An Efficient Offline Signature Verification System using Local Features

Basheer Mohamad Al-Maqaleh
Faculty of Computer Science & Information Systems, Department of Information Technology ,Thamar University, Thamar, Republic of Yemen

Abdulbaset Mohammed Qaid Musleh
Faculty of Computer Science & Information Systems, Department of Computer Science ,Thamar University, Thamar, Republic of Yemen

ABSTRACT

The most common secure personal authentication in biometrics is handwritten signature. It's widely used in many felids as banks , business transactions , and documents which are being authorized via signatures. The main challenging problem in design offline signature verification system is the phase of extracting features that distinguish between forged and genuine signatures. In this paper, a novel feature of extraction method based on static image splitting is proposed. The center of density of the signature image is used for the splitting. In the proposed system, a new feature called Pixel Length (F4)is suggested. This feature is used in combination with other three features: Pixel Density (F1), Cell Angle (F2), and Pixel Angle (F3) which are common features in the offline verification signature process. Euclidean distance measure was used for classification. The proposed system is implemented and tested using GPDS database. The performance of the proposed system is measured and the experimental results show the usefulness and effectiveness of the proposed system.

Keywords

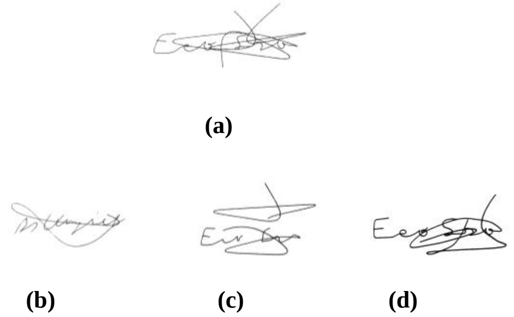
Biometrics, Offline Signature Verification, Feature Extraction, Euclidean Distance Model.

1. INTRODUCTION

A biometric system is used to identify the person through the physiological and behavioral characteristics [1]. The biometrics is primarily used for verification, identification, and watch list [2]. A handwritten signature is widely used in biometric which is the most employed form of secure personal authentication. Thousands of financial and business transactions are being authorized via signatures. Manual verification of signatures of legal licenses and documents is common [3]. So, the verification of signatures within the manual process creating a huge number of documents is considered to be difficult. Based on that, designing an offline handwritten signature verification system is important to differentiate between genuine and forged signatures.

Signature verification contains two areas: offline signature verification and on-line signature verification. Offline signature samples are scanned into image representation by scanners or digital cameras, but on-line signature samples are collected from a digitizing tablet which is capable of pen movements during the writing. Dynamic information like speed, pressure is captured in addition to a static image of signature [4],[5]. Offline handwritten signature verification is used in many domains such as banks, cheque cashing, credit card transactions and other documents. Generally, the offline signature verification system is composed of four stages: capture, preprocessing , features extraction and classification or verification [6],[7]. The offline systems are difficult to design compared to online signature systems because static image does not have many desirable characteristics such as the order of strokes, speed, and other dynamic characteristics [8]. Therefore, the verification process depends only on the features extracted from the static image of signatures [9],[10],[11]. The extract features usually classified into the following types: global features ,local features and transition features [12]. The global features are characterized by much more clarity than width, height and aspect ratio. These features are used in combination with other features in the verification process. Global features are less sensitive to noise and commonly used in signature recognition process [13]. The local features are taken after the division of the image so they are considered to be more efficient than others because they get the smallest details within the image. They are calculated by splitting the signature image into parts with the help of geometric center, density center or some other means. The transition features counts the transition in the signature image from black to white pixel or vice versa in binarized signature images [14].

There are three different types of forgeries as shown in Figure (1). The first, the person who forges another person's signature does not know the shape of the original signature is called random forgery. The second, is called the simple forgery which occurs when the forger person knows the person's signature shape, but has not practiced much on it. The last type, known as skilled forgery which is represented by a reasonable imitation of the genuine signature model [9].



