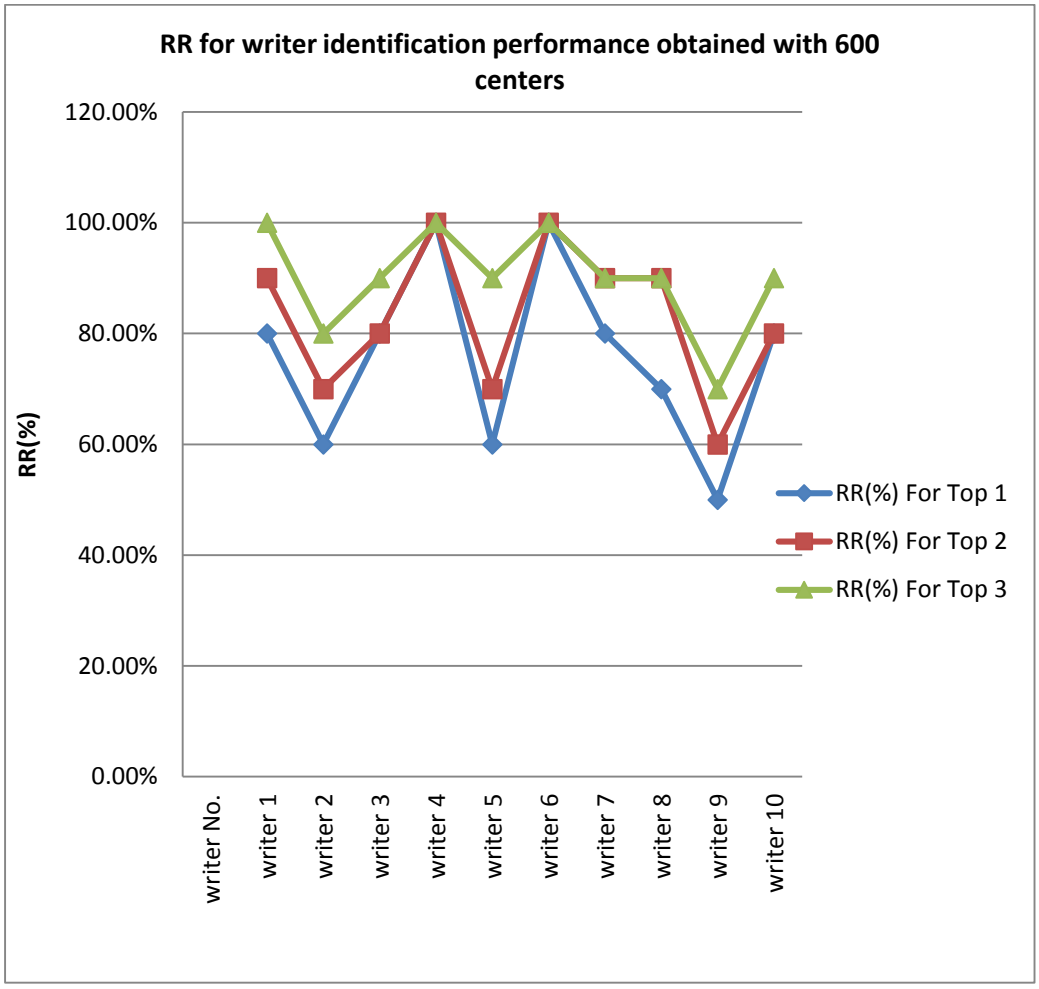


Figure 21: The RR with 600 centers.

The RRs percentage in the all above tables 1, 2 and 3 were calculated using the equation (6).

Then, we calculated the average of RR for all writers with all above cases to see the performance of system as shown in table 4.

Table 4: Comparison of the RR averages for writer identification performance for all writers with 150, 300 and 600 centers.

RR (%) / NO. of Centers	RR (%) For Top 1	RR (%) For Top 2	RR (%) For Top 3
150	77%	82%	90%
300	81%	86%	94%
600	76%	83%	90%

Table 4 shows that when using 300 centers of SIFT descriptor clusters we achieve the best performance for system in all of top 1, top 2, and top 3.

Figure 22 shows the Comparison of the RR averages for writer identification performance for all writers with 150, 300 and 600 centers.

