Tech Science Press

A New Segmentation Framework for Arabic Handwritten Text Using Machine Learning Techniques

Saleem Ibraheem Saleem[1,*], Adnan Mohsin Abdulazeez[1] and Zeynep Orman[2]

[1]Department of Information Technology, Technical Informatics College of Akre, Duhok Polytechnic University, Duhok, 42004, Kurdistan Region, Iraq
[2]Department of Computer Engineering, Faculty of Engineering, Istanbul University-Cerrahpasa, Istanbul, 34320, Turkey
*Corresponding Author: Saleem Ibraheem Saleem. Email: saleem.saleem@dpu.edu.krd

Abstract: The writer identification (WI) of handwritten Arabic text is now of great concern to intelligence agencies following the recent attacks perpetrated by known Middle East terrorist organizations. It is also a useful instrument for the digitalization and attribution of old text to other authors of historic studies, including old national and religious archives. In this study, we proposed a new affective segmentation model by modifying an artificial neural network model and making it suitable for the binarization stage based on blocks. This modified method is combined with a new effective rotation model to achieve an accurate segmentation through the analysis of the histogram of binary images. Also, propose a new framework for correct text rotation that will help us to establish a segmentation method that can facilitate the extraction of text from its background. Image projections and the radon transform are used and improved using machine learning based on a co-occurrence matrix to produce binary images. The training stage involves taking a number of images for model training. These images are selected randomly with different angles to generate four classes (0–90, 90–180, 180–270, and 270–360). The proposed segmentation approach achieves a high accuracy of 98.18%. The study ultimately provides two major contributions that are ranked from top to bottom according to the degree of importance. The proposed method can be further developed as a new application and used in the recognition of handwritten Arabic text from small documents regardless of logical combinations and sentence construction.

Keywords: Writer identification; handwritten Arabic; biometric systems; artificial neural network; segmentation; skew detection model

1 Introduction

The increased usage of innovative and sophisticated hacking and falsification technologies in recent years has contributed to the rising demand for alternate strategies for identifying individual identities [1]. Biometric systems remain promising innovations that differ from conventional methods as they can be used in digital systems to identify and/or check an individual's

identity uniquely and efficiently without having to bring or recognize something. Specifically, these systems, such as the automation of access control over physical and virtual places (e.g., borne controls, ATMs, safety and surveillance systems, financial transactions, and computer/networking security), are commonly used in various governmental and civil-sensitive applications [2]. Personal identification based on biometric features offers numerous benefits over conventional knowledge-based approaches (e.g., passwords or personal identification numbers) because it eliminates the difficulties emerging from switching, losing, forgetting, or duplicating driver's licenses, passports, identification cards, or even simple keys. Therefore, this approach is easy and user-friendly for clients who no longer need to recall or bring anything with them in the process of verifying their identities [3]. Biometric systems are subject to higher safety standards in comparison with conventional methods. A biometric system identifies and classifies biometric functions that match extracted personal features against a reference collection stored in a system database with a sequence of discriminating features from captured objects. Furthermore, their ultimate judgment depends on the implications of the comparison. Biometric systems benefit users who do not want to have to recall information or take something with them to verify their identities. In this case, losing, exchanging, or missing individual knowledge is prevented. Given the low contrast and high complexity of biometrical models, physiological biometrics, particularly iris, fingerprint, and DNA, are relatively accurate. Such biometrics are commonly used and implemented in recognition schemes. By contrast, conduct biometrics are seldom used because of the strong comparison between different biometric models extracted from actions. Nonetheless, the recognition of an individual based on handwriting samples appears to be a valuable biometric tool primarily because of its forensic applicability. Writer identity is a type of behavioral biometry because personal writing is an early learning and perfected activity [4].

The writer identification (WI) of Handwritten Arabic text is the foundation of our WI program. The Arabic language has a wide spectrum of uses, with 347 million individuals collectively speaking it [5]. However, more than 1.2 billion Muslims in the world use Arabic every day to recite the Quran and to pray. Several other languages use Arabic letters in their writing or the same letter types but with small modifications (population of about 700 million) [6]. Examples include Hausa, Kashmiri, Tatar, Ottoman Turkish, Uyghur, Urdu and Cashmiri, Malay, Moriscco, Pashto. Kazakhstan, Kurdish, Punjabian, Sindhi, and Tataran. Arabic is also one of the five most commonly spoken languages in the world (including Chinese, French, Spanish, and Hindi/Urdu) [7]. The current research focuses on author recognition (WI) on the basis of handwritten Arabic text. WI is a mental biometric that utilizes a handwritten document in any language. It is regarded as useful because personal handwriting is an early learning and redefined habit. Humans are also valuable biometric tools through the recognition of their signature patterns and the digitalization of theological and historical records, forensics, violent incidents, and terrorist attacks. Technically, the automated recognition of an individual's handwriting may be treated in the same manner as any biometric identification. It involves the retrieval of digital vector representation, the availability of adequate samples from multiple consumers of these vectors, and the presence of a certain measurement/distance between such vectors. Such distance may represent a correlation between the persons from which the samples are collected and correct classification. This research may profit from current WI work utilizing handwriting in other languages, such as English. However, in our attempt to tackle this issue, we should understand the form and characteristics of Arabic language. The success of existing schemes is influenced by several factors, such as the variations in the national and educational histories of Arabic text authors [8].

WI has been revived in recent years for a range of purposes. Today, forensic proof and verification techniques are being widely used by courts around the world. Furthermore, the rise in crimes and terror activities has prompted authorities to carry out proactive counter-activities on the basis of WI. WI's handwritten Arabic text is now of great concern to intelligence agencies following the recent attacks perpetrated by Middle East terrorist organizations. WI is also useful in the digitalization and attribution of old text to other authors of historic studies, including old national and religious archives [1–3]. WI from handwritten text structures is typically focused on digital characteristics, with letters/strokes representing information acquired from current research on the integration of individual writing habits/styles. According to [1], the idea of WI emerged from previous research that reported the better recognition of a word's attributes than those of a character or stroke. Given the complexity of Arabic handwriting, segmenting and separating letters and strokes from a script presents another challenge on top of WI schemes [8].

The main aim of the current work is to build a system that can identify writers on the basis of their handwriting. We attempt to investigate and highlight the limitations of existing methods. Subsequently, we build a powerful model that can help to identify writers by using their sentences, words, and subwords. The main contributions of this study can be summarized as follows:

- We propose a new affective segmentation model by modifying an existing ANN model and making it suitable for the binariztion stage based on blocks. This model is combined with a new effective rotation model to achieve an accurate segmentation through the analysis of the histogram of binary images.
- We propose a new framework for correct text rotation that will help us to establish a segmentation method that can facilitate the extraction of text from its background. Image projections and the radon transform are used and improved using machine learning based on a co-occurrence matrix to produce binary images.
- The main handwritten texts display a confident orientation method from the horizontal text baseline called "text skews." Skew detection is an important stage that can facilitate segmentation and the extraction of good features or signs for WI. To address related limitations, we propose a framework that is based on binary images produced from the initial segmentation stage.

The rest of the paper is organized as follows. Section 2 presents the related work and highlights the research problem. Section 3 introduces the proposed system that can identify writers on the basis of their handwriting. Section 4 discusses the experimental results of the pre-processing and segmentation stages for Arabic handwriting recognition. Finally, Section 5 details the conclusion and future works.

2 Related Work

Preprocessing and segmentation are critical factors in obtaining significant and vital function vectors from accessible handwritten text samples to be used for the identification of text writers. In this context, preprocessing is aimed toward the preparations for the requested segmentation of Arabic handwritten text based on input signal scanning [9]. The preprocessing phase involves measures that are unique to (1) the scanning method, which may add errors or noise; (2) the general design of handwriting, including the mismatch of written lines; and (3) the arrangement of Arabic text, including the overlap between text lines, phrases, subwords, letters, and idiomatic phrases. The last two categories of preprocessing activities include the form of the pen used, the pressure added to the pen, and the font size [10]. Segmentation is the method of separating a scanned image's written text into elements for recognition. The separation covers the rather

critical trends in the text that must be derived from, contrasted, and adapted to new prototype vectors. This section discusses in depth the problems of preprocessing and segmentation and the possible algorithms for addressing them. To provide evidence in support of our views and to direct our research toward the second segmentation portion, we discuss herein the composition of the elements of Arabic documents [11].

Text preprocessing is a critical feature of all content classification methods. The key strategy is to delete unrealistic functions while maintaining features that support the classification method in later steps to forecast the correct form of text. Text preprocessing typically requires the removal of stop phrases, writing, numerals, and terms at low levels. Different experiments to determine the impact of text preprocessing on various facets of text classification have been performed. For example, Ahmad et al. [9], GonÇalves et al. [12], and Srividhya et al. [13] compared the preprocessing results of English, Turkish, and Portuguese texts. They reported that text planning increases grading precision and reduces the characteristic scale while improving the grading speed. Arabic text identification involves two forms of preprocessing, as well as the elimination of punctuation, prefixes, and low-frequency phrases and stop words. The first form includes specific measures involving the deletion of Latin letters, Arabic and Kashida idiomatic phrases, and the normalization of Hamza and Taa Marbutah. This step minimizes the function size and addresses the lack of data. This procedure is also commonly adopted in the classification of Arabic text and in other Arabic natural language processing applications. The characteristics resulting from this type of preprocessing are the same as those obtained from term phonology. Several experiments have shown that when such basic preprocessing measures are implemented, the accuracy of classification improves [14,15]. Meanwhile, the second form of preprocessing is required for the complex manipulation of details, stemming, and root extracting. The stems and roots that are removed are most often used as characteristics. Stemming and root extraction minimize the usable size and quantity of records while term phonology is converted into one stem or root. Instead of being implemented separately, the first and second preprocessing modes are usually combined and performed simultaneously. Many studies have tested the impact of this approach on the Arabic text categorization of stems and roots relative to term orthography.