(a)

(b) (c) (d)

Figure 1: Types of forgeries (a) Genuine Signature (b) Random forgery (c) Unskilled forgery (d) Skilled forgery.

The most challenging problem in automatic signature verification is to extract features that discriminate between genuine and forged signatures. The issue of the automated signature verification is to obtain more accurate features that enable us to differentiate between signatures. Therefore, the results of any verification system depend on the algorithm that chooses the features from the signatures. In this work, the proposed system used local features to extract features from signatures images and designed to detect skilled forgeries effectively.

The rest of the paper is organized as follows: Section 2 presents related work. The methodology and implementation are presented in Section 3. Experimental results are provided in Section 4. Conclusions and future directions are given in Section 5.

2. RELATED WORK

Signatures are generally recognized as a legal means of verifying an individual identify by administrative and financial sectors. During the last few years, researchers have made great efforts on signature verification. Two approaches are used to design an offline handwritten verification system. These approaches are writer-dependent (WD) and writer-independent (WI). Handwriting signature verification using neural network is proposed in [15]. They used (WD) method in this model aiming at designing a system for each person having its own signature model and parameters. An offline arabic signature recognition and verification system is presented in [16]. This system is designed for two phases: a recognition phase is dependent on a multistage classifier and a combination of global and local features. The second one, a verification phase, is based on fussy concepts. Offline arabic signature verification using combination of geometrical and grid features is proposed in [17]. In this method four border points are used as geometrical features and the grid features from the core of the signature image. In [7] One Class Support Vector Machine (OC-SVM) based on writer-independent parameters is suggested. Only original signatures (OC-SVM) are taken into consideration in this method because they are effective when plenty of samples of signatures are available reach an accurate classification. An effective method to perform offline signature verification based on intelligent techniques is proposed in [18]. Two neural network based techniques and Support Vector Machines (SVMs) were investigated and compared with the process of signature verification.

A study that compares between a signer dependent and a signer independent for two-class and one-class classification using a variety of classifiers is described in [19]. Handwriting Signature Verification System (HSVS) using different configurations of Local Binary Pattern (LBP) and different classifiers is proposed in [20]. This method aims at measuring the gray level features robustness when it is distorted by a complex background and also attempts to propose more stable features. Arabic and Persian signatures verification based on Discrete Wavelet Transform (DWT) to extract common features to aid the verification step is presented in [21]. Different methods for offline signature verification system that incorporates a novel feature extraction technique is proposed in [22] and [23]. Three new features are extracted from a static image of signatures using this technique and Euclidean distance for classification is used. A new formalism for signature representation based on visual perception is proposed in [24].

The most frequently used feature extraction techniques in the above offline signature verification systems are only tools to detect simple and random forgeries. In the proposed system four robust features are extracted from a static image of signature to detect skilled forgeries effectively. The mean values and stranded deviation of all the original signature features are computed. Euclidian distance in the feature space between the claimed signature and the template serves as a measure of similarity between the two.

3. METHODOLOGY & IMPLEMENTATION

Offline signature verification is a pattern recognition problem and the main stages of the proposed pattern recognition systems as shown in Figure 2. These stages are described below:

3.1 Image Preprocessing Stage

There is a range of essential operations that must be applied to the image signature before starting the extraction of features in order to improve the accuracy of feature extraction and verification. In the proposed system, the following operations are used [25]:

3.1.1 Binarization

Binarization: is a process which converts signature image to binary image. The binary image consists of the writing (black) and the background (white) [2].

3.1.2 Cropper

The required image that contains unwanted space is not used during the verification process. So, it is cropped by using the limits of the image. Therefore, to obtain signature height, the signature image is scanned from top to bottom.

3.1.3 Normalization

The signatures image usually may have different sizes; so, all signatures must have equal sizes to get more reasonable results through the normalization of the length or width of the image [22]. In this study, the sizes used are (150×400) pixel.

