OFFLINE SIGNATURE RECOGNITION USING SPATIAL METHOD DISTRIBUTION

Shahad S. Hadi¹, Nassir H. Salman², Loay E. George³

¹Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq.
²Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq.
³University of Information Technology and Communication for Scientific, Baghdad, Iraq.

¹um3sobi@scbaghdad.edu.iq, ²drnassir@sc.uobaghdad.edu.iq, ³loayedwar57@yahoo.com

Abstract

There has been challenging the pattern recognition that more attention needs to be paid to this area Offline Signature Verification (OSV), particularly when it is relied upon to popularize fully on the skilful frauds that are not accessible during the preparation. Its difficulties additionally incorporate little training tests and great intra-class divergence. At times the crude signature can incorporate additional pixel known as noises or may not be in the legitimate structure where preprocessing is obligatory. Insomuch as a signature is preprocessed accurately, it leads to a superior outcome for both signature matching and fraud disclosure. For example; an appropriate estimation of gamma value improves the contrast of the signature image, on another hand, Preparing likewise comprises binarization, noise elimination, so forth...The proposed method is for extraction features (such as ;Energy, Contrast, Entropy, and Correlation) from Offline Signature Verification System. In this paper, the data processing deals with twain parallel styles viz signature training and signature testing analysis. Insomuch as that the extracted features from a signature picture doesn't powerful, this will cause higher verification error rates particularly for skilful fabrications in hacking the system. The results show that’s the (UTSig) and the combination of (NISDCC, CEDAR, SigComp2012). Comparing with the other researches, the results in this Paper is the best and the system is more efficient with (UTSig) signature which were 97%.

Keywords: Offline Signature Verification, Insomuch, estimation of gamma value, twain parallel styles, UTSig, NISDCC, CEDAR, SigComp2012.
I. Introduction

Biometrics has been assigning automatically utilized approaches for authenticating entity and identity. With regard to traditional identification systems, with features (passwords and ID cards) might be stolen or forgot, the biometric systems have been on the basis of behavioral or physiological features related to individuals which have been difficult for other individuals to imitate; therefore, decreasing the possibility of forgery. Each individual’s biometric characteristics have been distinctive and might not be broken, stolen, or lost (Gunjal, Dange, & Brahmane, 2016). Biometrics has been considerable benefit over the conventional approaches (smart cards, PIN numbers, passwords, and so on) because the individual’s biometric characteristics not simply transferable have been distinctive for each individual and might not be broken, stolen, or lost. Selecting single biometric solutions are according to many factors including (acceptance of users, required level of security, accuracy, and implementation time and costs) (Kaur & Kaur, 2014). The aim of the signature verification systems have been differentiating between forged and original signatures, associated to inter personal and intra personal variability (Mahanta & Deka, 2013). Also, the intra personal variation has been a distinction between same person’s signatures, while inter personal has been variation between forgeries and originals (Rashidi, Fallah, & Towhidkhah, 2012). There is going to be some slight variations related to hand-written signature of humans, the consistency that is created via natural motion as well as practicing generating certain pattern which make hand-written signature adequate for the biometric identifications. Furthermore, the signature forgery indicating an attempts for copying the signature of another person and using it for stealing their identity (Kaur & Kaur, 2014).

Related Work

Some of the articles relevant to this work were previously published. Soleimani et al. (Soleimani, Araabi, & Fouladi, 2016) Presents a novel classification method, Deep Multitask Metric Learning (DMML), for offline signature verification. Unlike existing methods that to verify questioned signatures of an individual merely consider the training samples of that class, DMML uses the knowledge from the similarities and dissimilarities between the genuine and forged samples of other classes too. To this end, using the idea of multitask and transfer learning, DMML train a distance metric for each class together with other classes simultaneously. DMML has a structure with a shared layer acting as a writer-independent approach that is followed by separated layers which learn writer-dependent factors. We compare the proposed method against SVM, writer-dependent and writer-independent Discriminative Deep Metric Learning method on four offline signature datasets (UTSig, MCYT-75, GPDS synthetic, and GPDS 960 Gray Signatures) using Histogram of Oriented Gradients (HOG) and Discrete Radon Transform (DRT) features. Results of our experiments show that DMML achieves better performance compared to other methods in verifying genuine signatures, skilled and random forgeries. Soleimani et al. (Soleimani, Fouladi, & Araabi, 2016b) used fixed-point arithmetic, which is described in (Ferrer, Alonso, & Travieso, 2005) as ‘description of the signature envelope and the interior stroke