The characterization of the usage of stems is deemed more precise than that of the usage of roots [16,17]. If the use of word orthography is contrasted with the use of roots and stems for the grading of Arabic language, then conflicting results are obtained. For example, Haralambous et al. [18] found that stems yield better results for identification than spelling and roots while Hmeidi et al. [19] reported that orthography provides better classification results than roots and stems. Wahbeh et al. [20] analyzed two datasets, recognized that they have just a particular dataset, and inferred that stemming with limited datasets and term spelling with large datasets are reliable. The possible interpretations of these contradictory findings contribute to the usage of multiple stemmers of specific and low precision and to the design of datasets for experiments.

According to [21] suggested a Latin letter or component of character feature extraction and proposed a testing algorithm. The authors collected related factors into diagrams and then separated them into potential strokes (characters or a portion of characters). Grapheme k-means clustering was used to describe a function context that is specific to all database documentation. Experiments were conducted on three different datasets containing 88 books and 39 historical accounts. These datasets were authored by 150 scholars. In a knowledge recovery system, a writer's identity was investigated, and writer verification was focused on shared information among the graphics distributed in the manuscripts. The tests revealed an accuracy of almost 96%. For feature extraction, the same technique for grapheme classification has been used [22]. For the study

of Latin text fragments (characters or pieces of characters), Schomaker used related methods but focused on Kohlon's self-organized feature map (SOM). The author introduced a writer recognition algorithm by categorizing documents into fragments [23]. After attempts to realize smoothing and binarizing, the information obtained was based on the part contours for such fragments. Then, the information was used to measure the Moore contour. The Moore district comprises eight pixels in a 2D square matrix next door with a center pixel. In comparison with Kohonen's self-organizing map, Kohlon's SOM shows the "connection contours" of the fragments of the learning set. SOM is a form of artificial neural network that is taught to identify the inputs of vectors by classifying the vectors into maps or cartoons in the input field. SOM uses a multifaceted system presentation method in relatively narrow dimensional spaces.

In another study by [24], an algorithm was evaluated on the basis of a separate database of Western language collected from texts by 150 authors. K-nearest neighbor algorithm was used to identify the top 1 writers with a precision of 72% and the top 10 authors with an accuracy of 93%. The same technique was adopted for uppercase Western text. They introduced an algorithm to classify Latin writers on the basis of the detection of handwriting character attributes. The specifications included parameters such as height, distance, numbers of end points, frequency of joints, roundness, duration of circle, field and center of gravity, angle, and frequency of loops. The device was checked 10 times by 15 authors in random sequences of 0 to 9. For identification, the hamming interval was used with a reliability rate of 95%. Another study by [24] suggested methods for the recognition and authentication of Arab and Latin writers on the basis of extraction structure and textural and allographic characteristics. The study proposed the segmentation of text into characters by using allographic characteristics as author style with a focus on the premise that words should not overlap. In the lowest contour, the segmentation was performed to a point. K-means was used for codebook clustering. This codebook was introduced as a teaching package of sample graphics. The codebook was also standardized to construct a histogram for any related term by using the Euclidian width. The analysis was performed by 350 authors, each of whom conducted five tests using an unbiased database. The most effective recognition rate was 88%. The result related to Arabic text were lower than those reported for Latin text.

In another work by [25], independent Persian characters were checked for the recognition of Persian authors. The pictures were analyzed and eventually segmented into multiple strokes. A collection of characteristics, including the stroke orientation, horizontal and vertical outlines, and stroke measuring ratio, was defined in every stroke. To classify researchers, the study used a variation of fuzzy rule-based quantization and FLVQ. The algorithm was evaluated on a standalone database by 128 readers, with the findings and accuracy levels in different test circumstances reaching 90%–95%. Also, another study by [26] proposed an algorithm instead of dealing with alphabetic characters to classify Latin authors by using handwritten digits to retrieve characteristics. The specifications comprised parameters such as height, distance, frequency of end points, frequency of joints, roundness, duration of circle, field and center of gravity, angle, and volume of loops. The device was checked 10 times by 15 authors on random strings between 0 and 9. For identification, the hamming interval was used with a precision rate of 95%.

Many researchers have conducted writer recognition by using other languages. Existing approaches rely on a single type of recognition (e.g., online *vs.* offline similarity) or on other text content and functions, as explained and mentioned in the introduction section (text dependent *vs.* nontext dependent). Other studies have utilized articles, chapters, sections, terms, or characters in writer recognition. Certain methods are more involved than others in terms of small sections, such as character or stroke pieces found in text written in Latin, Persian, and other languages. Such

methods have been employed in various languages. Researchers should clarify the observable traits of current handwritten text analytics with brief samples. Considering that most of the existing literature is focused on offline text, we structure our analysis according to the collection of text elements that are deemed to be biased and widely used in writer recognition. Our analysis is not strictly focused on Arabic because the growing curiosity in WI lies in Arabic handwritten script.

3 The Proposed Method

Fig. 1 shows the framework for of the proposed algorithm for segmenting text from a static image and assessing the affective features for writer recognition. To correctly locate the text in the given image, the proposed algorithm utilizes histogram equalization for noise elimination, highlights the text, and increases the contrast for dark regions. In this stage, we enhance the existing histogram equalization by using an adaptive threshold that can help to identify dark regions. We also use the histogram equalization based on the complexities of a region. In other words, we develop a new adaptive method for histogram control. In this method, the first step is segmentation, which involves binarizing an image in the next stages. Specifically, we propose a powerful trainable model that can help to identify a piece of text from the rest of a given image even if the text is written in a different color. This model helps to capture poor-quality text. Then, the skew of the text is corrected using the binary image as the input. In this stage, we rotate the image to the correct position. We then propose a new trainable technique to identify the angle of rotation for the text. If the text is in the right angle, then we move to the post-segmentation of the text; otherwise, we use a morphological operation and radon transform to rotate the text. In the previous stages, we binarize the image and rotate it in preparation for the post-segmentation stage. The post-segmentation stage includes investigating the histogram of the binary image for each vertical line and adjusting the threshold for the extraction of each line that includes text while ignoring others. At this stage, we have extracted each line from the image that includes text. Then, for each line and by using the same technique, we extract each subword. At this point, we analyze each line and use a threshold for extracting each subword. A new model based on new features is finally investigated and implemented for WI.

3.1 The Preprocessing Stage

3.1.1 Adaptive Histogram Equalization

Given the poor quality of some of the images used in our research, we use histogram equalization to enhance these images and highlight the pieces of text for the segmentation and identification stages. Fig. 2 shows the poor-quality images in our dataset.

As shown in Fig. 2, the images contain unclear text, noise, and unclear backgrounds. On the basis of our investigation and analysis of existing articles, we find that histogram equalization is one of the most effective techniques with minimal limitations. We enhance the existing adaptive histogram equalization and make it suitable for our problem. One of the most frequently utilized approaches to image contrast improvement is histogram equalization because of its simplicity and high efficiency. It is accomplished by normalizing intensity distribution using the method's cumulative distribution function. The resultant images will then have identical intensity distributions. Histogram equalization may be categorized into two types depending on the utilized transformation function: local or global. Global histogram equalization (GHE) is fast and simple, but its contrast enhancement power is relatively small. By contrast, local histogram equalization (LHE) improves total contrast efficiently. The histogram of the overall input image is utilized in GHE for computing the function histogram transformation. As result, the dynamic range of the

image histogram is expanded and flattened, and the total contrast is enhanced. As the computational complexity of GHE is relatively small, it is an appealing tool for several applications involving contrast enhancement. Useful histogram equalization enhancement applications involve texture synthesis, speech recognition, and medical image processing, and they are commonly used to modify histograms. Histogram-based image improvement techniques typically depend on the equalization of image histograms and the expansion of the range of dynamics corresponding to images.