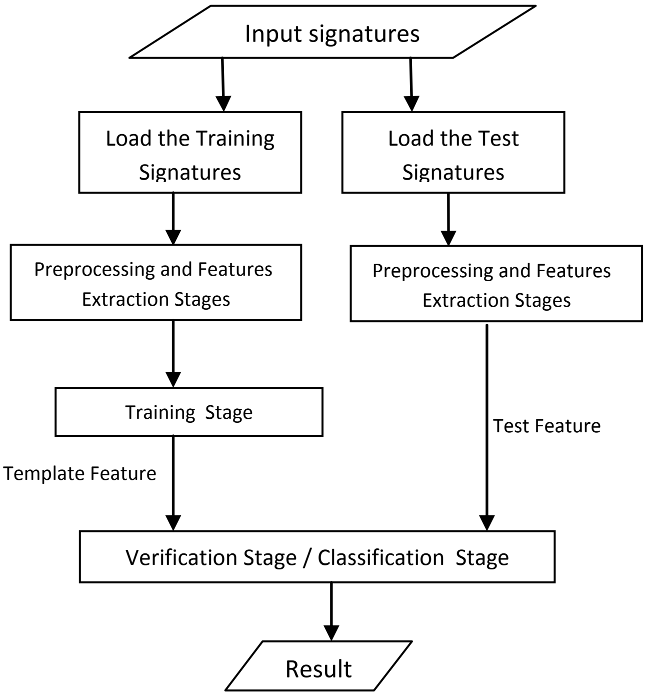

Input signatures
Load the Training Signatures
Load the Test Signatures
Preprocessing and Features Extraction Stages
Preprocessing and Features Extraction Stages
Training Stage
Test Feature
Template Feature
Verification Stage / Classification Stage
Result

Figure 2: Block diagram of the proposed System.

3.1.4 Skeletonization and Thinning

Removing selected foreground pixels from the binary image of the signature is called Skeletonization. A representation of a signature pattern will be the outcome by a group of thin arcs and curves [26]. Eliminating the thickness difference of pen by making the image signature one pixel thick is the goal of thinning [23].

3.2 Feature Extraction Stage

The most challenging problem in automatic signature verification is to extract features that discriminate between genuine and forged signatures. The issue of the automated signature verification is to obtain more accurate features that enable us to differentiate between signatures. Therefore, the results of any verification system depend on the algorithm that chooses the features from the signatures. The feature extraction process is based on signature image splitting. The center of gravity of the signature image is used for splitting. Equation (1) is used to compute center of gravity $(\bar{X}, \bar{Y})$ of the signature image [14]. The signature images are partitioned into rectangle cells up to 64 sub-image cells of moderate resolution.

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n}(X_i) \quad (1)$$

$$\bar{Y} = \frac{1}{n}\sum_{i=1}^{n}(Y_i)$$

where n is a number of white pixels.

The following steps are used to determine the center of gravity:

Step 1. The image is partitioned into sub-image parts as shown in Figure 3.

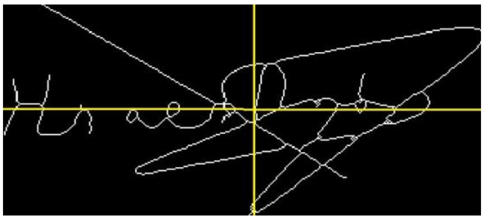


Figure 3: Image is divided into four sub-image parts.

Step 2. Each sub-image is partitioned into four rectangular parts as shown in Figure 4.

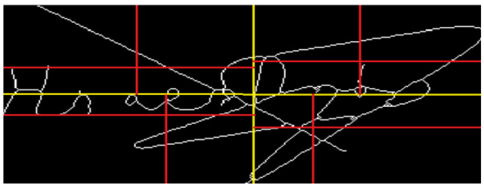


Figure 4: Image is partitioned into 16 sub-image parts.

Note that, all sub-image parts that contain white pixel will be splitted.