Copyright reserved © J. Mech. Cont.& Math. Sci.
Shahad S. Hadi et al
distribution in polar and Cartesian coordinates’. In fixed-point arithmetic feature extraction by using the geometric centres of samples, three parameters are calculated in polar coordinate: derivative of radius of signature envelope, its angle, and the number of black pixels that the radiuses cross when rotated from one point to the next point. In Cartesian coordinates, height, width, and the number of transitions form black to white or white to black pixels of signatures and are calculated with respect to their geometric centres. Younesian et al. (Younesian, Masoudnia, Hosseini, & Araabi, 2019) Proposed OSV includes three steps: feature learning, active learning, and final verification. We benefit from transfer learning using a pre-trained CNN for feature learning. We also propose SVM-based active learning for each user to separate his genuine signatures from the random forgeries. We finally used the SVMs to verify the authenticity of the questioned signature. We examined our proposed active transfer learning method on UTSig: A Persian offline signature dataset. We achieved near 13% improvement compared to the random selection of instances. Our results also showed 1% improvement over the state-of-the-art method in which a fully supervised setting with five more labeled instances per user was used. Araabi et al. (Soleimani, Fouladi, & Araabi, 2016a) inspired by the cursive nature of Persian signature, and motivated by the opinion of Persian forensic handwritten experts that put emphasis on this factor, we propose a framework for Persian offline signature verification that uses histogram of curvature (HOC) and histogram of oriented gradient (HOG) as features. Calculating curvature for offline signature due to lack of temporal information is not a straightforward task. To this end, we use a discrete curvature estimator that works by normalized gradient and its Jacobian matrix. We investigate the effect of preserving signatures’ scale, center of gravity, grid size, grid overlapping, number of bins, and smoothing factor on the accuracy. Support vector machine (SVM) is used to train writer-dependent classifiers on genuine signatures and random forgeries as training samples. Narwade et al. (Narwade, Sawant, & Bonde, 2018) In this proposed approach, the signature pixels are represented by: (1) Gaussian Weighting Based Tangent Angle, to represent the curve angle at the reference pixel; (2) a new shape descriptor, i.e. cylindrical shape context is proposed for a detailed and accurate description of the curve at corresponding pixel. Experimental results show that desired pixel matching results is obtained by using cylindrical shape context which automatically increases the accuracy of verification of offline handwritten signatures. The shape dissimilarity measures are computed and given to the Support Vector Machine with Radial Basis Function (RBF) kernel for classification of signature.

Biometric System Architecture

This problem can be formulated as a Pattern Recognition task: given a dataset of genuine signatures from the users who have been enrolled in the system, a classifier is trained for a user so that it can discriminate new samples (not seen during training) as genuine signatures or forgeries. It follows a standard Pattern Classification pipeline:

1. **Data acquisition:** The first step is to obtain the data. For the problem at hand, this is accomplished by scanning the documents containing the signatures.
2. **Preprocessing:** After the document is scanned, the signature is obtained from the document and image processing techniques are applied to reduce noise or in general, enhance the quality of the samples.

3. **Feature Extraction:** This step consists of obtaining a set of measurements from the samples. Given a preprocessed signature image, this step produces a real-valued feature vector.

4. **Training:** Once we have feature vectors from the samples, we train a machine learning model, by optimizing its parameters to minimize a cost function in the training set (Hafemann, 2019).

The initial steps of data acquisition and signature extraction are not considered in the majority of the studies that already consider individual signature images as the input for the systems. As an exception, this has been explored for bank cheque that contains a complex background. Other preprocessing steps vary among different studies, but the majority use simple techniques like noise removal, size normalization and centering, and a variety of morphological transformations such as binarization and thinning (Hafemann, 2019). The system workflow comprises dual main stages: (a) enrollment phase and (b) identification and verification phase. In the enrollment stage, the system is trained to the identity of every person using its characteristic features. While in the cognition stage the system works either as an identification process or a verification process. The layout for the suggested system has been illustrated in figure below (1.1); it includes three stages: Signature preprocessing, feature extraction using local distribution for statistical features to each block, and matching stage using similarity measures.

![Fig. 1: SYSTEM MODEL](image_url)

**Data Acquisition**

The acquisition of the image of the signature is one of the crucial stages of all recognition systems. The skew, resolution, and the isolated elements of the signature increase the complexity of the issue (V.G. & Patil, 2014). The image of the signature is input into the system as bitmap file (.bmp). The input images’ color resolution is 24bit/pixel. The data of the image (in other words, the elements of the RGB components) is loaded. The collection of the signatures is gathered with the use of the blue or black inks within a rectangular 5x3cm box. Every one of the persons gives a
minimum of 10 signature samples with the maximal possibility of variations. After that, the loaded color components are used to compute the gray image variant in the preprocessing stage.

**Preprocessing Stage**

Initially, the signature is taken and converted to a format which a computer is capable of processing. After this stage, the image is prepared to be pre-processed (Kaur & Kaur, 2014). At this step, the signature’s RGB image is transformed to gray-level and afterwards to the binary. The aim of such stage is making the signatures prepared for extracting the features (Karouni, Daya, & Bahlak, 2011). The process of the pre-processing comprises the following procedures: thresholding, skeletonization, clipping and segmentation. The thresholding is performed, the main element in thresholding is the selection of the value of the threshold. After that, the signature’s skeleton is separated through the application of the matching morphologic process as an attempt of filtering the maximum amount of the undesired noise which has originated from the characteristics of the device, in other words, the writing tool’s variability. The experimental process indicated the fact that in the case where the algorithm of one pixel wide thinning is utilized, after that, there will be a decrease in the informative feature components’ content affecting its discriminative powers. Thereby, the procedure of the skeletonization produces the traces of the signature with greater than one pixel wide, which typically range between 3 and 5 (Pirlo et al., 2014). Therefore, the step of the pre-processing is viewed to be highly fundamental and significant over the system of signature verification. Typically, the image of the signature is made ready for the feature extraction succeeding numerous steps of preprocessing:

**2.3.2 Grayscale Images**

Grayscale images have just information related to the intensity of light, yet not the color. There have been a lot of approaches for generating gray-scale images, often referred to as black and white images (S. A. Tuama et al., 2019). Usually, there are 3 channels related to the color images, also the information might be mixed in to single channel, store the intensity information, and also lose color information. Red, green, also blue parts have been major color space with regard to the digital images, indicating that the RGB’s intensity of an image has been independently stored. After utilizing the average related to such 3 channels, gray-scale version of images will be obtained. Colored images might be changed in to greyscale ones with the use of the next equation (Pratt, 1994):

\[
Gry(x, y) = \frac{1}{3} \left( I_{\text{red}}(k,l) + I_{\text{green}}(k,l) + I_{\text{blue}}(k,l) \right)
\]  