Figure 1: General framework of proposed system

Figure 2: Poor-quality images. Image (A) shows poor edges and an overlap between the words and the lines, image (B) shows noise and indistinguishable objects

Histogram equalization has two main disadvantages that reduce its efficiency [27–33]. First, histogram equalization assigns one gray level to two dissimilar neighbor gray levels with dissimilar intensities. Second, if a gray level is included in most parts of an image, then histogram equalization allocates a gray level with greater intensity to that gray level, resulting in a phenomenon called "wash out." Assume that $X = X(i, j)$ represents a digital image while $X(i, j)$ represents the pixel gray level at (i, j) place. The number of pixels in the entire image is denoted as n, and the intensity of the image is digitized in L levels, that is, $\{X0, X1, X2, \ldots, XL - 1\}$. Thus, $\forall X(i, j) \in \{X0, X1, X2, \ldots, XL - 1\}$. Suppose that nk represents the overall number of pixels with gray level of Xk in the image; the Xk probability density is written as

$$p(X_k) = \frac{n_k}{n}, \quad k = 0, 1, \ldots, L-1 \tag{1}$$

The relationship between p(Xk) and Xk is expressed as the probability density function (PDF), and the graphical appearance of the PDF is called the histogram. Depending on the image's PDF, its cumulative distribution function is expressed as

$$c(X_k) = \sum_{j=0}^{L-1} P^{(X_j)} = \sum_{j=0}^{L-1} \frac{n_k}{n} \tag{2}$$

where $k = 0, 1, \ldots, L - 1$ and $c(XL - 1) = 1$. A transform function f(x) is described as follows depending on the cumulative density function:

$$f(x) = X_0 + (X_{L-1} - X_0)c(x) \tag{3}$$

Next, the output image of GHE, $Y = Y(i, j)$ may be defined as

$$Y = f(x) = \{f(X(i, j)) \forall X(i, j) \in X\} \tag{4}$$

Bihistogram equalization (BHE) is used to solve the challenges in brightness preservation. BHE divides the histogram of an input image into two portions depending on the input mean before separately equalizing them. Several BHE approaches have been suggested to solve the aforementioned challenges. Essentially, BHE involves dividing the input histogram into two portions. Those two portions are independently equalized. The key distinction among the techniques in this family is the criterion utilized to choose the separation threshold represented by XT. The techniques described above increases the image contrast but often with insufficient details or undesirable effects. The next section describes histogram equalization and BHE methods as implemented and then suggest modifications to remove such undesirable effects for images with mean intensity toward the lower side.

Clearly, $XT \in \{X0, X1, \ldots, XL - 1\}$. Depending on the threshold, the input image X may be separated into two sub-images XL and XU:

$$X = X_L \cup X_U$$

where $X_L = \{X(i, j) \mid X(i, j) \leq X_T, \ \forall X(i, j) \in X\}$

and $X_U = \{X(i, j) \mid X(i, j) > X_T, \ \forall X(i, j) \in X\}$ \hfill (5)

Then, the respective PDFs of the sub-images XL and XU are described as

$$p(X_k) = \frac{n_k}{n_L}, \quad k = 0, 1, \ldots, T \tag{6}$$

and

$$p(X_k) = \frac{n_k}{n_U}, \quad k = T + 1, \ldots, L - 1 \tag{7}$$

nk denotes the Xk numbers in XL and XU and nL and nU are the overall sample numbers in XL and XU, respectively. The respective cumulative density functions for XL and XU are therefore

$$C_L(X_k) = \sum_{j=0}^{k} PL^{(X_j)} \tag{8}$$

$$C_U(X_k) = \sum_{j=T+1}^{k} PU^{(X_j)} \tag{9}$$

Similar to the GHE case, in which a function of total density is utilized as a transform function, we describe the next transform functions by using the functions of total density. The result of the proposed adaptive histogram equalization is shown in Fig. 3.

3.1.2 Text Orientation (Skew) Problem

Text orientation can be made flexible by scanning written text. In this case, we face two problems. First, the writer has problematic handwriting and is not capable of controlling his hand to write on the same line. A skewed image is then produced, as shown in image A in Fig. 4. Second, when we scan the document to convert it into an image, we obtain a skewed image,

as shown in image A of Fig. 4. Skew detection is an important activity in the preprocessing of text images because it directly affects the efficiency and reliability of the segmentation and feature extraction phases. The major methods for correcting skew are smearing, projection, Hough transform, grouping, graph-based, and correlation [34]. In projection, the profile is measured by summing up the intensities discovered at every scan line from whole pixels. The corresponding profile is smoothed, and the created valleys are recognized. The space between text lines is specified by these valleys. In smearing, the black pixels sequential in the horizontal profile are marked. Whether the distance among the white space is over a particular edge, it is completed with black pixels. The bounding boxes of linked components in the marked image are proposed as lines of text. Grouping techniques establish lines of alignments through grouping units. These units can be blocks, linked components, or pixels. They are then linked together to extract the lines of alignments. Hough transform is also utilized in skew detection. The points in the Cartesian coordinate system are defined as a sinusoidal distribution in the following equation:
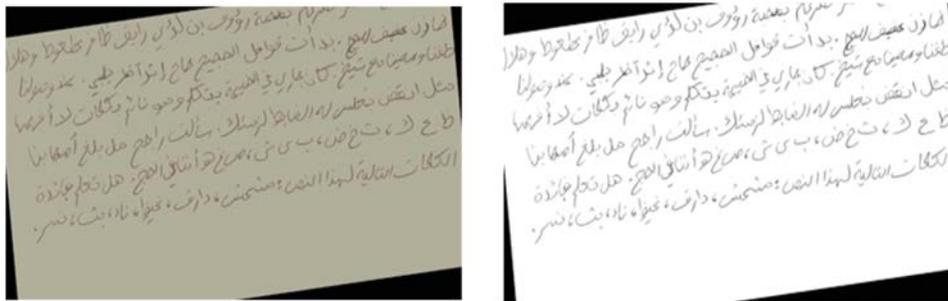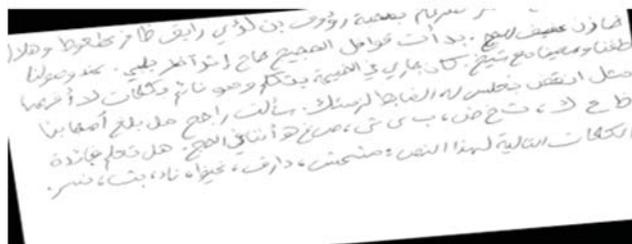
$$p = x cos\theta + y sin\theta \tag{10}$$

Figure 3: Affective result of proposed adaptive histogram equalization

Figure 4: Types of image skews: Image (A) text skew and image (B) whole image skew

The outcomes show that the proposed algorithm (utilizing our database) for estimating and correcting skewed text is effective and provides decent line separations depending on the projection. An additional problem is to keep the text features unchanged while addressing challenges or at least restoring them to their state after solving the challenges [35]. Overlapping lines make this problem increasingly difficult to fix. In this stage, we propose a new framework for rotating text correctly [36]. The framework should help us find a segmentation method that can facilitate the extraction of text from backgrounds. The proposed model includes a trainable model for binarizing images and rotating skewed images.

3.2 Initial Segmentation (Image Binarization)

The aim of the binarization is to isolate a piece of text from the rest of an image. We use a trainable model instead of the traditional threshold method. The proposed model can capture text with poor structures and different colors as it depends on structure edges and texture. This step includes two stages: training and testing stages. The training process begins with the selection of a definite number of images to be trained. In our work, we use 100 images. These images include different texts with different challenges. From every training image, many samples, such as a minor window of square areas of a specific size (e.g., 5×5), are obtained from text and the background. In this case, we take 100 windows from each image as two classes; 50 windows include text, and the remaining ones show the background, which includes white background, noise, or unwanted objects. This selection helps us to keep the true positive objects and ignore false positive objects, including noise, unwanted objects, and artifacts. For every window of any class, the texture features are utilized instead of pixel intensity. A selection of histogram of oriented gradient (HOG) features and moments features is derived and defined herein. The labeled feature vectors gathered from the samples are subsequently utilized to train ANN classifiers. As presented in Fig. 5, an ANN represents a network of interrelated computational nodes (units). Every node denotes two processing functions: transformation and summarization. The function of summarization is to estimate a weighted sum on the basis of the inputs, each of which contains of an input value and its related weight. Weight wji is the product of the node of input j in the former layer of the network and the node of output I of the present layer. A function of transformation of the node, called activation function, applies transformation, which is usually nonlinear, to the summarization function outcome.