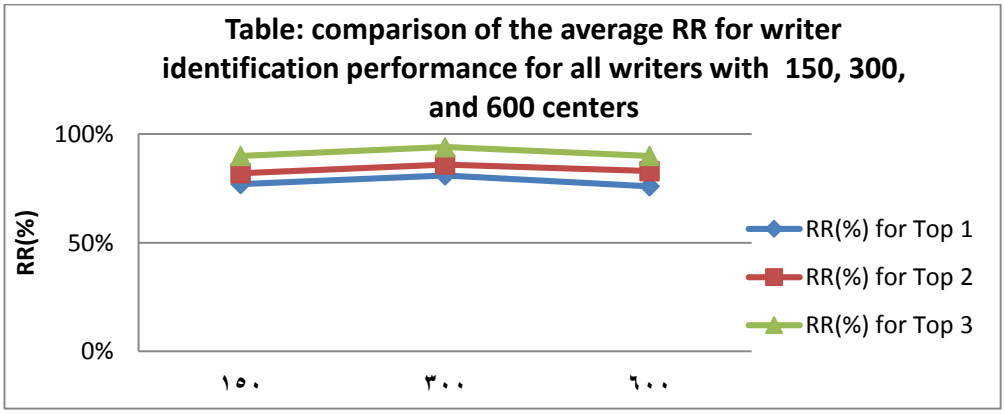


Figure 22: Comparisons of RR average with 150,300 and 600 centers

5.2 Discussions

The proposed system is tested using different inputs data; handwritten samples, centers and threshold value.

As shown in results section, the proposed system using SIFT algorithm with k-means clustering achieved an deferent RR ratio depends to the number of centers of SIFT descriptors.

As shown in table 1 for the first case (150 centers), system testing showed that, the system retrieved the correct samples 77 times in top 1, 82 times in top 2 and 90 times in top 3 from 100 samples, which makes the recognition ratio of the proposed system 77% in top 1, 82% in top 2 and 90% in top 3.

As shown in table 2 for the second case (300 centers), system testing showed that, the system retrieved the correct samples 81 times in top 1, 86 times in top 2 and 94 times in top 3 from 100 samples, which makes the recognition ratio of the proposed system 81% in top 1, 86% in top 2 and 94% in top 3.

As shown in table 3 for the third case (600 centers), system testing showed that, the system retrieved the correct samples 76 times in top 1, 83 times in top 2 and 90 times in top 3 from 100 samples, which makes the recognition ratio of the proposed system 76% in top 1, 83% in top 2 and 90% in top 3.

As a result, from the above deferent experiments, the performance of the system achieved best result when the system tested using 300 centers for the SIFT descriptors clusters. As shown on table 4.

It's obvious that some of writers have unique and special manner of writing and the most -if not all- of his handwritten texts are written in the same way and any one can note the similarity between them. This kind of writers usually accomplishes high RR percentage as writer 4 and writer 6 in all tables 1, 2 and 3, where the system could identify all their tested samples with RR = 100%. On the other side, the same writer may be written the same sample in a deferent manner. And this kind of writers causes a low RR percentage as writer 9 in all tables 1, 2 and 3.

In literature review section we showed other systems that used other feature extraction methods and deferent matchers to achieve a good result.

(Chawki and Labiba, 2010), they proposed Arabic off- line Text-Independent writer identification method. Where they implemented a texture classification approach particularly primarily based on a set of new proposed features extracted from Grey Level Run Length (GLRL) Matrices. They used IFN/ENIT Database in their experiment. The identification rates achieved from their experiment were 77.53% in Top 1, 84,46% in Top 2 and 88,62% in Top 3. And they made a Comparison with other two methods for them

((Combination Black GLRL, White GLRL and GLCM Features), GLCM Features).as shown in table 5.

A comparison between our proposed system with (Chawki and Labiba, 2010), had been done as shown in table 5.

Table 5: Comparison between RR of our proposed system and (Chawki and Labiba, 2010) and other two methods for them (Combination Black GLRL, White GLRL and GLCM Features, GLCM Features).

RR(%) / ALGORITHM	Top1	Top2	Top3
Combination Black and White GLRL (Chawki and Labiba, 2010)	77.23 %	84.46%	88.62%
Combination Black GLRL, White GLRL and GLCM Features	82.62%	89.08%	91.85%
GLCM Features	76.46%	85.23%	90.15%
Our Proposed Algorithm	81%	86%	94%

Table 5 shows a comparison between the performance of our system and (Chawki and Labiba, 2010) system. The RR of identify the writer as Top1 is 81% , as Top 2 is 86% and 94% as Top 3 for our system. While the RR of identify the writer as Top 1 is 77.23%, as Top 2 is 84.46% and 88,62% as Top 3 for their system.