(1.1)

In which \(Gry()\) representing created grey image. The next figures showing the converting:
2.3.3. Digital Image Statistical Operations

Thorough analysis was provided with regard to statistics which has been utilized in the suggested system, also in certain published systems, have been provided as follows:

A. Mean

The mean of image (referred to as average), this could contributing to such trend individual pixel intensity with regard to the whole image (S. Tuama & George, 2016). The number is telling where distribution has been centered, typically referred to as \( \mu \). It has been estimated with the use of (Taylor & Cihon, 2004):

\[
\mu = \frac{1}{N} \sum_{i=1}^{N} X_i = \frac{1}{N} (X_1 + X_2 + X_3 + \cdots + X_n)
\]  

(2.2)

In which \{\(X_i\)\} representing samples values, also \(N\) representing number Regarding the samples (samples set size).

B. Standard Deviation and Variance

Variance utilize due to the evaluation of variability, referred to as \(\sigma^2\). Put differently, it has been an evaluation regarding the way that the distribution has been spreading out from the mean (\(\mu\)) values; it has been specified as:

\[
\sigma^2 = \frac{1}{N} \sum_{i=1}^{N} (X_i - \mu)^2
\]  

(2.3)

In which \{\(X_i\)\} representing samples values, \(\mu\) representing all samples mean, also \(N\) representing the number of samples. Large \(\sigma^2\) value indicating that data have been widely spread, whereas small value \{\(\sigma^2\)\} indicating the data have been spread close together. Standard deviation value \{\(\sigma\)\} has been square root related to variance; it was specified via the equation;

\[
\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (X_i - \mu)^2}
\]  

(2.4)

2.3.4. Digital Image Enhancement

The enhancement of images is significant approaches in image processing. Also, the enhancement of images have been an approach indicating certain

Shahad S. Hadi et al
information images, also weakening or removing certain secondary information, aiming for improving identification’s quality in the process. The enhancement of images majorly involving such parameters: Brightness, Contrast, Sharpness, Saturation.

2.3.5. Gamma Correction

Attempting on enhancing the image require selection of adequate gamma value for ensuring adequate contrast enhancements for image’s input. The process of correction includes utilizing the next equation:

$$G'(x,y) = 255 \left( \frac{G(x,y)}{255} \right)^\gamma$$ \hspace{1cm} (2.5)

In which $G(x,y)$ has been unique pixel rate, $G'(x,y)$ has been pixel intensity as it is appearing on the display, and also $\gamma$ has been gamma value. Figure (3) after utilizing gamma:

Fig. 3: Convert to gamma

2.3.6. Contrast Stretching

Contrast stretching has been linear contrast enhancement operations utilized for lessening variances between image’s gray level values that might be indicated as the accuracy each one of gray level values(Aldhaher & George, 2014). With regard to linear contrast enhancement, there have been various approaches like piecewise, percentage, also min-max of (linear contrast stretching). The estimation of current intensity interval has been managed via minimum “G-min” as well as maximum “G-max” values, after that such interval will be stretched for covering full brightness range [0,255].

$$G_{\text{min}} = \mu - \alpha_1 \sigma$$ \hspace{1cm} (2.6)

$$G_{\text{max}} = \mu - \alpha_2 \sigma$$ \hspace{1cm} (2.7)

In which, $\alpha_1$ & $\alpha_2$ have been multiplying for representing the degree of gray levels deflection from mean at bright and dark sides of the image. Furthermore, the ($\mu$ & $\sigma$) representing standard deviation and mean values, related to gray image. The used mapping function with regard to such enhancement type has been specified via the equation (2.8):

$$G_{\text{out}}(x,y) = \begin{cases} 0 & \text{If } G \leq G_{\text{min}} \\ 255 \left( \frac{G_{\text{in}} - G_{\text{min}}}{G_{\text{max}} - G_{\text{min}}} \right)^\gamma & \text{Otherwise} \\ 255 & \text{if } G \geq G_{\text{max}} \end{cases}$$ \hspace{1cm} (2.8)

Copyright reserved © J. Mech. Cont.& Math. Sci.
Shahad S. Hadi et al
In which, \( G_{in} \) & \( G_{out} \) have been the gray levels input as well as the output pixel. Also, \( G_{min} \) & \( G_{max} \) have been minimum and maximum gray level related to input image; \( \gamma \) representing gamma value (Shakour, 2018).

**2.3.7 Digital Image Binarization**

The image is binaries that is signature is embodied in practice have one color for text (black pixels) and other areas are in (white pixels) for background (Radhiaka & Gopika, 2015). Threshold is of high importance in binarization as well as choosing fitting threshold value has been central one. Greyscale image pixels values are extending from (0-255) (Eds, 2018). Total of 127 thresholds has been utilized for attaining the required binary images utilizing Equation (2.9):

\[
Bin(x,y) = \begin{cases}
0 & \text{if } Gray(x,y) < Thr \\
1 & \text{if } Gray(x,y) \geq Thr
\end{cases}
\]  

(2.9)

![Fig. 4: Showing binary image's example, in which the signature specified the object which requires recognition, also has white color, whereas the image's background has been black.](image)

**2.3.8. Morphological Image Processing**

The morphological image processing has been an approach to extract or modify information on structure or shape of objects in the image. Also, the morphological operators, like skeletonization, erosion, and dilation have been adequate for binary image’s analysis, yet they might be utilized with the gray-scale images. The morphological operators have been nonlinear; also common usages include find object’s mid-line, lessening of noise, count image’s objects, feature detection, edge detection, in addition to filtering (Sindhu & Jeeva, 2013).