Step 3. Partition each of the sub-image parts in Figure 4 into four signature cells. In this step, a set of 64 sub-image cells is produced as shown in Figure 5.

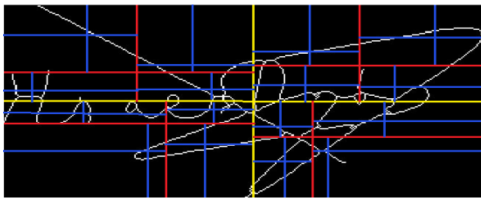


Figure 5: Image is partitioned into 64 sub-image parts.

It is to be noted that, all sub-image parts that contain white pixels will be considered for splitting, and will have the value one, otherwise the value zero is given and these pixels are called black pixels.

In the proposed system, the following features are extracted from pre-processed signature images:

- *Pixel Density (F1)*

i. To calculate the area of each of the 64 sub-image cells.

ii. To calculate the total number of white pixels of each cell.

iii. The Pixel Density (F1) is calculated using equation (2).

$$\text{Pixel Density (F1)} = \frac{\text{Area of each cell}}{\text{Total number of white pixels in each cell}} \quad (2)$$

- *Cell Angle (F2)*

i. Find the pixel density value of cell by using equation (1).

ii. The inclination angle between sub-image center of gravity and the lower right corner of the cell is calculated for each cell.

$$\text{Cell Angle (F2)} = \text{Angle of inclination of center of gravity to the lower right corner} \quad (3)$$

- *Pixel Angle (F3)*

i. To calculate the angle of inclination of each white pixel in each cell to the lower right corner of the cell.

ii. To calculate the sum of angle in each cell.

iii. The Pixel Angel (F3) is calculated using equation(4).

$$\text{Pixel Angle (F3)} = \frac{\text{Angles Sum}}{\text{Total number of white pixels in each cell}} \quad (4)$$

- *Pixel Length (F4)*

i. To calculate the distance between each white pixel in each cell with the lower right corner of the cell.

ii. To calculate the sum of distances in each cell.

iii. The Pixel Length (F4) is calculated using equation(5).

$$\text{Pixel Length (F4)} = \frac{\text{Distances sum}}{\text{Total number of white pixels in each cell}} \quad (5)$$

The above features F1,F2,F3 and F4 are extracted and stored in feature vector. This feature vector is used to train the proposed system as well as for verification of a sample. Each feature vector has 64 values are (f_1, f_2, f_3…………f_{64}).

3.3 Training Stage and Threshold Selection

In this stage a set of reference signature images for each person is used. The proposed system is trained three times with three different numbers of reference signature image (n) 4,8 and 12 respectively .

Given training signature samples are represented as S1,S2,..,Sn for each signature image , Si={F1,F2,F3,F4}.

The corresponding feature vector components of each signature are represented as :

F1=$\{f_{1,1}, f_{1,2} \ldots, f_{1,64}\}$

F2=$\{f_{2,1}, f_{2,2} \ldots, f_{2,64}\}$

F3=$\{f_{3,1}, f_{3,2}, \ldots, f_{3,64}\}$

F4=$\{f_{4,1}, f_{4,2} \ldots, f_{4,64}\}$

Each signature samples are stored in the database as a template. The mean values (F_{mean}) of each corresponding feature vector components are computed using equation (6).

$$F_{mean,j} = mean\left[F_{j,1}, F_{j,2}, F_{j,3}, ...F_{j,n}\right]$$
(6)

where j is number of feature, and n is number of training signatures.

These values constitute the template feature vector. Euclidean distance (d) between the template feature vector (F_{mean}) and the feature vector components (F) for each training sample is calculated using equation (7).

$$d(F_{j,i}) = \sqrt{\sum_{i=1}^{64}(F_{mean,j} - F_{j,i})^2}$$
(7)

where j is number of feature and i is current feature vector components of each signature

Two main parameters are used in threshold calculation are mean(dj) and (σ) standard deviation of training sample equation (8) and equation (9) show the calculation of these two parameters.

$$mean(\mathrm{dj}) = \frac{F_j}{n} \quad (8)$$

$$\sigma = SD(dj) \quad (9)$$

where j is number of feature , and n is number of training signatures.