Table 5 also shows A comparison between (Chawki and Labiba, 2010) and two other methods for them. Combination Black and White GLRL (Chawki and Labiba, 2010) method enhanced the recognition rate if we compare it with GLCM Features. But the combination between both methods (Black GLRL, White GLRL) and GLCM Features accomplished better results as shows in Table 5.

Figure 23 Shows comparison between RR of our proposed system and (Chawki and Labiba, 2010) and other two methods for them.

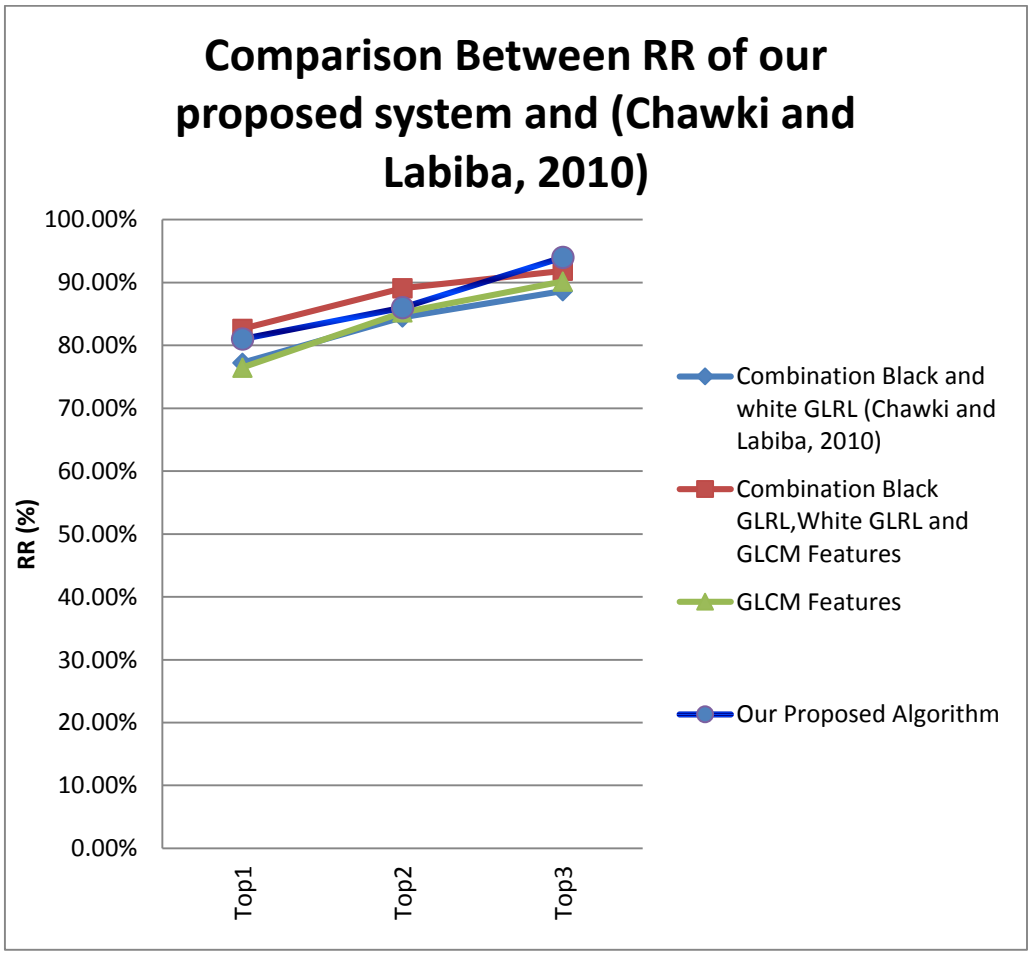


Figure 23: Comparison between RR of our proposed system and (Chawki and Labiba, 2010).

Chapter Six: Conclusions and Future Works

The improvement in image processing technology can be applied to resolve the personal identification problem, which is considered as one of the notably challenged problems in that technology. Many the traditional ways of personal identification (e.g. PIN, Keys, etc.) were failed and caused fake authentication, because they may be shared, lost or stolen. Therefore, the need to apply rapid, non-traditional and strong authentication way has increased, to maintain the security of our life. Biometric systems are among of the maximum reliable authentication systems.