**2.3.8.1. Seed Filling**

This is a process that has many names. With regard to graphics community, flood fill and seed fill utilized for such process. The seed filling approach that belonging to growing type of the segmentation approaches utilized for dividing binary image in to isolated regions on the basis of pixel’s values (pixel’s colors), where the set of the connected pixels with same color, also extended in the bounded region has been specified as isolated segment (AL-OBIADIE, 2016). Seed filling method starts with allocating a seed point, and then collects all object points that are connected, directly or indirectly, with that seed point. Seed filling operation is applied to pick up the unwanted small noise patches and to remove them, and then filling the
holes. To eliminate the noise patches, the four connected neighbors of each signature pixel (e.g. pixel value is one) are checked to see if they are signature pixels or not.

![Fig. 5: Seed Filling](image)

2.3.8.2. Thinning and Skeletonization Algorithms

Producing a form that is resembling the original form through not providing another novel feature related to original form, not it is eliminating object’s features (Widiarti, 2011). The Skelton algorithm "engaged two the scanning in each execution for each one of the pixels. With regard to first scan, the edge related to the while right edge as well as left edge point have been indicated as undependable spot for the deletion. A similar procedure will be iteration for every one of the points in bottom and top edge point at the second scan. Also, the cancellation has been achieved so that there has been no marking point.

![Fig. 6: Skeleton Algorithm](image)

2.3.8.3. The Thinning Algorithm of Zhang-Suen

This method of skeletonization is a parallel approach which indicates that the new value which has been obtained is only dependent upon the preceding value of the iteration. It is simple and fast to implement. The approach is carried out using two sub-iterations. Ultimately, these pixels which satisfy those settings are going to stand

Copyright reserved © J. Mech. Cont.& Math. Sci.
Shahad S. Hadi et al
eliminated. If these algorithm stops at the end of any one of the sub iterations, there isn’t any pixel to delete (Ahmed, 2018).

2.3.9. Image Clipping

Clipping is responsible for eliminating those parts of the scene which do not locate in the window rectangle, because they are outside the interesting volume for next stage task. Such a procedure is implemented to order the numeral image via the removal of the right, left, bottom, and top spaces in a way that the new boundaries of the image confine the zone of the numeral object (Mohammed & George, 2016). The target of this process is locating the actual area in the signature image describing the signature ROI and gets rid of the regions of an image covering irrelevant information.

![Clipping Algorithm](image)

Fig 7: Clipping Algorithm

2.4 Image Partitioning – Blocking

In order to facilitate pattern classification, and to detect the local features of the image and afford a primeval representation scheme; in the proposed system the signature image is divvied to blocks, after that, the local features of every one of the blocks are determined. The block dimensions are determined by dividing the image to several blocks that have equal sizes, and the value of length is predefined as a ratio of block length. The consequences of both the number of blocks is examined to figure out their apt values, which should lead to unsurpassed recognition rate (Fadhil & George, 2017). Horizontal scan is thorough from leftward to right & from upper to undermost during image blocks (HASSAN, GEORGE, & MOHAMMED, 2018).

2.5. Feature Extraction

The progression of Features Extraction can be defined as a significant portion of somewhat system of pattern recognition. The procedure by which the digital information undergoes modification, simplification, combination in a way that permits the classification of the salient information, is referred to as the feature extraction. To facilitate the analysis of this problem, features will be classified into three categories:

- Physical features such as color, and brightness of a region of pixels.
- Structural features such as shape, texture, and other geometrical properties of patterns.
• Mathematical features such as correlation coefficients, statistical means, Eigen value and the covariance matrices’ Eigenvectors, and other invariant characteristics.

In automatic pattern recognition, physical and structural features are used primarily in the image processing field (Ellen, Day, & Davies, 2018).

After passing through the pre-processing steps, the stage of feature extraction begins. In the cases where input data is too large to process and is expected to have redundancy (in other words, a big amount of data, but a small amount of info), already that input data can be converted to a summary illustration set of features (which is referred to as feature vector as well). In the case where the obtained feature set is carefully selected, the expectation is that this features set is going to obtain useful information from input data for the sake of performing the wanted task. This is carried out by the use of a reduced representation rather than the input of the full size. The texture can be described as one of the most widely utilized features in the interpretation of the medical images; it can be applied to many different problems of image processing. Many texture analysis approaches were developed over the past years; the co-occurrence matrix features such a method have been used in this work. Haralick has proposed a set of 14 texture characteristics that may be obtained from the matrix of co-occurrence, containing the knowledge on the textural features of the image. In the presented study, 4 Haralick feature functions have been calculated. Which are: Entropy, Energy, Correlation, and Contrast (Hamza & Al-Assadi, 2012). The texture features of the Gray level co-occurrence suppose that the information of the texture in an image is included in the general spatial correlations amongst image pixels. This is carried out by initially specifying the GLCM (George, Al-Daamy, Al-Daamy, & Ahmed, 2016). The co-concurrence matrix features are:

A. **Energy (Angular-Second-Moment (ASM))**:  
Which performs the measuring of an image uniformity. In the case where the pixels are very alike, the ASM, which is referred to as the energy or uniformity, value is going to be high (Jabur & Ali, 2014).

