AUTOMATIC WRITER VERIFICATION ALGORITHM FOR CHINESE CHARACTERS USING SEMI-GLOBAL FEATURES AND ADAPTIVE CLASSIFIER

ABSTRACT

Writer verification is to identify whether the script was written by a person himself. However, although there are many advanced machine learning methods now, automatic writer verification is still a very challenging work since the training data in the false case (forged case) is usually very hard to acquire. In practice, to avoid being convicted, a criminal may write scripts that are fully different from what he wrote on the forged document. In this manuscript, we adopt global and semi-global information, include log-Gabor features, advanced moments features, and co-occurrence matrix features for writer verification. In addition, a more flexible classifier is proposed. Though the convolutional neural network is popular, its performance is limited when the training data is not enough. Therefore, another classifier based on the weighted squared Euclidean distance is adopted. Simulations show that the proposed algorithm outperforms other methods and will be very helpful for identifying forged scripts.

.

Index Terms— Handwriting; feature extraction; end-point information; forensic image processing; pattern recognition

1. INTRODUCTION

In this paper, we propose an advanced feature extraction and classification methods for writer verification. Writer verification is to identify whether the script was written by the litigant himself or the script is forged. It is important for criminal investigation, forensics, signature verification, and security authentication. It is different from character recognition. Character recognition aims to recognize what the character is while writer identification tries to identify whom the handwriting is written by.

Writer verification is one of the pattern recognition and classification problems and the machine learning techniques can be applied. However, compared to other classification problems, writer verification is even more challenging. To apply machine learning, especially the convolutional neural network (CNN) based method, there should be enough number of training data for both the positive and the negative cases. However, for the writer verification problem, in practice, the training data in the false case is very hard to acquire. To avoid being accused, a suspect may write scripts that are fully different from what he wrote on the forged document. Therefore, one should apply a robust writer verification method that has good performance even if the number of training data is insufficient. To achieve it, an advanced algorithm for feature extraction and selection is required.

There are some ways to extract the features of a character. For example, in [1], they used the 2-D Gabor wavelet to extract the features of characters. The Gabor wavelet can extract the location and direction information of edges and are helpful for identifying the structure of a character. In [2], Bensefia proposed a concept of writer's invariance. In [3], Berkay came up with the idea of applying the histogram oriented gradient (HoG) and the local binary pattern (LBP). The HoG reflects the gradient orientations of grid zones and the LBP reflects the occurrence of binary patterns. In [4], Jain introduced the K-adjacent segment (KAS) features to represent the relationship between sets of neighboring edges and has the ability to capture discriminative local stroke information. In [5, 6], the scaled invariant feature transform (SIFT), which is invariant to scaling and rotation, is applied to extract the features of characters.

After extracting the features, a classifier is applied to conclude whether the script is forged. For example, the weighted Euclidean, Manhattan, or Chi-square distances [6, 7], the *K*-nearest neighbor (KNN) [1], the hidden Markov model [8], the vector quantization [5, 6, 9], and the support vector machine (SVM) [3, 5, 10] can all be used as the classifier for writer verification.

2. PROPOSED ALGORITHM

As shown in Fig. 1, the proposed method can be separated into 2 parts: feature extraction and classification. For feature extraction, 3 types of features are combined. The first type is log-Gabor features because they represent the spatial-frequency response of visual neurons more properly than Gabor features due to the symmetry on the log-axis. The second type is advanced moment features including Hu moment invariants, affine moment invariants, and Tsirikolias-Mertzios moments, which are robust to scale,

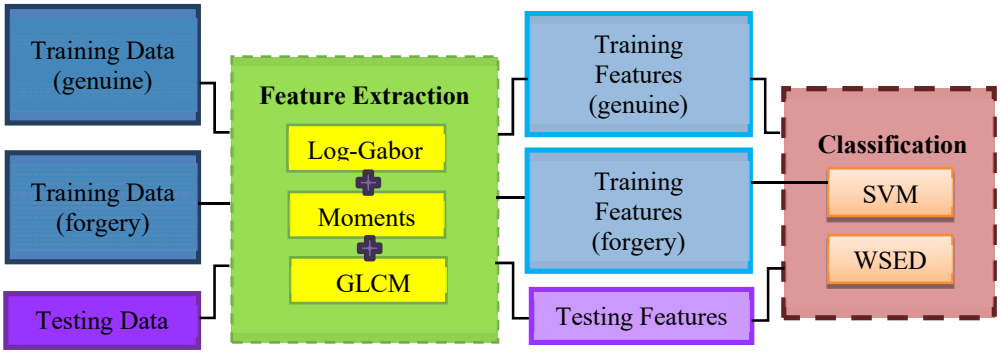

Fig. 1. The framework of the proposed method

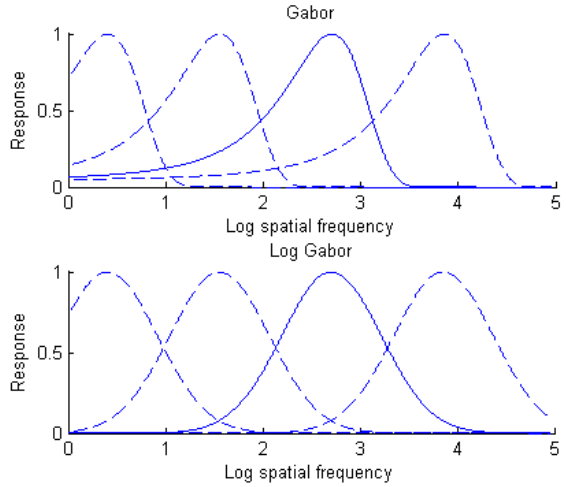

Fig. 2. The Gabor function and the log-Gabor function in the logarithmic spatial frequency.

translation, and rotation. The third type is the features generated from gray level co-occurrence matrices (GLCM), which is related to the information of texture homogeneity, contrast, entropy, and correlation.