6.1 Conclusions

Among different types of biometric systems, writer identification systems are considered one of the most popular behavioral biometric systems and recently they are a very active area of research due to the noticeably enhancement in information technology and its applicability in many fields such as security, financial activity, forensic, decision-making systems and to solve the expert problems in criminology.

In this thesis, we proposed an Arabic offline text-independent writer identification system based on Scale Invariant Feature Transform (SIFT) algorithm and k-means clustering algorithm. The system consists of two stages: training and identification stages. In the training stage, the SIFT descriptors (SDs) are extracted from the input handwritten samples, and then the k-means

clustering algorithm is applied on these SDs to produce a centers for each writer and store them in the codebook. In the identification stage, the SDs are extracted from the test input handwritten sample and matched with the ones in the codebook for identification by using *k*-nearest neighbors matcher (*k*-NN). We used 569 samples for 10 writers from Arabic handwriting IFN/ENIT dataset. Each writer has (50-60) different samples, some of these samples are written more than one time for the same writer and some of the same samples are written from different writers.

In this thesis, a comparison between three cases was applied by changing the centers of SIFT descriptors clusters; the first one with 150 centers, the second one with 300 centers and the third one with 600 centers. The results showed that the best case was when using 300 centers and the recognition ratio (RRs) of identifying the writer were 81% as Top 1, 86% as Top 2 and 94% as Top3.

In this thesis, a comparison between the performance of our system and (Chawki and Labiba, 2010) system was done. The RR of identify the writer as Top1 is 81% , as Top 2 is 86% and 94% as Top 3 for our system. While the RR of identify the writer as Top 1 is 77.23%, as Top 2 is 84.46% and 88,62% as Top 3 for their system.

6.2 Future Works

As a future work for our proposed system, we suggest to use another kind of local descriptors on feature extraction stage. Also the combination between local features and global features (slope, slope direction, density of thinned image, width to height ratio and skeweness… etc) may be enhanced the performance of the system and produce more reliable classification accuracy. And we suggest also to do some tests with others classifiers such as SVM or Bayesian classifiers.

References

[1] Ahmed, A., Sulong, G. (2014). Arabic writer identification: a review of literature, **J. Theo. Appl. Info. Tech**, 2014, 69(3), 474-484.

[2] Al-Dmour, A., Zitar, R. A. (2007). Arabic Writer Identification based on Hybrid Spectral-Statistical Measures, **J. Experimental and Theoretical Artificial Intelligence, 2007,** volume (19), 307—332.

[3] Al-Maadeed, S. (2012). Text-dependent writer identification for arabic handwriting, **Journal of Electrical and Computer Engineering**, 2012, 13.

[4] Al-Maadeed, S., Ferjani, F., Elloumi, S., & Jaoua, A. (2016). A novel approach for handedness detection from off-line handwriting using fuzzy conceptual reduction. **EURASIP Journal on Image and Video Processing**, 2016(1), 1.

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

[6] Baumberg, A. (2000). Reliable feature matching across widely separated views, **In Computer Vision and Pattern Recognition,** Proceedings. 2000, IEEE Conference on (Vol. 1, pp. 774-781). IEEE.

[7] Bazmara, M., & Jafari, S. (2013). K Nearest Neighbor Algorithm for Finding Soccer Talent. **Journal of Basic and Applied Scientific Research**, *3*(4), 981-986.

[8] Benjelil, M., Kanoun, S., Mullot, R., & Alimi, A. M. (2009, July). Arabic and latin script identification in printed and handwritten types based on steerable pyramid features, **In Document Analysis and Recognition,** 2009, ICDAR'09. 10th International Conference on (pp. 591-595). IEEE.

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

[10] Bezdek, J. C. (2013). Pattern recognition with fuzzy objective function algorithms, 2013, **Springer Science & Business Media.**

[11] Bouletreau, V., Vincent, N., Sabourin, R., & Emptoz, H. (1998), Handwriting and signature: one or two personality identifiers?, **In Pattern Recognition,** 1998**. Proceedings.** Fourteenth International Conference on (Vol. 2, pp. 1758-1760). IEEE.