The HOG and moments features are applied to each window, and the output is used as the input for the ANN with class labels. Upon completion of the training, every set testing image is utilized as an input for the binarization algorithm. The algorithm images a pixel of image through pixel. For every pixel, a minor square area of an identical window size with the pixel as the center is built, and the HOG and moment features of the area are extracted through the trained neural network and classified into two classes, namely, "text" and "nontext." As mentioned previously, we use HOG and moments features (Fig. 5) to describe objects in terms of structure and texture. The HOG represents a feature descriptor that facilitates object identification in digital images. The HOG is framed according to the gradient orientation quantification in localized image sections. The HOG feature extraction procedure includes taking a window called cell around the pixels. A mask [−1, 0, 1] is utilized to calculate the image gradients. In our version of this extraction technique (Fig. 6) for orientation binning, we utilize the gradient directly for every image position in the matching orientations. The orientation of the cells is set to 0°–180° with 9 bins. The contrast normalization of the local histogram is employed to improve the invariance to illumination, shadowing, etc.
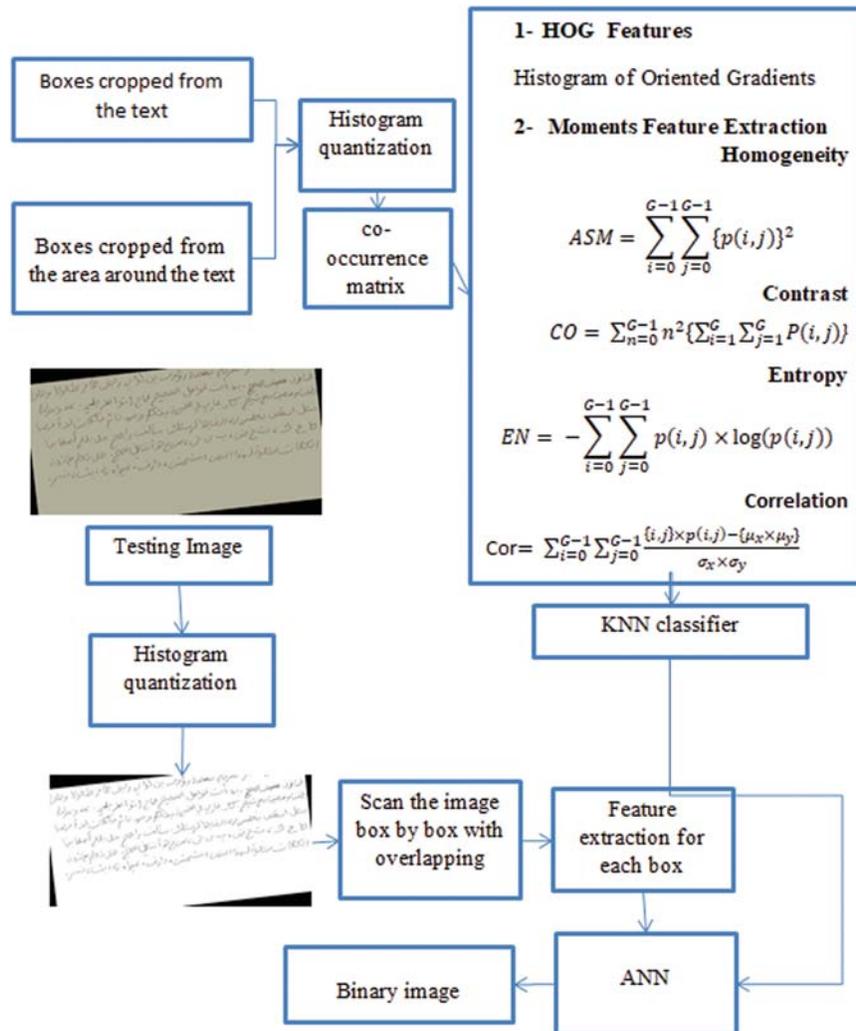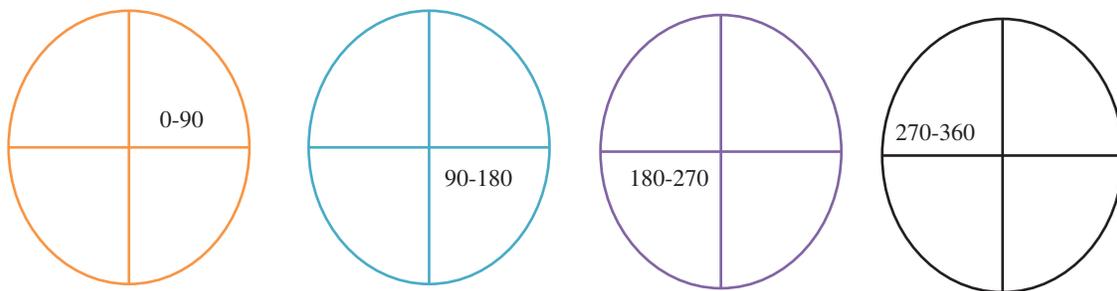
Figure 5: Framework of binarization step

Figure 6: Rotation angles

The extraction algorithm for the HOG descriptor functions over a sequence of steps. Initially, the input image is divided to minor interlinked regions (cells), and a gradient direction histogram relative to the cell pixels is created for every cell.

- Step 1: The gradient orientation is utilized to delineate each cell in angular bins.
- Step 2: The weighted gradient is provided through the pixel of each cell for its corresponding angular bin.
- Step 3: Adjacent cells are clustered into spatial regions called blocks for histogram foundation normalization and aggregation.
- Step 4: The histogram of each block represents the consequence of a histogram group that has been regularized. The descriptor is also represented through a set of block histograms.

Moments feature extraction is used to describe texture. The following four features are considered to identify each window:

Homogeneity

$$ASM = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \{p(i, j)\}^2 \tag{11}$$

Contrast

$$CO = \sum_{n=0}^{G-1} n^2 \left\{ \sum_{i=1}^{G} \sum_{j=1}^{G} P(i, j) \right\} \tag{12}$$

Entropy

$$EN = - \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} p(i, j) \times \log(p(i, j)) \tag{13}$$

Correlation

$$Cor = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{\{i, j\} \times p(i, j) - \{\mu_x \times \mu_y\}}{\sigma_x \times \sigma_y} \tag{14}$$

One of the points that have been used to support trainable models is the fusion of two features. In this process, each feature is used as an input for the ANN model; on the basis of such feature, two trained models are subsequently derived. Majority voting for the two ANNs is used to arrive at the final decision.

Image Skewing Processing as Skew detection is an important stage that can facilitate segmentation and the extraction of good features or signs for WI. To avoid known limitations, we propose a framework on the basis of the binary image produced from the initial segmentation stage. Specifically, we use image projections and the radon transform. Our results indicate that this method requires angles for estimation to detect the correct angle within a given range. Therefore, we propose a model that can estimate two angles (maximum and minimum angles) by using the model based on the histogram of the binary image. In the first stage, we use a binary image

and calculate the frequencies of the pixels that comprise the text for each line in the image. The histogram for the lines for each image is then generated. In this stage, the binary image includes 0s and 1s. This image is produced by the trainable model described in the previous stage. To detect the angle of rotation, we propose a model that can help rotate text in the right angle. In this model, we use the ANN. The training stage involves taking a number of images to train the model. These images are selected randomly with different angles to generate four classes (0–90, 90–180, 180–270, and 270–360), as shown in Fig. 6.

Fig. 7 shows the classes of skewed images. For each binary image, we extract the histogram representing the number of white pixels. Suppose that there is a binary image I; the image size is M × N, where N denotes the number of rows and M represents the number of columns. The number of bins of the histogram is M. In each row, we calculate the number of pixels comprising the text. As shown in Fig. 8, we take a sub-image and calculate its histogram.
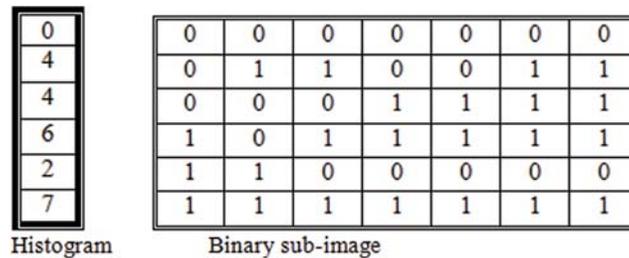

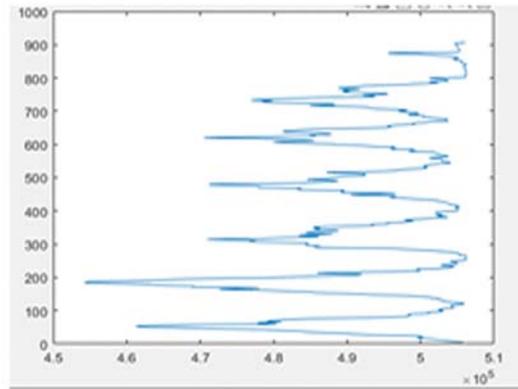**Figure 7:** Example of histogram of a sub-image

We apply the same process to three images with different rotations to show the differences between the histograms of skewed images. As shown in Fig. 8, the distribution of the histogram is present in each line of text in the image. As for the second image, we note that it is skewed. The third image shows a different histogram distribution. We thus decide to use this histogram to detect the skew angle. This histogram helps us to develop a new method that can rotate images in the correct angles. The histogram is calculated for all the training images, and the result is fed to the ANN described previously. In addition, we use the labels for the four classes with features to train the ANN model to predict labels for the new testing images.

Fig. 9 shows the whole model for identifying the angles of the skewed images. In the testing stage, we use the same process. Specifically, the testing image is used as the input for feature extraction, and the output features are used as the input for the trained model. The main objective is to identify the range of angles. Then, radon transform is used to correct the rotation.
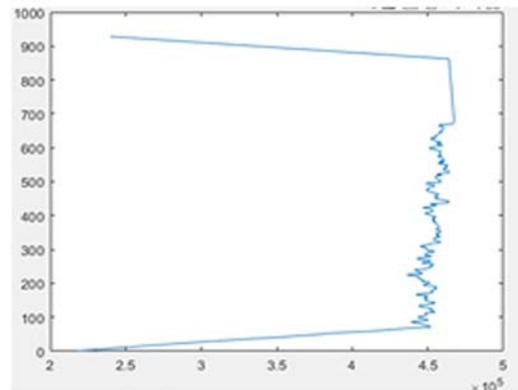
3.3 Post Segmentation Stage

We process a binary image to correct the angle of the skewed image. We calculate the histogram for the binary image by using the same process as that mentioned previously. As shown in Fig. 10, the histogram represents each line in the image and we label the histogram to show that the image includes six lines and one unwanted object. Therefore, we should use a suitable threshold that can help to distinguish the text line from the unwanted object by analyzing each line in the image.
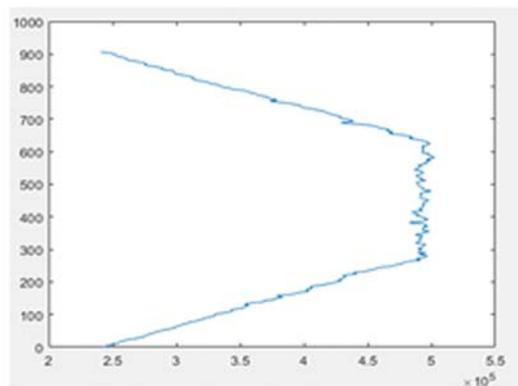
Figure 8: Histograms of different skewed images