\[ \text{ASM} = \sum_{|i-j|=0}^{N_g-1} \sum_{j=0}^{N_g-1} [p(i,j)]^2 \]  
(2.11)

B. **Contrast (C):**  
Is measures the intensity or the variations of the gray-scale amongst the reference pixel & its neighbor. In the real world visual perception, the contrast is specified by differences in the object brightness and color as well as different objects within a similar field of view.

\[ C = \sum_{n=0}^{N_g-1} n^2 \left( \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} p(i,j) \right), \ |i-j| = n \]  
(2.12)

C. **Correlation (Cor):**  
The feature demonstrates the gray level values’ linear dependency in the GLCM. It presents the way a reference pixel is associated with its neighbor, 1 is perfectly correlated and 0 is not correlated.

\[ \text{Cor} = \frac{1}{\sigma_x \sigma_y} \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} i j p(i,j) - \mu_x \mu_y \]  
(2.13)
D. **Entropy (H):**

Entropy can be defined as a statistical randomness measure which may be utilized for the representation of the input image texture.

\[ H = - \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} p(i,j) \log_2(p(i,j)) \]  

\[ (2.14) \]

### 2.5.1. Distance Matching

In the proposed signature identification system, the classifier computes an expanse of the participation signature from entirely sample signatures in the training set. In the case where the tiniest distance is not more than a predefined threshold value the class of the input signature is the same as the class of the nearest distance to the signature in the training set. Euclidean Distance is a famed technique that estimates the distance points which is used to find the distance between two signatures (Sigari, Pourshahabi, & Pourreza, 2012). In this stage we had used the nearest neighbor classifier to classify each input image to its class. This is done by comparing input image with stored templates (classes), and it would be classify according to the class of its nearest neighbor (sample) in feature space. In this work the Euclidean measures of distance were utilized for the determination of the level of the similarity between feature vectors, which has been obtained from tested samples, with template feature vectors, every one of the representing specific classes, which are kept in the database. The Model of this Euclidean Distance is utilized for the verification. That classifier is sufficient for the obtained features & faster in the calculation. This is the modest distance between two n-sized vectors. Assuming that \( X(x_1, x_2, \ldots, x_n) \) and \( Y(y_1, y_2, \ldots, y_n) \) are two n-sized vectors. The distance (d) can be calculated by using of the subsequent equation:

\[ \text{Distance (D)} \quad d_E = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \]  

\[ (2.16) \]

The first measure is the measure of the length between end-points. It is similar to the measurement of the distance between a pair of points with the use of a ruler.

### 2.5.2 The Generation of the Feature Vector

The feature vector represents a vector of n-dimensions, which has numerical features representing an object. The flow-chart which has been illustrated in Fig. (2.16) depicts the feature vector generation process (Inamdar, Rege, & Arya, 2010).

---

**Fig. 8:** Feature Vector Generation Flowchart.

It basically includes two steps, which are the preprocessing and the feature extraction. As it has been stated earlier, the preprocessing is carried out on signature images from a data-base for the sake of preparing it for feature extraction process and for ensuring
the fact that all of signature images are equal in their dimensions, in order to make it easier and more convenient extracting those features. Figure below illustrates the feature vector generation flow-chart.

**Fig. 9**: The flowchart of the typical System of Automatic Signature Verification

### 2.5.3. Verification

Verification is the third stage in any verification system after preprocessing and feature extraction as shown in figure (2.16) system diagram. In verification a decision is making either the input image is original or not. The arrangement would match of an obtained feature vector of the input signature image with corresponding templates which have been stored in data base and return decisions that verify the claim of that person. This work adopts approaches for the signature verification, is Euclidean Distance method.

### Data set

Participants have scanned the forms with 600 dpi resolution, and stored as 8-bit greyscale TIF files, and selected from Iranian students of University. Their ages were between 18 and 31 and 90% of them were right-hand writers. Genuine participants consisted of 100% males; however, 40% female and 60% male forgers participated. UTSig dataset consists of 8280 images from 115 classes. Each class belongs to one specific authentic male person and has 27 genuine and 45 forged samples of his signatures (Soleimani, Fouladi, & Araabi, 2017).

### System Requirements

The proposed system has been built using a C# programming language as a windows application. According to the following requirements:

a) **Hardware Specs**: The system shall work on any machine with a minimum of 1GB of memory, 2.1 GHz processor.

b) **Working Environment**: System shall work on any machine running Windows XP, Windows Vista, Windows 7, Windows 8, or Windows 10.

c) **Software packages**: Microsoft .Net framework 4.5 or newer version is needed to run the system.

### Implementation-of- Proposed System

In this part, the proposed system will apply on numbers of samples for signature with dataset contains 55 people for each person 24 samples of signature.
3.1. Training of Dataset

The first part of system implementation is to use the dataset for the training system. Load dataset and run system to train a dataset and extract features of each signature then store in the database.