[12] Bulacu, M., & Schomaker, L. (2007). Text-independent writer identification and verification using textural and allographic features, **IEEE transactions on pattern analysis and machine intelligence**, 2007, 29(4).

[13] Bulacu, M., Schomaker, L., & Brink, A. (2007). Text-independent writer identification and verification on offline Arabic handwriting. **In Document Analysis and Recognition,** ICDAR 2007. Ninth International Conference on (Vol. 2, pp. 769-773). IEEE.

[14] Bulacu, M., Schomaker, L., & Vuurpijl, L. (2003). Writer identification using edge-based directional features, **In: Seventh International Conference on Document Analysis and Recognition (ICDAR),** 2003.

[15] Carneiro, G., & Jepson, A. D. (2003, June). Multi-scale phase-based local features. **In Computer Vision and Pattern Recognition**, **Proceedings.** 2003 IEEE Computer Society Conference on (Vol. 1, pp. I-I). IEEE.

[16] Chawki, D., & Labiba, S. M. (2010). A texture based approach for Arabic Writer Identification and Verification. **In Machine and Web Intelligence (ICMWI),** 2010, International Conference on (pp. 115-120). IEEE.

[17] Chergui, L., & Kef, M. (2015). SIFT descriptors for Arabic handwriting recognition, 2015, **International Journal of Computational Vision and Robotics**, 5(4), 441-461.

[18] Cortes, C., & Vapnik, V. (1995). Support-vector networks**. Machine learning,** 1995, 20(3), 273-297.

[19] Djeddi, C., Siddiqi, I., Souici-Meslati, L., & Ennaji, A. (2013). Codebook for Writer Characterization: A Vocabulary of Patterns or a Mere Representation Space?. **In Document Analysis and Recognition (ICDAR),** 2013 12th International Conference on (pp. 423-427). IEEE.

[20] Dunn, J. C. (1973). A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. (1973): 32-57.

[21] Florack, L. M., ter Haar Romeny, B. M., Koenderink, J. J., & Viergever, M. A. (1994). General intensity transformations and differential invariants**. Journal of Mathematical Imaging and Vision,** 1994, 4(2), 171-187.

[22] Gazzah, S., & Amara, N. B. (2007). Arabic handwriting texture analysis for writer identification using the dwt-lifting scheme. **In Document Analysis and Recognition,** 2007, ICDAR 2007. Ninth International Conference on (Vol. 2, pp. 1133-1137). IEEE.

[23] Halder, Chayan, Sk Md Obaidullah, and Kaushik Roy, (2016), Offline Writer Identification and Verification—A State-of-the-Art**. Information Systems Design and Intelligent Applications.** Springer India, 2016. 153-163.

[24] He, Z., Tang, Y. Y., & You, X. (2005). A contourlet-based method for writer identification. **In Systems, Man and Cybernetics**, 2005 IEEE International Conference on (Vol. 1, pp. 364-368). IEEE.

[25] He, Z., You, X., & Tang, Y. Y. (2007). Writer identification of Chinese handwriting documents using hidden Markov tree model, 2007. **Pattern Recognition,** 41(4), 1295-1307.

[26] He, Z., You, X., & Tang, Y. Y. (2008). Writer identification using global wavelet-based features, 2008. **Neurocomputing,** 71(10), 1832-1841.

[27] Helli, B., & Moghaddam, M. E. (2010). A text-independent Persian writer identification based on feature relation graph (FRG), 2010. **Pattern Recognition**, 43(6), 2199-2209.

[28] Jain, A., Bolle, R., & Pankanti, S. (Eds.). (2006). Biometrics: personal identification in networked society, 2006. **Springer Science & Business Media** (Vol. 479).

[29] Jamnejad, M. I., Heidarzadegan, A., & Meshki, M. (2014). Text Recognition with k-means Clustering, 2014. **Research in Computing Science**, 84, 29-40.

[30] Kanade, T., Jain, A. K., & Ratha, N. K. (2005). Audio-and video-based biometric person authentication. **In 5th International Conference, AVBPA 2005, Hilton Rye Town, NY, USA, July 20-22, 2005, Proceedings.**

[31] Louloudis, G., Stamatopoulos, N., & Gatos, B. (2011). ICDAR 2011 writer identification contest. **In Document Analysis and Recognition (ICDAR),** 2011 International Conference on (pp. 1475-1479). IEEE.