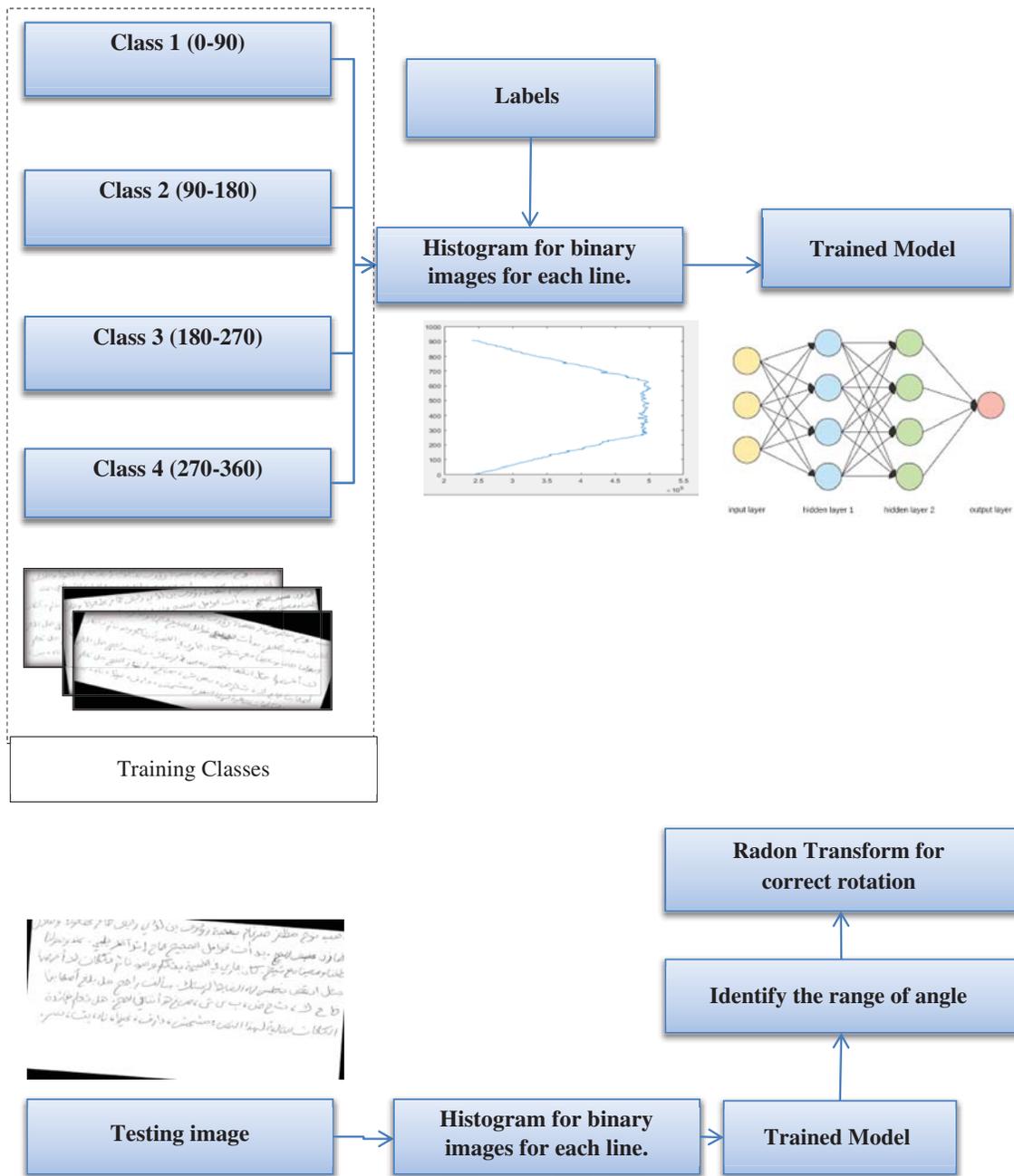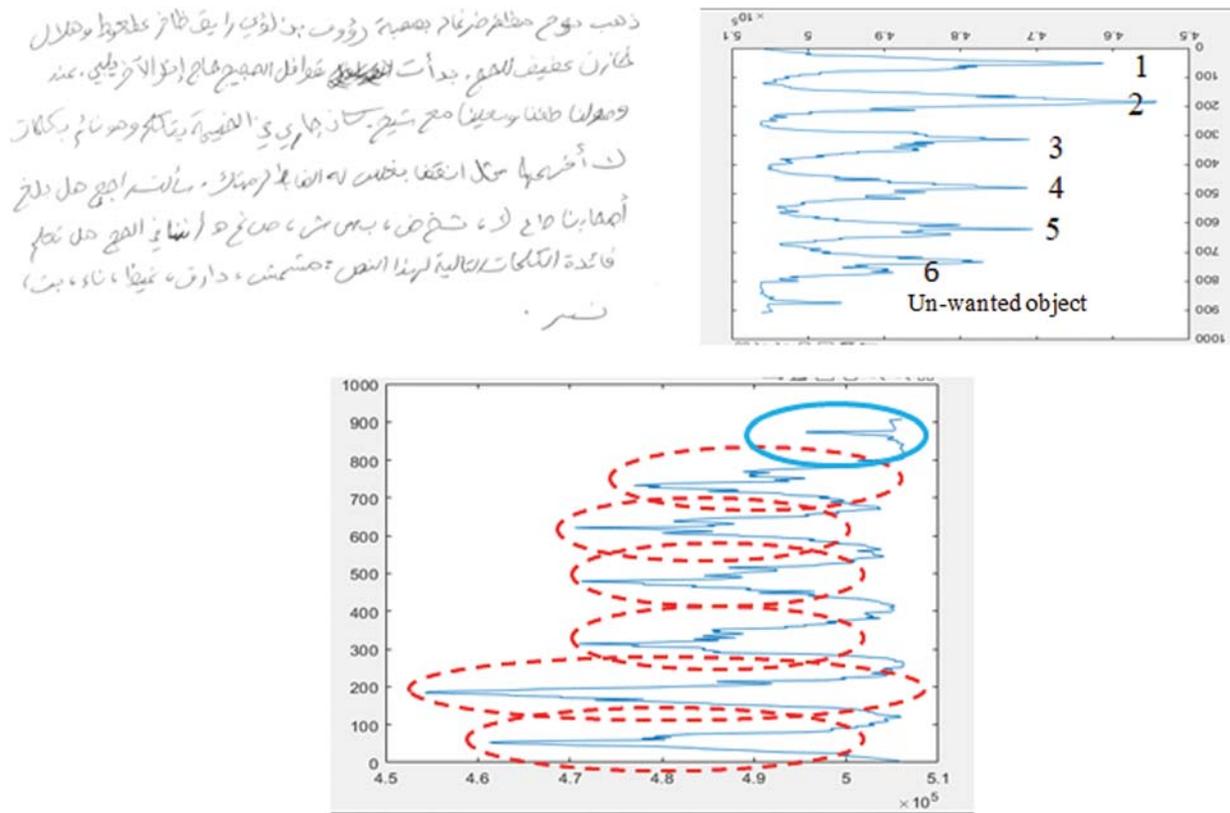**Figure 9:** Model for identifying the skew angle

Figure 10: Histogram of binary image for segmentation

4 Results and Discussion

A wide range of experiments are carried out to evaluate the efficiency of the proposed segmentation method, that is, the HOG and deep learning method for WI. We compare its execution with that of other methods in related works. The datasets utilized in this study are described in terms of the experiments used and the application of data augmentation to increase the number of samples. The execution of the proposed segmentation framework is assessed. Finally, broad tests are performed to discover the optimal trainable model for WI. The model is then compared with existing methods in terms of rank 1 identification rate and identification time.

4.1 Experiment Setup

Extensive experiments are conducted to evaluate the WI of Arabic handwriting using the hybrid trainable model. Specifically, 2D maximum embedding difference is utilized in feature extraction. The experiments are performed in MATLAB using a 64-bit Windows 10 Proficient machine equipped with an Intel Pentium 2.20 GHz processor and 8 GB RAM. The sum of 10 distinctive tests is used to evaluate WI of Arabic handwriting from different datasets.

4.2 Data Preparation

An unlimited handwritten Arabic text database published by 1,000 individual authors contains a dataset that applies to a list of KHATT's (KFUPM Handwritten Arabic TexT). This database was developed by Professor Fink from Germany, Dr. Märgner, and a research group headed

by Professor Sabri Mahmoud from Saudi Arabia. The database comprises 2,000 images with identical texts and 2,000 images of different texts with the text line images removed. The images are followed by a manually verified reality and a Latin ground-truth representation. For other handwriting recognition-related studies, including those on writer detection and text recognition, the same database may be used. For further details about the selected database, reviewers may refer to [37,38]. For academic purposes, version 1.0 of the KHATT platform is open to academics.