The threshold value for each feature is given using equation (10).

$$\text{Threshold } \left(F_j\right) = \sqrt{(mean(\mathrm{dj}) + \sigma j)2} \quad (10)$$

3.4 Verification Stage

The last stage is the verification stage , this stage compares the incoming test signatures with the user's signature template in the database. The Euclidean distance model is one of the most suitable classifier used to obtain distance measurement between two vectors of equal size on a two dimensional plane [25]. The proposed system generates four vectors for both user's signature template and the incoming test signature. Each vector refers to one feature . Euclidean distance d(Fj) between the average value of each feature that is calculated by equation (6) in the training phase and each feature vector of the testing signature is computed by equation (11).

$$d\left(F_j\right) = \sqrt{\sum_{i=1}^{64}(F_{j,i} - F_{mean,j})^2} \quad (11)$$

where j is number of feature and i is current feature vector components of each signature

In the proposed system, two features F3 and F4 are used in the verification stage. So , the Euclidean distance d(Fj) of equation (11) and Threshold of equation (10) is compared based on these two features for each user. If d(F3) is less than or equal to Threshold (F3) OR d(F4) is less than or equal to Threshold (F4) then the incoming test signature is accepted otherwise the signature is rejected.

4. EXPERIMENTAL RESULTS

A publicly available GPDS database [27],[28] is used in training and testing the proposed system. This database composes of 24 genuine signatures and 30 simulated forgeries from 4000 individuals. All image signatures, either genuine or forged, in the database are collected from a group of individuals. Each individual singed on a sheet of white A4 paper by ballpoint. Each sheet provided two different box sizes for the signature. The image signatures in the database were scanned at 600 dpi with 256 gray levels. Experiments have been conducted to evaluate the performance of the proposed system for skill forgery signature.

A total number of 3600 gunnies signatures and 9000 forgery signatures made up 7300 signatures for training. Also, 16200 signatures are tested. Four different parameters have been used to measure the performance of the proposed system. These are False Acceptance Rate (FAR) ,False Rejection Rate (FRR), Average Error Rate (AER) and Accuracy.

FAR measures the present of the forgeries signature that are incorrectly classified [29].

FRR measures the present of the originals signature that are incorrectly classified.

The Average Error Rate (AER) is the average of FAR and FRR.

Accuracy measures the presence of signatures which are exactly classified.

The proposed system achieves the best performance when F3 and F4 are joint by logical operator OR. Table 1 and Table 2 summarize the results along with its accuracy for GPDS database. The signature image is partitioned by 64 sub-image parts and with 16 sub-image parts respectively. The proposed system is compared with some of other systems based on an algorithm which is capable of deciding whether to accept or reject the signatures which are under test. AER is used to compare the results as shown in Table 3.

Table 1. The results for GPDS database with 64 sub-image parts.

Features	FAR	FRR	AER	Accuracy
Pixel Density (F1)	20.14	69	29.17	45.77%
Cell Angle (F2)	30.82	28.79	29.80	70.60%
Pixel Angle (F3)	41.63	16.54	29.09	75.87%
Pixel length (F4)	36.92	21.42	44.57	73.90%
(F3) OR (F4)	18.94	8.81	13.87	**88.13%**

Table 2. The results for GPDS database with 16 sub-image parts.