[32] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In Computer vision, 1999. **The proceedings of the seventh IEEE international conference on (Vol. 2, pp. 1150-1157). IEEE.**

[33] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints, 2004 . **International journal of computer vision,** 60(2), 91-110.

[34] Lutf, M., You, X., & Li, H. (2010). Offline Arabic handwriting identification using language diacritics. **In Pattern Recognition (ICPR),** 2010 20th International Conference on (pp. 1912-1915). IEEE.

[35] MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. **In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability** (Vol. 1, No. 14, pp. 281-297).

[36] Maliki, M. J. R. (2015). Biometrics Writer Recognition for Arabic language: Analysis and Classification techniques using Subwords Features **(Doctoral dissertation, University of Buckingham),** 2015.

[37] Märgner, V., Pechwitz, M., El Abed, H. (2005). ICDAR 2005 Arabic handwriting recognition competition, **In Proc. of 8th ICDAR**, pp 70-74.

[38] Mikolajczyk, K., & Schmid, C. (2005). A performance evaluation of local descriptors. **IEEE transactions on pattern analysis and machine intelligence**, 27(10), 1615-1630.

[39] Minaei-Bidgoli, B., Parvin, H., Alinejad-Rokny, H., Alizadeh, H., & Punch, W. F. (2014). Effects of resampling method and adaptation on clustering ensemble efficacy, **Artificial Intelligence Review**, 2014, 1-22.

[40] Mindru, F., Tuytelaars, T., Van Gool, L., & Moons, T. (2004). Moment invariants for recognition under changing viewpoint and illumination. **Computer Vision and Image Understanding,** 2004, 94(1), 3-27.

[41] Mirkes, E. (2011). KNN and Potential Energy (Applet). **University of Leicester**.

[42] Newell, A. J., & Griffin, L. D. (2014). Writer identification using oriented basic image features and the delta encoding, **Pattern Recognition**, 2014, 47(6), 2255-2265.

[43] Palhang, M., & Sowmya, A. (1999, June). Feature extraction: Issues, new features, and symbolic representation, **In International Conference on Advances in Visual Information Systems (pp. 418-427).** 1999, Springer Berlin Heidelberg.

[44] Panchal, P. M., Panchal, S. R., & Shah, S. K. (2013). A comparison of SIFT and SURF, **International Journal of Innovative Research in Computer and Communication Engineering,** 2013, 1(2), 323-327.

[45] Parvin, H., Minaei-Bidgoli, B., & Alizadeh, H. (2011). A new clustering algorithm with the convergence proof, 2011**. Knowledge-Based and Intelligent Information and Engineering Systems,** 21-31.

[46] Pavelec, D., Justino, E., Batista, L. V., & Oliveira, L. S. (2008). Author identification using writer-dependent and writer-independent strategies**. In Proceedings of the 2008 ACM symposium on Applied computing (pp. 414-418). ACM**.

[47] Pechwitz, M., Maddouri, S. S., Märgner, V., Ellouze, N., & Amiri, H. (2002, October). IFN/ENIT-database of handwritten Arabic words, 2002. **In Proc. of CIFED (Vol. 2, pp. 127-136).**

[48] Plamondon, R., & Lorette, G. (1989). Automatic signature verification and writer identification—the state of the art, 1989. **Pattern recognition**, 22(2), 107-131.

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

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

[51] Saranya, K., & Vijaya, M. S. (2013). An interactive tool for writer identification based on offline text dependent approach, 2013. **International Journal of Advanced Research in Artificial Intelligence**, 2(1).

[52] Schlapbach, A., Liwicki, M., & Bunke, H. (2008). A writer identification system for on-line whiteboard data, 2008. **Pattern recognition**, 41(7), 2381-2397.

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

[54] Shahabi, F., & Rahmati, M. (2006). Comparison of Gabor-based features for writer identification of Farsi/Arabic handwriting, 2006, **In Tenth International Workshop on Frontiers in Handwriting Recognition**.

[55] Siddiqi, I., & Vincent, N. (2008). Combining global and local features for writer identification, 2008, **In Proceedings of the 11. Int. Conference on Frontiers in Handwriting Recognition, Montreal.**

[56] Siddiqi, I., & Vincent, N. (2009, July). A set of chain code based features for writer recognition. 2009, **In Document Analysis and Recognition**, ICDAR'09. 10th International Conference on (pp. 981-985). IEEE.