3.2. Implementation of Testing

When load training database to the system the testing step begun with load signature to test. The second step is preprocessing, at first change color image to a gray. Then apply gamma to image enhancement, then apply binarization. The next step is to apply the seed filling algorithm to remove the image’s noise, and then apply the skeleton (thinning) algorithm. Then apply clipping algorithm to remove anything otherwise signature part. Then split image to set of blocks and apply features extraction for each block from testing signature by applying equation for Energy, for extract Contrast, extract Correlation, and for extract Entropy, as in table 3.1.

| Block | Energy     | Contrast    | Correlation | Entropy       |
|-------|------------|-------------|-------------|---------------|
| 1     | 0.240625957| 0.240854673 | 0.3984682   | 0.526049151   |
| 2     | 0.238850155| 0.230141737 | 0.468021906 | 0.567251147   |
| 3     | 0.24321391 | 0.286717514 | 0.295688388 | 0.50772761    |
| 4     | 0.255661237| 0.356274381 | 0.255114716 | 0.536809084   |
| 5     | 0.231073183| 0.203886576 | 0.436686605 | 0.510701889   |
| 6     | 0.23582847 | 0.267052583 | 0.417262621 | 0.528925628   |
| 7     | 0.273451591| 0.357452066 | 0.405375569 | 0.581484233   |
| 8     | 0.265339579| 0.403120998 | 0.32162684  | 0.603379676   |

Table 1: Table 3.1. Sample of features extraction from testing signature

Features Extraction for some signers in figure (7)

Fig. 10: Features Extraction for some signers in figure
Matching step: by applying matching algorithm by taking six people as a sample to review table 1.2 each row in a table represents the mean of features of all training signatures of that person.

|       | Energy       | Contrast     | Correlation  | Entropy     |
|-------|--------------|--------------|--------------|-------------|
| Person 1 | 0.048998684  | 0.217840733  | 0.026098496  | 0.072178495 |
| Person 2 | 0.040165747  | 0.049231085  | 0.060762133  | 0.046119352 |
| Person 3 | 0.026321734  | 0.136609273  | 0.095321478  | 0.945687431 |
| Person 4 | 0.035069189  | 0.168047962  | 0.081254789  | 0.137450087 |
| Person 5 | 0.951236547  | 0.963547581  | 0.093214783  | 0.987451236 |
| Person 6 | 0.035389943  | 0.119441469  | 0.069845321  | 0.085147869 |

When test the below features extraction from signature table 1.3.

|       | Energy | Contrast | Correlation | Entropy |
|-------|--------|----------|-------------|---------|
|       | 0.125987 | 0.98741  | 0.654789    | 0.98745 |

The minimum value will appear with person 5 as in table 1.4.

| Person 1 | Person 2 | Person 3 | Person 4 | Person 5 | Person 6 |
|----------|----------|----------|----------|----------|----------|
| 1.831137735 | 2.126516272 | 1.048543063 | 1.731061541 | 0.996971831 | 1.917885441 |

A. Matching Stage

In biometric system, usually a discriminating description for the captured sample is established, and then a search for the database element that show best match with the captured sample is done. This is classified as a process to associate the proper class label with the testing sample according to measures describing it. One of the ways of making the correlation is through locating the class member with the measurements differing as low as possible from the measurements of the test sample. Thereby, the class is assigned through the analysis of similarity distance. There have been considerable efforts in finding the appropriate measures among such an excessive amount of choices because it is of essential stage to pattern classification.
The choosing of distance/similarity measure depends on the type of measurement or objects representation.

B. Verification

Verification is the third stage in any verification system after preprocessing and feature extraction as shown in figure (2.16) system diagram. In verification a decision is making either the input image is original or not. The arrangement would match of an obtained feature vector of the input signature image with corresponding templates which have been stored in data base and return decisions that verify the claim of that person. This work adopts approaches for the signature verification, is Euclidean Distance method.

C. Euclidean Distance

This stage is utilized to compute similarity between the test image and templates that have been stored in the data-base. The next task is to find a similarity score using the equation (2.16), thus if the distance between test and data-base images is small then Test image is of a higher similarity to the data-base image. The verification steps are shown in Algorithm.

➢ Compute the distance between the Test Signature templates with all stored templates.
➢ Search the Minimum Distance value between all templates.

3.5. System Performance

The system rate was checked on a different number of data sets several times and the best result was taken. A data set that has been relied on was (UTSig) one of the rather modernistic Iranian signatures collection. Which consists of (115) signers with (27) real signing up, three opposite-hand signatures, and (42) fraud for every signer. The signers have been instructed to sign within six differently sized Surround box towards simulating several, cases. These output signatures have been investigated by 600 dpi. Samples have been tested for a set of data sets and the results shown in this table contain the number of samples in each dataset, the number of samples that have been recognized, the number of unrecognized samples and the percentage. The last dataset got a higher approval rate, and that's why we used it. Table 3.7 shown the results.

| Dataset | Number of Total Samples Per Dataset | Number of Accepted Samples | Number of Rejected Samples | Project Performance % |
|---------|-----------------------------------|-----------------------------|----------------------------|-----------------------|
| NISDCC  | 1898                              | 1734                        | 164                        | 91 %                  |
| NSIGCOMP| 280                               | 261                         | 19                         | 93 %                  |
| CEDAR   | 1683                              | 1569                        | 114                        | 93 %                  |
| UTSIG   | 3078                              | 2993                        | 85                         | 97 %                  |
Table (8), Shows the time taken for each stage of the data set (UTSig):

| Stage       | Time Taken |
|-------------|------------|
| Gray        | 0.00:00.421943 |
| Gamma       | 0.00:04.4845099 |
| Binarization| 0.00:04.4402086 |
| Seed Filling| 0.00:08.604559 |
| Skeleton    | 0.00:08.955009 |
| Clipping    | 0.00:08.5891781 |
| Matching    | 0.00:05.6854264 |

This Table (9) contains the number of false signatures that we applied to the program, and as we can see all of them rejected, this means the program performed 100% because it rejected all forged signatures.