Features	FAR	FRR	AER	Accuracy
Pixel Density (F1)	15.67	70.81	43.24	45.86%
Cell Angle (F2)	27.33	30.88	29.11	69.38%
Pixel Angle (F3)	30.96	25.81	28.39	72.63%
Pixel length (F4)	28.03	31.94	30	70 %
(F3) OR (F4)	13.99	13.08	13.54	**86.64%**

5. CONCLUSION AND FUTURE WORK

An efficient offline verification system is needed to detect all kinds of forgeries particularly in paper documentation environment, like banks, schools and government ministries. The achievement made in this work will go a long way to improve the current situation in this research area. The four new features extracted in this work are robust enough to prevent signature forgeries. The performance of the proposed system is comparable to other offline signature verification systems as indicated by the results. The experimental results have shown the ability of the proposed system against all skilled of forgeries. But there is a need to combine different classifiers with different feature vectors in future work to enhance performance.

Table 3. Comparison results for GPDS 300 users from database with other systems.

References	Classifier	Feature	Signature for training	AER(%)
Kumar et al [30].	Neural network	Surroundedness	24 Genuine + 24 forged	13.76
Batista et al [6].	HMM+SVM	Grid segmentation	4 Genuine 8 Genuine 12 Genuine	20.53 17.24 16.84
Y. Guerbai [7].	OC-SVM	Curve let transform	4 Genuine 8 Genuine 12 Genuine	16.92 15.95 15.07
The proposed system	Euclidean Distance	(Angel and Length) Local features	4 Genuine 8 Genuine 12 Genuine	**18.56** **13.87** **12.53**

6. ACKNOWLEDGMENTS

Thanks are given to all the people who have helped in this research. Acknowledgement is also for the support of Dr . Miguel A. Ferre who kindly provided this research with sample signatures used in this study.

7. REFERENCES

[1] Guerbai, Y., Chibani, Y. and Abbas, N. 2012. One-class versus bi-class SVM classifier for offline signature verification. In Multimedia Computing and Systems, International Conference on IEEE (ICMCS2012), pp. 206-210.

[2] Eliza Y. D. 2013. Biometric From Fictions to Practice . U.S. Government works Version , International Standard Book no.13, pp. 978-981.

[3] Vitthal , K. B. and Anil, R. K. 2013. Automatic static signature verification system. International Journal of Computational Engineering Research, vol.3,Issue.2, pp. 8-12.

[4] Bhattacharya, I., Ghosh, P. and Biswas, S. 2013. Offline signature verification using pixel matching technique. Procedia Technology. vol. 10, pp. 970-977.

[5] Bertolini, D., Oliveira, L. S., Justino, E. and Sabourin, R. 2010. Reducing forgeries in writer-independent offline signature verification through ensemble of classifiers. Pattern Recognition. vol. 43, pp. 387-396.

[6] Batista, L., Granger, E. and Sabourin, R. 2012. Dynamic selection of generative–discriminative ensembles for offline signature verification. Pattern Recognition, vol. 45 , pp.1326-1340.

[7] Guerbai, Y., Chibani, Y. and Hadjadji, B. 2015. The effective use of the one-class SVM classifier for handwritten signature verification based on writer-independent parameters. Pattern Recognition, vol. 48, pp. 103-113.

[8] Huang K., and Hong Y. 1997. Offline signature verification based on geometric feature extraction and neural network classification. Pattern Recognition, vol. 30 , pp. 9-17.

[9] Ghandali, S. and Moghaddam, M. E. 2009. Offline persian signature identification and verification based on image registration and fusion. Journal of Multimedia, vol. 4, pp.137-144.

[10] Baltzakis, H., and Papamarkos, N. 2001. A new signature verification technique based on a two-stage neural network classifier. Engineering Applications of Artificial intelligence, vol. 14, pp.95-103.

[11] Srivastava, S. and Agarwal, S. 2013. Offline signature verification based on pixel oriented and component oriented feature extraction. International Journal of Advanced Research in Computer Science, vol. 4, pp.77-83.

[12] Abuhaiba, I. S. 2007. Offline signature verification using graph matching. Turk J Elec Engin, vol.15, pp. 89-104.