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

[58] Sreerag, S., Thambi, S., Menon, V. (2015). Offline Text Document Authorization on the Basis SIFT and SURF, 2015, **IJSTE - International Journal of Science Technology & Engineering** | Volume 1 | Issue 10.

[59] Sreeraj and S. M. Idicula, (2011). A Survey on Writer Identification Schemes, **International Journal of Computer Applications**, vol. 26, no. 2, pp. 23–33, July 2011.

[60] Srihari, S. N., Cha, S. H., Arora, H., & Lee, S. (2002). Individuality of handwriting, 2002, **Journal of forensic science**, 47(4), 1-17.

[61] Thasneem .P and Febina .P (2015)**.** Offline Hand Writer Identification Based on Scale Invariant Feature Transform, 2015, **International Journal of Science and Research**, Volume 4, Issue 5.

[62] Ubul, K., Adler, A., & Yasin, M. (2012). Multi-Stage Based Feature Extraction Methods for Uyghur Handwriting Based Writer Identification, 2012, **INTECH Open Access Publisher.**

[63] Wu, X., Tang, Y., & Bu, W. (2014). Offline text-independent writer identification based on scale invariant feature transform, 2014, **IEEE Transactions on Information Forensics and Security,** 9(3), 526-536.

[64] Zhu, Y., Tan, T., & Wang, Y. (2000). Biometric personal identification based on handwriting. In Pattern Recognition, 2000, **Proceedings. 15th International Conference on (Vol. 2, pp. 797-800). IEEE**.

Arabic summary
الملخص

مع التطورات المستمرة في مجال الاتصالات و المجالات الصناعية الأخرى يزداد الطلب على استخدام أنظمة التوثيق ذات المصداقية العالية. تستخدم هذه الأنظمة في العديد من أعمالنا اليومية بما في ذلك: الأعمال المصرفية نشر المعلومات و أنظمة التجار الإلكترونية و التجارة عبر الانترنت.

نظم تحديد هوية كاتب النص هي أحد أكثر أنظمة التوثيق شيوعا، بالرغم من التطور الهائل في الأنظمة إلا أنه لم يتم دراسة عملية التعرف على هوية الكاتب العربي كما هو الحال للكاتب اللاتيني أو الصيني حتى السنوات القليلة الماضية. تطوير أنظمة التعرف على هوية الكاتب العربي تواجه العديد من التحديات بما في ذلك خصائص الكتابة العربية، التشويش، وترقيق النص فضلا عن الملامح أو رسم الكتابة اليدوية؛ فإنه يتأثر بسهولة من الميل.

في هذه الأطروحة قدمنا مقترح لنظام التعرف على هوية كاتب النصوص المختلفه المدخله مسبقا باللغه العربيه باستخدام خوارزميات (Scale Invariant Feature Transform (SIFT)) and (k-means clustering) للتعامل مع التحديات التي تواجه اللغه العربيه.

النظام المقترح يتكون من مرحلتين: التدريب و التعرف . في مرحلة التدريب ، تقوم خوارزمية (SIFT) باستخراج ((SIFT descriptors (SD's من النصوص المدخله المكتوبه باليد، وبعد ذلك نطبق عليها خوارزمية (k-means clustering) لانشاء مجموعه من المراكز لكل كاتب و تخزينها في سجل. في مرحلة التعرف ، تقوم

خوارزمية (SIFT) باستخراج ال (SDs) من النص المدخل لغاية الفحص المكتوب باليد ومن ثم مقارنتها مع (SDs) الموجوده في السجل للتعرف على كاتب هذا النص باستخدام (*k*-nearest neighbors matcher (*k*-NN)). تم استخدام قاعدة البيانات (IFN/ENIT) في النظام المقترح.

في هذه الأطروحه تمت المقارنه بين ثلاث حالات من خلال تغيير عدد مراكز(SDs clusters) التي تم انتاجها من خلال تطبيق خوارزمية (k-means clustering)والثلاث حالات هي (150,300,600) مركز.

بينت النتائج ان افضل نتيجه كانت عند تطبيق 300 مركز حيث ان نسبة التعرف كانت 81% كأعلى قيمه، 86% كأعلى ثاني قيمه و 94% كأعلى ثالث قيمة. وكانت هذه النتائج أفضل عند مقارنتها مع (Chawki and Labiba, 2010)