| Dataset     | Number of Tested Forgery Samples | Number of Accepted Samples | Number of Rejected Samples | Project Performance % |
|-------------|----------------------------------|----------------------------|----------------------------|------------------------|
| NISDCC      | 150                              | 0                          | 150                        | 100 %                  |
| NSIGCOMP    | 65                               | 0                          | 65                         | 100 %                  |
| CEDAR       | 312                              | 0                          | 312                        | 100 %                  |
| UTSIG       | 684                              | 0                          | 684                        | 100 %                  |

3.6. Results Comparison

In this segment, the finding here is comparable with results from published studies appeared in Table (10)

| Method                              | Year | 1st author - Ref #                                      | Rate % |
|-------------------------------------|------|--------------------------------------------------------|--------|
| HOG-Deep multitask metric          | 2016 | Soleimani (Soleimani, Araabi, et al., 2016)            | 82.85% |
| Fixed-point arithmetic             | 2016 | Soleimani (Soleimani, Fouladi, et al., 2016b)         | 70.29% |
| Active Transfer Learning           | 2019 | Younesian (Younesian et al., 2019)                    | 83.60% |
| HOC + HOG (linear kernel)          | 2016 | Soleimani (Soleimani, Fouladi, et al., 2016a)        | 83.98% |
| GWBTA (Cylindrical Shape Context)  | 2018 | Narwade (Narwade et al., 2018)                       | 83.82% |

Conclusion

A signature authentication machine is enforced to produce an easy, safe, quick biometric behavioral security system. By exploiting, texture options make this technique quicker than different strategies. The matching technique makes it safer.
The user-interface of this application is incredibly effortless that makes it user-friendly and straightforward. This is one of the principle limitations of the system:

- Our system able to identify the static changes in a signature, it cannot establish any dynamic changes during a signature.
- This signature verification framework works offline. No online device is connected with this system. And the techniques of signature authentication confirmed only offline. Furthermore; the proposed system has the following conclusions:

1- Off-line signature recognition system is implemented with two phases (training and identification) using C# programming language. Enhancement applications are applied to the image containing the signature through pre-processing stage that make the system more reliable in dealing with the problems related to the signature's images such as noises, width of signature lines.

2- The images were enhanced using gamma correction method the optimal value for the proposed system was 0.9.

3- The binarization step was implemented using global thresholding. The optimal value of the threshold was obtained using trial and error method which was 128.

4- The defects in the signature (gabs, holes) due to the variation of the papers and the pens used by the signers was corrected using seed filling algorithm.

5- Clipping algorithm was implemented to reduce the processing time and extract the signature of the user from the background, which helps to obtain the optimal features for the signature itself rather than the entire image.

6- Clipping algorithm (dynamic size) was better than cropping (fixed size) because the signature is different from person to person in size, shape; which leads to better recognition and enhance the accuracy due to its beneficial for compensating the small shifts.

7- The accuracy of the proposed system reached 97% on a Parisian dataset (UTSig).

**Future Work**

In the future work, new directions might be included to the proposed approach as follows:

1. Future advancements for signature acknowledgment can be made by gathering more written by hand datasets from diverse genuine individuals and utilizing them within the preparing and testing the classifier which can lead to upgrade the acknowledgment rate.

2. Utilizing another assortment of features for the matching methodology like invariant moments to check the accuracy of the proposed system.
Reference

I Ahmed, Z. J. (2018). Fingerprints Matching Using the Energy and Low Order Moment of Haar Wavelet Subbands. Journal of Theoretical and Applied Information Technology, 96(18), 6191–6202.

II AL-OBIADIE, S. N. M. (2016). Emotion Detection Using Facial Image Based on Geometric Attributes. University of Baghdad.

III Aldhaher, E., & George, L. (2014). Detection of Diabetic Maculopathy Using Image Analysis Techniques -Introduction and Implementation.

IV Eds, A. D. H. (2018). New Trends in Information and Communications Technology Applications (Vol. 938). https://doi.org/10.1007/978-3-030-01653-1

V Ellen, D., Day, S., & Davies, C. (2018). Scientific examination of documents: methods and techniques. CRC Press.

VI Fadhil, R., & George, L. E. (2017). Finger Vein Identification and Authentication System. LAP Lambert Academic Publishing.

VII Ferrer, M. A., Alonso, J. B., & Travieso, C. M. (2005). Offline geometric parameters for automatic signature verification using fixed-point arithmetic. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(6), 993–997.

VIII George, L. E., Al-Daamy, N., Al-Daamy, S. A., & Ahmed, R. K. (2016). The using of graylevel co-occurrence matrix for features extraction of the breast cancer biopsy image (glcm). Int. J. Engg. Res. and Sci. & Tech, 5(1).

IX Gunjal, S. N., Dange, B. J., & Brahmane, A. V. (2016). Offline Signature Verification using Feature Point Extraction. International Journal of Computer Applications, 975, 8887.

X Hafemann, L. G. (2019). Learning features for Offline Handwritten Signature Verification by MANUSCRIPT-BASED THESIS PRESENTED TO ÉCOLE DE IN PARTIAL FULFILLMENT FOR THE DEGREE OF.

XI Hamza, R. M., & Al-Assadi, T. A. (2012). Genetic algorithm to find optimalGLCM features. Department of Computer Science College of Information Technology.