Description of the database: (1) forms written by a thousand different authors; (2) scanned in various resolutions (200, 300, and 600 DPI); (3) authors from various cultures, races, age classes, expertise levels, and rates of literacy with accepted writings with clear writing styles; (4) 2,000 specific images of paragraphs (text origins of different subjects, such as art, education, fitness, nature, and technology) and their segmented line images; (5) 2000 paragraphs, each of which includes all Arabic letters and types, and their line images, including related materials. Free paragraphs written by writers about their selected topics: (i) manually checked ground realities are given for paragraph and line images; (ii) the database is divided into three rare collections of testing (15%), education (70%), and validation (15%); and (iii) encourage research in areas such as WI, line segmentation, and noise removal and binarization techniques in addition to manual text recognition. Fig. 11 shows four samples from one writer, that is, KHATT_Wr0001_Para1, where the first part of the label refers to the name of the data set (KHATT), the second part refers to the number of the writer, and the third part refers to the paragraph number. In Fig. 11, we introduce four images that include texts from the same writer. The first and second images involve similar text paragraphs, and the remaining images show different texts. Fig. 11 also shows paragraphs from a second set of writers. The texts in the first and second images are the same, but they are written by different writers. As explained previously, the database includes 4,000 samples from 1,000 writers, each of whom produced 4 samples. In this study, forms are gathered from 46 various sources covering 11 various topics. Tab. 1 presents the sources' topics, the number of gathered passages in each topic, and the number of pages from which the passages are extricated. The forms were filled out by individuals from 18 nations. The distribution of the number of forms by nation is presented in Tab. 2. Individuals from Syria, Bahrain, Algeria, Sudan, Australia, Libya, Canada, Lebanon, and Oman also filled out the forms.

On the basis of the data, we build and test our proposed model. As mentioned in the previous section, the deep learning method is used to identify writers, and its features are fused with HOG features. Deep learning is powerful, but it usually needs to be trained on massive amounts of data to perform well [39–43]. Such requirement can be considered as a major limitation. Deep learning models trained on small datasets show low performance in the aspects of versatility and generalization from validation and test sets. Hence, these models suffer from problems caused by overfitting. Several methods have been proposed to reduce the overfitting problem. Data augmentation, which increases the amount and diversity of data by "augmenting" them, is an effective strategy for reducing overfitting in models and improving the diversity of datasets and the generalization performance. In the application of image classification, various improvements have been made; examples include flipping the image horizontally or vertically and translating the image by a few pixels. Data augmentation is the simplest and most common method for image classification [44–46]. It artificially enlarges the training dataset by using different techniques [47,48]. In this section, we propose an augmentation method that can help to enlarge the training data. Fig. 12 shows the proposed augmenter.

KHATT_Wr0002_Para1

KHATT_Wr0001_Para1

KHATT_Wr0002_Para2

KHATT_Wr0001_Para2

KHATT_Wr0002_Para3

KHATT_Wr0001_Para3

KHATT_Wr0002_Para4

KHATT_Wr0001_Para4

Figure 11: Sample of four images of text from the same writer. The first and second images show similar text paragraphs, and the remaining images use different text, The sample on the right shows paragraphs from second writers

As shown in Fig. 12, we use two methods for generating more samples from the training samples. The first method is based on images generated by using the line text of images and reconstructing new images. In this method, we randomly select the lines and obtain new images that include different orders of text lines. Then, rotation augmentation is performed by rotating the image to right or left on an axis between 1° and 359°. The safety of rotation augmentation is heavily determined by the rotation degree parameter. Slight rotations, such as that between 1 and 20 or between −1 and −20 could be useful in digit recognition tasks, such as MNIST. However, as the rotation degree increases, the label of the data is no longer preserved after the transformation.

Table 1: Source data's topics, paragraphs, and source pages

Category	Reference	Paragraph	Pages
Art	3	399	182
Economy	4	78	59
Education	1	46	4
Health	3	105	72
History	6	174	107
Literature	5	635	319
Management	5	86	128
Nature	7	136	70
Social	5	130	720
Technology	4	189	103
World	3	22	14
	46	**2000**	**1788**

Table 2: Writers' countries of origin

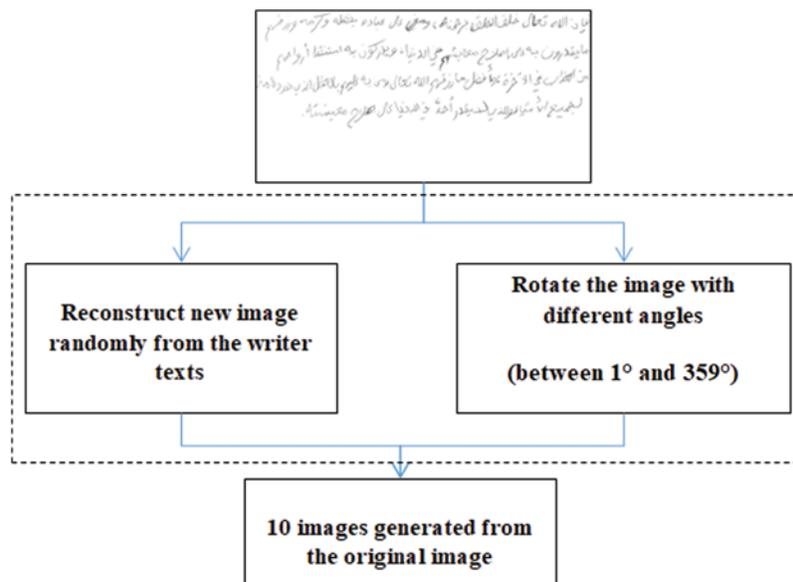| Country	Number of writers (%)	Country	Number of writers (%)
Saudi Arabia	675 = 67.5	USA	16 = 1.6
Morocco	90 = 9	Egypt	13 = 1.3
Jordan	81 = 8.1	Tunisia	13 = 1.3
Yemen	45 = 4.5	Kuwait	11 = 1.1
Palestine	29 = 2.9	Others	27 = 2.7

Figure 12: Proposed augmenter

4.3 Evaluation of Preprocessing Stage

The noise removal in the image processing refers to the presence of pixels of diverse intensities relative to the neighborhood. The noise is ordinarily discernible in smooth zones of an picture, but it debases the whole picture, involving the critical highlights like the edge. In the noise of contents and text all pixels distributed around the content don't have a place to any of the veritable content components. It is the outcome of filtering a text written on a diverse type of page employing a diverse type of ink. The determination level of the scanner could be too create undesired noises in and around the content. The scanner may thus reduce image quality and produce dark images. In addition, the quality of some images of papers written in different times is prone to be influenced by the scanner used. For the text image segmentation in our work, we try to resolution text overlap and other preprocessing stages could present artifacts that might have the similar result as text noise as they do not belong to the honest text constituents. Thus, the proposed denoising processes must be used after the preprocessing step. We propose an adaptive histogram equalization method that can help to enhance images and highlight texts to ensure clarity. Fig. 13 shows the affective result of the proposed adaptive histogram equalization. This method clearly facilitates the recovery of text from poor-quality images.
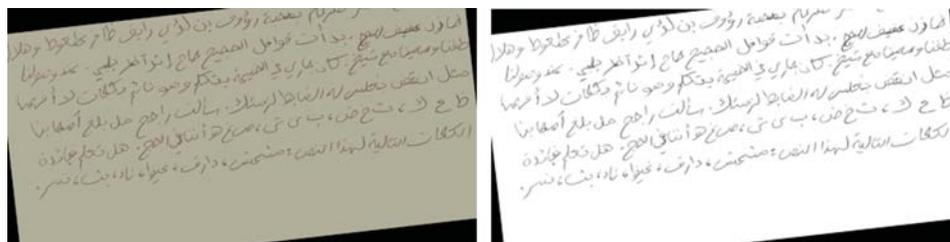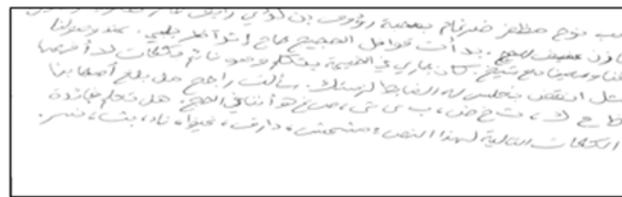
Figure 13: Stage 1 of preprocessing (adaptive histogram equalization)

Skew detection is an important preprocessing task that is used as the second stage in correcting the angle of orientation. As mentioned previously, we propose a new model for detecting image skewness. Specifically, we binarize an image by using a trainable model to segment the image and then extract the text from the rest of the image. The proposed model uses four features: homogeneity, contrast, entropy, and correlation. Fig. 14 shows the affective result of each feature on sample KHATT_Wr0001_Para1_1, which is selected randomly from the dataset.

On the basis of our investigation, we fuse the four features together to produce a powerful model that can help segment texts from backgrounds. Figs. 15 and 16 show the affective result of the proposed trainable binarization model for three samples, namely, KHATT_Wr0001_Para1_1, KHATT_Wr0013_Para1_1, and KHATT_Wr0131_Para1_1.