[13] Al-Omari, Y. M., Siti Norul H. S. Abdullah and Omar, K. 2011. State-of-the-art in offline signature verification system. In Pattern Analysis and Intelligent Robotics (ICPAIR2011), International Conference on IEEE , vol. 1, pp. 59-64.

[14] Samuel, D. and Samuel, I. 2010. Novel feature extraction technique for offline signature verification system. International Journal of Engineering Science and Technology, vol. 2, pp. 3137-3143.

[15] Pansare, A. and Bhatia, S.2012. Handwritten signature verification using neural network. International Journal of Applied Information Systems (IJAIS), vol. 1, no. 2, pp.44-49.

[16] Ismail, M. A. and Gad, S. 2000. Offline arabic signature recognition and verification. Pattern Recognition, vol. 33, pp.1727-1740.

[17] Ahmed, S. M. 2012. Offline arabic signature verification using geometrical features. National Workshop on Information Assurance Research, Proceedings of (WIAR2012) ,pp. 1-6.

[18] Nguyen, V., Blumenstein, M., Muthukkumarasamy, V., and Leedham, G. 2007. Offline signature verification using enhanced modified direction features in conjunction with neural classifiers and support vector machines, In Proceedings of 9th International Conference on Document Analysis and Recognition (ICDAR 2007), IEEE Computer Society Washington, USA, vol. 2, pp. 734-738.

[19] Srihari, S. N., Xu, A. and Kalera, M. K. 2004. Learning strategies and classification methods for offline signature verification. In Proceedings of 9th International Workshop on Frontiers in Handwriting Recognition (IWFHR-2004), IEEE Computer Society Washington, DC, USA , pp. 161-166.

[20] Ferrer, M., Vargas, J., Morales, A. and Ordóñez, A. 2012. Robustness of offline signature verification based

on gray level features. IEEE Transactions on Information Forensics and Security, vol. 7, pp. 966-977.

[21] Tan, X., Jaafar, A. A., Yahya, A., Ahmad, R., Zain, A., Salman, M. and Linlin, W. 2013. Offline signature verification system based on DWT and common features extraction. Journal of Theoretical and Applied Information Technology, vol. 51, pp. 165-174.

[22] Hetal V. Davda. and S. K. G. 2014. Offline signature verification system using energy on grid level , International Journal of Engineering Research, vol. 3 , pp. 104-107.

[23] Htight ,W. H. and Soe, A.L. 2014. Offline signature verification system using neural network, International Conference on Advances in Engineering and Technology, vol. 1, pp.302-306.

[24] Sabourin, R., Genest, G. and Prêteux, F. J. 1997. Offline signature verification by local granulometric size distributions. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19,Issue 9, pp. 976-988.

[25] Majhi, B., Reddy, Y. S. and Babu, D. P. 2006. Novel features for offline signature verification. International Journal of Computers, Communications & Control, vol.1, pp. 17-24.

[26] Ramachandra, A. C., Rao, J. S., Raja, K. B., Venugopla, K. R., and Patnaik, L. M. 2009. Robust offline signature verification based on global features. In Advance Computing Conference, IEEE International (IACC 2009), pp. 1173-1178.

[27] Vargas, J. F., Ferrer, M., Travieso, C. M. and Alonso, J. B. 2007. Offline handwritten signature GPDS-960 corpus, International Conference on Document Analysis and Recognition (ICDAR), vol. 2, pp. 764-768.

[28] Ferrer, M., Díaz-Cabrera, M. and Morales, A. 2013. Synthetic offline signature image generation. In Proceedings of the 9th International Conference on Biometrics (ICB2013) , IEEE , pp. 1-7.

[29] R. Verma and Rao, D.S. 2013. Offline signature verification and identification using angle feature and pixel density feature and both method together. International Journal of Soft Computing and Engineering (IJSCE), vol. 2, Issue 4, pp.740-746.

[30] Kumar, R., Sharma, J. D. and Chanda, B. 2012. Writer independent offline signature verification using surroundedness feature. Pattern Recognition Letters, vol.33, pp. 301-308.