XII HASSAN, E. K. H., GEORGE, L. E., & MOHAMMED, F. G. (2018). Color image compression based on DCT, differential pulse coding modulation, and adaptive shift coding. Journal of Theoretical and Applied Information Technology, 96(11), 3160–3171.
XIII Inamdar, V. S., Rege, P. P., & Arya, M. S. (2010). Offline Handwritten Signature based Blind Biometric Watermarking and Authetication Technique using Biorhogonal Wavelet Transform. International Journal of Computer Applications, 11(1), 19–27. https://doi.org/10.5120/1547-1970

XIV Jabur, Z. F., & Ali, S. K. (2014). Off line Handwritten Signature Recognition based on Fusion of Global and GLCM Features Using Fuzzy Logic. JOURNAL OF THI-QAR SCIENCE, 4(3), 151–158.

 XV Karouni, A., Daya, B., & Bahalak, S. (2011). Offline signature recognition using neural networks approach. Procedia Computer Science, 3, 155–161.

 XVI Kaur, H., & Kaur, S. (2014). Offline Hindi Signature Recognition Using Surf Feature Extraction and Neural Networks Approach. Ijsr. Net, 3(8), 1141–1146.

 XVII Mahanta, L. B., & Deka, A. (2013). A study on handwritten signature. International Journal of Computer Applications, 79(2).

 XVIII Mohammed, S. N., & George, L. E. (2016). Illumination-Invariant Facial Components Extraction Using Adaptive Contrast Enhancement Methods. Current Journal of Applied Science and Technology, 1–13.

 XIX Narwade, P. N., Sawant, R. R., & Bonde, S. V. (2018). Offline handwritten signature verification using cylindrical shape context. 3D Research, 9(4), 48.

 XX Pirlo, G., Impedovo, D., Fairhurst, M., Pirlo, G., Impedovo, D., & Fairhurst, M. (2014). Advances in digital handwritten signature processing: a human artefact for e-society. World Scientific Publishing Co., Inc.

 XXI Pratt, W. K. (1994). Digital Image Processing. In European Journal of Engineering Education (Vol. 19). https://doi.org/10.1080/03043799408928319

 XXII Radhika, K. S., & Gopika, S. (2015). Online and offline signature verification: A combined approach. Procedia Computer Science, 46, 1593–1600. https://doi.org/10.1016/j.procs.2015.02.089

 XXIII Rashidi, S., Fallah, A., & Towhidkhah, F. (2012). Feature extraction based DCT on dynamic signature verification. Scientia Iranica, 19(6), 1810–1819. https://doi.org/10.1016/j.scient.2012.05.007

 XXIV Shakour, A. A. (2018). Biometric Authentication and Recognition System Using Hand Palm Images. Baghdad University.

 XXV Sigari, M. H., Pourshahabi, M. R., & Pourreza, H. R. (2012). An ensemble classifier approach for static signature verification based on multi-resolution extracted features. International Journal of Signal Processing, Image Processing and Pattern Recognition, 5(1), 21–36.

*Copyright reserved © J. Mech. Cont.& Math. Sci.
Shahad S. Hadi et al*

543
XXVI Sindhu, B., & Jeeva, J. B. (2013). Automated Retinal Vessel Segmentation Using Morphological Operation And Threshold. International Journal of Scientific & Engineering Research, 4(5), 1614–1617. Retrieved from http://www.ijser.org

XXVII Soleimani, A., Araabi, B. N., & Fouladi, K. (2016). Deep Multitask Metric Learning for Offline Signature Verification. Pattern Recognition Letters, 80, 84–90. https://doi.org/10.1016/j.patrec.2016.05.023

XXVIII Soleimani, A., Fouladi, K., & Araabi, B. N. (2016a). Persian offline signature verification based on curvature and gradient histograms. 2016 6th International Conference on Computer and Knowledge Engineering (ICCKE), 147–152. IEEE.

XXIX Soleimani, A., Fouladi, K., & Araabi, B. N. (2016b). UTSig: A Persian offline signature dataset. IET Biometrics, 6(1), 1–8.

XXX Soleimani, A., Fouladi, K., & Araabi, B. N. (2017). UTSig: A Persian offline signature dataset. IET Biometrics, 6(1), 1–8. https://doi.org/10.1049/iet-bmt.2015.0058

XXXI Taylor, J. K., & Cihon, C. (2004). Statistical Techniques for Data Analysis. Retrieved from https://books.google.iq/books?id=yw6JwuAclCUC

XXXII Tuama, S. A., George, L. E., Okelola, M. O., Olabode, E. O., Mbah, E. N., Attah, A. J., … Sudharmaidevi, C. R. (2019). Current Research in Science and Technology Vol. 1. Current Research in Science and Technology Vol. 1, 1–17. https://doi.org/10.9734/bpi/crst/v1

XXXIII Tuama, S., & George, L. (2016). Retina Recognition Based on Texture Analysis: Building a system for individual recognition based on vascilar retina pattern.

XXXIV V.G., Y., & Patil, A. (2014). Offline and Online Signature Verification Systems: a Survey. International Journal of Research in Engineering and Technology, 3(3), 328–332.

XXXV Widiarti, A. R. (2011). Comparing Hilditch, Rosenfeld, Zhang-Suen, and Nagendraprasad-Wang-Gupta Thinning. International Journal of Computer and Information Engineering, 5(6), 563–567.

XXXVI Younesian, T., Masoudnia, S., Hosseini, R., & Araabi, B. N. (2019). Active Transfer Learning for Persian Offline Signature Verification. (February), 234–239. https://doi.org/10.1109/pria.2019.8786013

Copyright reserved © J. Mech. Cont.& Math. Sci.  
Shahad S. Hadi et al

544