Skew detection is an important stage that can help realize a suitable segmentation and extract good features or signs for WI. We use machine learning to identify the angles of rotation, on the basis of which we can rotate an image. To detect angles, we propose a model that can help rotate text in the correct angle. In this model, we use ANN. The training stage involves taking a number of images to train the model. These images are selected randomly with different angles to generate four classes (0–90, 90–180, 180–270, and 270–360). Tab. 3 shows the affective result of the proposed model.
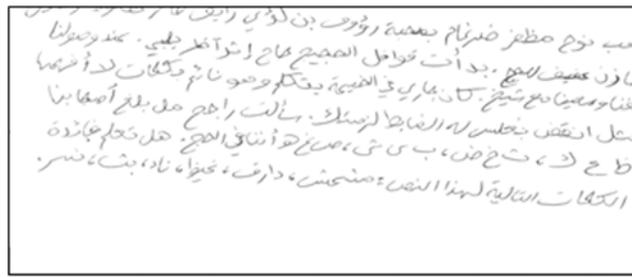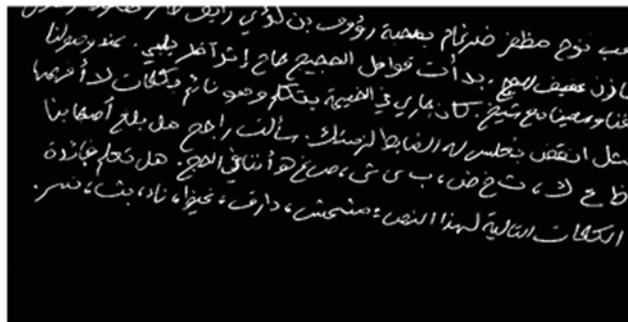
Figure 14: Affective result of proposed trainable binarization method. (A) Original image, (B) binary image based on homogeneity, (C) binary image based on contrast, (D) binary image based on entropy, and (E) binary image based on correlation

On the basis of the skew detection model, we correct the skew, as shown in Fig. 17. The proposed model estimates the angle and reconstructs the image. At this stage, the image is ready for segmentation.
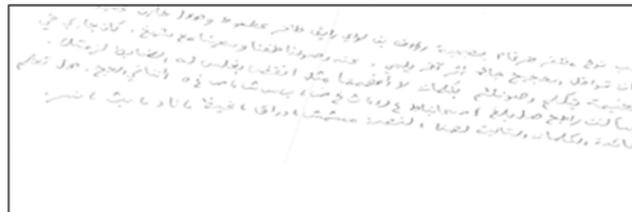
Figure 15: Affective result of proposed fusion model for sample KHATT_Wr0001_Para1_1. (A) Original image, and (B) binary image based on homogeneity for proposed fusion model
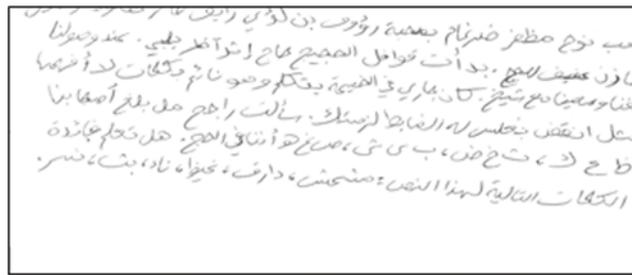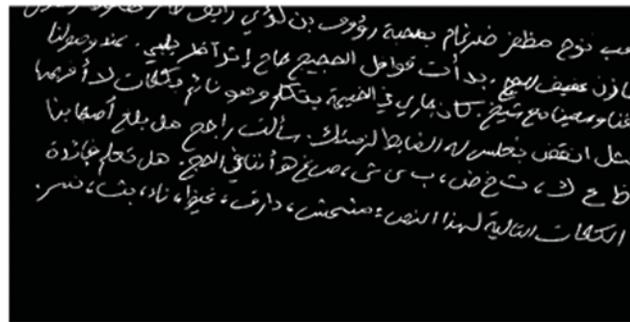

Figure 16: Affective result of proposed fusion model for sample KHTT_Wr0013_Para1_1. (A) Original image, and (B) binary image based on homogeneity for proposed fusion model

Table 3: Accuracy of skew detection model

	Number of testing images	Correct classification	Misclassification	Accuracy
Class 1 (0–90)	50	45	5	90
Class 2 (90–180)	60	52	8	86.666
Class 3 (180–270)	55	54	1	98.181
Class 4 (270–360)	50	47	3	94
Total	215	198	17	92.093

Figure 17: Correction of rotation of sample KHATT_Wr0001_Para1_1. (A) Original image, (B) binary image based on homogeneity for proposed fusion model, and (C) binary image based on contrast of rotation of sample

4.4 Segmentation Results

After preprocessing the binarized scanned text samples and correcting the text skew method to validated our goal for arabic handwriting segmentation. In this stage comprises a number of processes, begain with the segmentation process to divided the lines of the text page, fragment every line into associated components, and after that fragmenting the subwords. Thus, in every stride, we may got to move forward the result by recouping feature extraction that will get misplaced through every stride. The segmentation of the Line stage aimed to extricated content lines from paragraph(s) or the pages. The text lines or Page include single features or attributes, therefore, such this features were of no significance to our use because our focusing on small texts or lines, involving subwords and their diacritics. In any case, a few line traits offer assistance protect subwords and diacritic highlights. In specific, the incline feature may ended up debased as a outcome of the method of deskewing a page for line division. Hence, we recommend re-rotating the lines of the sample content on the premise of the initial page point. The pages segmentation

into lines and after that into characters and/or into words is commonly performed by OCR and WI analysts. We follow the similar approach in developing the suggested algorithm for WI. After deskewing the text as deliberated in the earlier sections that identify the minim arguments for horizontal projection histogram in the deskewed page text to be respected as line fragment focuses, such as we check the histogram for each picture row to recognize the valley points. The valley point could be a minima point in a flat projection content picture to be utilized as a segmented points. It encourages the automted of the method. These valleys demonstrate the space between the lines of content and are respected as fragmented line focuses. Fig. 18 shows the proposed model successfully segments an image into lines.

Figure 18: Image segmented into lines on the basis of the histogram of the image lines

5 Conclusion

This study proposes a new affective segmentation model by modifying the ANN model and making it suitable for binarization based on blocks. This model is combined with the new effective rotation model to achieve accurate segmentation through the analysis of the histograms of binary images. This work presents all the experimental results in accordance with the proposed objectives. Series of experiments are conducted using a database comprising 2,000 images with identical texts and 2,000 images with different texts and with their text line images removed. The images are followed by a manually verified reality and a Latin ground-truth representation. For other hand-writing recognition-related studies, such as those on writer detection and text recognition, the same

database may be used. The proposed technique is employed to evaluate WI of Arabic handwriting by using the hybrid trainable model. The experimental results are analyzed, interpreted, compared, validated, and discussed in depth. Diverse investigates and discussions of the outcomes are also performed. The weaknesses and strengths of the proposed system are highlighted accordingly. One of the limitations of this study is that the dataset includes few samples for each writer. This drawback prompts us to use an augmentation method to increase the number of samples for each writer. However, such method does not provide enough flexibility to extract different patterns from texts. Using more samples might help us to obtain more paragraphs that can support the method for WI using different words. For our future work, we will enhance images with other effective filters to remove noise and then highlight the ROI. Finally, the proposed method is found to perform remarkably better than those recently proposed in this area.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Dargan and M. Kuma, "A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities," *Expert Systems with Applications*, vol. 143, pp. 113114, 2020.

[2] S. A. El_Rahman, "Multimodal biometric systems based on different fusion levels of ECG and fingerprint using different classifiers," *Soft Computing*, vol. 24, pp. 12599–12632, 2020.

[3] M. Elleuch, R. Maalej and M. Kherallah, "A new design based-svm of the cnn classifier architecture with dropout for offline Arabic handwritten recognition," *Procedia Computer Science*, vol. 80, pp. 1712–1723, 2016.

[4] M. Amrouch, M. Rabi and Y. Es-Saady, "Convolutional feature learning and cnn based hmm for Arabic handwriting recognition," in *Int. Conf. on Image and Signal Processing*, Cham, Cherbourg, France, Springer, pp. 265–274, 2018.

[5] K. Fergani and A. Bennia, "New segmentation method for analytical recognition of arabic handwriting using a neural-Markovian method," *International Journal of Engineering and Technologies*, vol. 14, pp. 14–30, 2018.

[6] S. K. Jemni, Y. Kessentini and S. Kanoun, "Out of vocabulary word detection and recovery in Arabic handwritten text recognition," *Pattern Recognition*, vol. 93, pp. 507–520, 2019.

[7] S. N. Srihari and G. Ball, "An assessment of arabic handwriting recognition technology," in *Guide to OCR for Arabic Scripts*. London, UK: Springer, pp. 3–34, 2012.

[8] A. Cheikhrouhou, Y. Kessentini and S. Kanoun, "Hybrid HMM/BLSTM system for multi-script keyword spotting in printed and handwritten documents with identification stage," *Neural Computing and Applications*, vol. 32, no. 13, pp. 9201–9215, 2020.

[9] R. Ahmad, S. Naz, M. Afzal, S. Rashid and A. Dengel, "A deep learning based arabic script recognition system: Benchmark on KHAT," *International Arab Journal of Information Technology*, vol. 17, no. 3, pp. 299–305, 2020.

[10] Y. Qawasmeh, S. Awwad, A. Otoom, F. Hanandeh and E. Abdallah, "Local patterns for offline Arabic handwritten recognition," *International Journal of Advanced Intelligence Paradigms*, vol. 16, no. 2, pp. 203–215, 2020.

[11] L. Berriche and A. Al-Mutairy, "Seam carving-based arabic handwritten sub-word segmentation," *Cogent Engineering*, vol. 7, no. 1, pp. 1769315, 2020.

[12] T. Gonçalves and P. Quaresma, *Evaluating preprocessing techniques in a text classification problem*, São Leopoldo, RS, Brasil: SBC-Sociedade Brasileira de Computação, pp. 841–850, 2005.

[13] V. Srividhya and R. Anitha, "Evaluating preprocessing techniques in text categorization," *International Journal of Computer Science and Application*, vol. 47, no. 11, pp. 49–51, 2010.

[14] A. Uysal and S. Gunal, "The impact of preprocessing on text classification," *Information Processing & Management*, vol. 50, no. 1, pp. 104–112, 2014.

[15] M. Hussien, F. Olayah, M. Al-dwan and A. Shamsan, "Arabic text classification using smo, naïve bayesian, J48 algorithms," *International Journal of Research and Reviews in Applied Sciences*, vol. 9, no. 2, pp. 306–316, 2011.

[16] R. Duwairi and M. El-Orfali, "A study of the effects of preprocessing strategies on sentiment analysis for arabic text," *Journal of Information Science*, vol. 40, no. 4, pp. 501–513, 2014.

[17] M. Hadni, A. Lachkar and S. Ouatik, "A new and efficient stemming technique for arabic text categorization," in *Int. Conf. on Multimedia Computing and Systems*, Tangier, Morocco, IEEE, pp. 791–796, 2012.

[18] Y. Haralambous, Y. Elidrissi and P. Lenca, "Arabic language text classification using dependency syntax-based feature selection," arXiv preprint arXiv: 1410.4863, 2014.

[19] I. Hmeidi, M. Al-Ayyoub, N. A. Abdulla, A. Almodawar, R. Abooraig *et al.,* "Automatic Arabic text categorization: A comprehensive comparative study," *Journal of Information Science*, vol. 41, no. 1, pp. 114–124, 2015.

[20] A. Wahbeh, M. Al-Kabi, Q. Al-Radaideh, E. Al-Shawakfa and I. Alsmadi, "The effect of stemming on Arabic text classification: An empirical study," *International Journal of Information Retrieval Research*, vol. 1, no. 3, pp. 54–70, 2011.

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

[22] A. Schlapbach, V. Kilchherr and H. Bunke, "Improving writer identification by means of feature selection and extraction," in *Eighth Int. Conf. on Document Analysis and Recognition*, Seoul, South Korea, IEEE, pp. 131–135, 2005.

[23] L. Schomaker, K. Franke and M. Bulacu, "Using codebooks of fragmented connected-component contours in forensic and historic writer identification," *Pattern Recognition Letters*, vol. 28, no. 6, pp. 719–727, 2007.

[24] M. L. Bulacu, "Statistical pattern recognition for automatic writer identification and verification," Ph.D. dissertation, University of Groningen, Netherlands, pp. 140, 2007.

[25] M. Baghshah, S. Shouraki and S. Kasaei, "A novel fuzzy classifier using fuzzy lvq to recognize online persian handwriting," in *2nd Int. Conf. on Information & Communication Technologies*, Damascus, Syria, IEEE, pp. 1878–1883, 2006.

[26] G. Leedham and S. Chachra, "Writer identification using innovative binarised features of handwritten numerals," in *Seventh Int. Conf. on Document Analysis and Recognition, Proc.*, Edinburgh, UK, IEEE, pp. 413–416, 2003.

[27] M. Mohammed, S. Gunasekaran, S. Mostafa, A. Mustafa and M. Ghani, "Implementing an agent-based multi-natural language anti-spam model," in *Int. Sym. on Agent, Multi-Agent Systems and Robotics*, Putrajaya, Malaysia, IEEE, pp. 1–5, 2018.

[28] S. Mostafa, A. Mustapha, A. Hazeem, S. Khaleefah and M. Mohammed, "An agent-based inference engine for efficient and reliable automated car failure diagnosis assistance," *IEEE Access*, vol. 6, pp. 8322–8331, 2018.

[29] M. Mohammed, M. Ghani, S. Mostafa and D. Ibrahim, "Using scatter search algorithm in implementing examination timetabling problem," *Journal of Engineering and Applied Sciences*, vol. 12, pp. 4792–4800, 2017.

[30] S. Mostafa, M. Ahmad, A. Mustapha and M. Mohammed, "Formulating layered adjustable autonomy for unmanned aerial vehicles," *International Journal of Intelligent Computing and Cybernetics*, vol. 10, no. 4, pp. 430–450, 2017.

[31] S. Mostafa, S. Gunasekaran, A. Mustapha, M. Mohammed and W. Abduallah, "Modelling an adjustable autonomous multi-agent internet of things system for elderly smart home," in *Int. Conf. on Applied Human Factors and Ergonomics*, Washington, USA, Springer, pp. 301–311, 2019.

[32] K. Abdulkareem, M. Mohammed, S. Gunasekaran, M. Al-Mhiqani, A. Mutlag *et al.,* "A review of Fog computing and machine learning: Concepts, applications, challenges, and open issues," *IEEE Access,* vol. 7, pp. 153123–153140, 2019.

[33] A. Mutlag, M. Ghani, N. Arunkumar, M. Mohammed and O. Mohd, "Enabling technologies for fog computing in healthcare IoT systems," *Future Generation Computer Systems,* vol. 90, pp. 62–78, 2019.

[34] K. Huang, Z. Chen, M. Yu, X. Yan and A. Yin, "An efficient document skew detection method using probability model and q test," *Electronics,* vol. 9, no. 1, pp. 55, 2020.

[35] P. Bezmaternykh and D. Nikolaev, "A document skew detection method using fast hough transform," in *Twelfth Int. Conf. on Machine Vision,* Amsterdam, Netherlands, pp. 114330J, 2020.

[36] M. Maliki, S. Jassim, N. Al-Jawad and H. Sellahewa, "Arabic handwritten: Pre-processing and segmentation," in *Mobile Multimedia/Image Processing, Security, and Applications.* Maryland, USA, pp. 84060D, 2012.

[37] S. Mahmoud, I. Ahmad, M. Alshayeb, W. Al-Khatib, M. Parvez *et al.,* "Khatt: Arabic offline handwritten text database," in *Int. Conf. on Frontiers in Handwriting Recognition,* Bari, Italy, IEEE, pp. 449–454, 2012.

[38] S. Mahmoud, I. Ahmad, W. Al-Khatib, M. Alshayeb, M. Parvez *et al.,* "Khatt: An open arabic offline handwritten text database," *Pattern Recognition,* vol. 4, no. 3, pp. 1096–1112, 2014.

[39] M. A. Mohammed, S. S. Gunasekaran, S. A. Mostafa, A. Mustafa and M. K. Abd Ghani, "Implementing an agentbased multi-natural language anti-spam model," in *Int. Symp. on Agent, Multi-Agent Systems and Robotics,* Putrajaya, IEEE, pp. 1–5, 2018.

[40] I. J. Hussein, M. A. Burhanuddin, M. A. Mohammed, M. Elhoseny, B. Garcia-Zapirain *et al.,* "Fully automatic segmentation of gynaecological abnormality using a new viola-jones model," *Computers, Materials & Continua,* vol. 66, no. 3, pp. 3161–3182, 2021.

[41] S. Albahli, A. Alsaqabi, F. Aldhubayi, H. T. Rauf, M. Arif *et al.,* "Predicting the type of crime: Intelligence gathering and crime analysis," *Computers, Materials & Continua,* vol. 66, no. 3, pp. 2317–2341, 2021.

[42] A. A. Mutlag, M. k. Ghani, M. A. Mohammed, M. Maashi, S. A. Mostafa *et al.,* "MAFC: Multi-agent fog computing model for healthcare critical tasks management," *Sensors,* vol. 20, no. 7, pp. 1853, 2020.

[43] M. K. Ghani, M. A. Mohamed, S. A. Mostafa, A. Mustapha, H. Aman *et al.,* "The design of flexible telemedicine framework for healthcare big data," *International Journal of Engineering & Technology,* vol. 7, no. 3.20, pp. 461–468, 2018.

[44] D. M. Abdulqader, A. M. Abdulazeez and D. Q. Zeebaree, "Machine learning supervised algorithms of gene selection: A review," *Technology Reports of Kansai University,* vol. 62, no. 3, pp. 233–244, 2020.

[45] D. A. Hasan and A. M. Abdulazeez, "A modified convolutional neural networks model for medical image segmentation," *Machine Learning,* vol. 62, no. 3, pp. 233–244, 2020.

[46] D. A. Zebari, D. Q. Zeebaree, A. M. Abdulazeez, H. Haron and H. N. A. Hamed, "Improved threshold based and trainable fully automated segmentation for breast cancer boundary and pectoral muscle in mammogram images," *IEEE Access,* vol. 8, pp. 203097–203116, 2020.

[47] D. Q. Zeebaree, H. Haron and A. M. Abdulazeez, "Gene selection and classification of microarray data using convolutional neural network," in *Int. Conf. on Advanced Science and Engineering,* Duhok, Iraq, IEEE, pp. 145–150, 2018.

[48] D. Q. Zeebaree, H. Haron, A. M. Abdulazeez and D. A. Zebari, "Machine learning and region growing for breast cancer segmentation," in *Int. Conf. on Advanced Science and Engineering,* Duhok, Iraq, IEEE, pp. 88–93, 2019.