Moreover, an adaptive classification is applied. The convolutional neural network (CNN) is more suitable for the condition where the data should be sufficient for both genuine and forgery cases. However, in real-life applications of writer verification, such as forensic justification, the amount of data that can be collected is less, especially for the forgery case. It is hard to ask a suspect to faithfully write the forgery scripts similar to that he wrote in the document.

Therefore, another classification method is designed to use only genuine training data as references based on the weighted squared Euclidean distance.

2.1. Log-Gabor Features

The log-Gabor function is a modification version of the Gabor function. The Gabor function represents a minimum in terms of the spread of uncertainty in space and spatial frequency. However, its mathematical property is only pure in Cartesian coordinates where the channels are the same size respectively in frequency and space. The relative spread and overlap of neighboring units would be altered if changed to polar distribution. Log-Gabor function can restore some destructive effects of such polar mapping with frequency response

$$G(f) = \exp\left(-\left(\log(\frac{f}{f_0}) \right)^2 \Big/ 2\left(\log(\frac{\sigma}{f_0}) \right)^2 \right) \qquad (1)$$

where f_0 is the central frequency and σ can control the decay rate. A very important property of the log-Gabor function is that its frequency response is symmetric on a log axis, which is the standard method for representing the spatial-frequency response of visual neurons. Though not the best-fitting function, log-Gabor function is a better model for suiting the visual system than Gabor function.

Another advantage of log-Gabor is that its bandwidth increases with frequency, meaning that the bandwidths are constant in octaves. As displayed in Fig. 2, the log-Gabor function spreads the information equally in each channel while Gabor function over-represents the low frequencies.

In the proposed system, the log-Gabor filter bank is designed in 4 scales and 6 orientations. The minimum wavelength of scale filter is 3. The scaling factor between successive filters is set to be 1.7 while the ratio of the standard deviation of the Gaussian describing the transfer function in frequency domain to the filter center frequency is set to be 0.65. The combination makes the bandwidth in the range of 1-2 octaves.

2.2. Moment Features

In the proposed system, Hu moments, affine invariant moments, and Tsirikolias-Mertzios moments are adopted. These moments can be derived from the conventional moments derived as follows. For a two-dimensional (2-D) image, the conventional moment of order $(p+q)$ is:

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q I(x,y), \quad p,q = 0,1,2,...,\infty . \qquad (2)$$

The central moment of an image is computed as

$$\mu_{pq} = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(x-\bar{x})^p (y-\bar{y})^q I(x,y) \quad (3)$$

where $\bar{x} = m_{10}/m_{00}, \quad \bar{y} = m_{01}/m_{00}$. (4)

They are invariant to translation and can be normalized as:

$$\eta_{pq} = \mu_{pq} / \mu_{00}^{\gamma} \quad (5)$$

where $\gamma = (p+q)/2+1, \quad p+q = 2,3,\ldots.$

Hu moments are invariant under scale, translation and rotation, which can be derived from the 2nd and the 3rd order conventional moments.

$$\begin{aligned}
&\phi_1 = \eta_{20} + \eta_{02}, \qquad \phi_2 = (\eta_{20}-\eta_{02})^2 + 4\eta_{11}^2,\\
&\phi_3 = (\eta_{30}-3\eta_{12})^2 + (3\eta_{21}-\eta_{03})^2,\\
&\phi_4 = (\eta_{30}+3\eta_{12})^2 + (\eta_{21}+\eta_{03})^2\\
&\phi_5 = (\eta_{30}-3\eta_{12})(\eta_{30}+\eta_{12})\left[(\eta_{30}+\eta_{12})^2 - 3(\eta_{21}+\eta_{03})^2\right]\\
&\quad + (3\eta_{21}-\eta_{03})(\eta_{21}+\eta_{03})\left[3(\eta_{30}+\eta_{12})^2 - (\eta_{21}+\eta_{03})^2\right],\\
&\phi_6 = (\eta_{20}-\eta_{02})\left[(\eta_{30}+\eta_{12})^2 - (\eta_{21}+\eta_{03})^2\right]\\
&\quad + 4\eta_{11}(\eta_{30}+\eta_{12})(\eta_{21}+\eta_{03}),\\
&\phi_7 = (3\eta_{21}-\eta_{30})(\eta_{30}+\eta_{12})\left[(\eta_{30}+\eta_{12})^2 - 3(\eta_{21}+\eta_{03})^2\right]\\
&\quad + (3\eta_{12}-\eta_{30})(\eta_{21}+\eta_{03})\left[3(\eta_{30}+\eta_{12})^2 - (\eta_{21}+\eta_{03})^2\right].
\end{aligned} \quad (6)$$

Affine invariant moments are robust to any affine transformation in the 2-D plane. Its explicit definition is shown in [7]. Tsirikolias-Mertzios moments are normalized by centroids and standard deviations, which are defined as:

$$m_k = \frac{1}{N}\sum_{j=1}^{N}\left[\frac{x_j - mean(x)}{\sigma_x}\right]^k, \quad k = 1,2,3... \quad (7)$$

where σ_x is the standard deviation of x:

$$\sigma = sqr(\mathrm{VAR}) = sqr\left[\frac{1}{N-1}\sum_{j=1}^{N}\left(x_j - \bar{x}\right)^2\right]. \quad (8)$$

One can generalize the moments in (7) into the 2D case:

$$m_{pq} = \frac{1}{LM}\sum_{x=1}^{L}\sum_{y=1}^{M}\left[\frac{x-\bar{x}}{\sigma_x}\right]^p\left[\frac{y-\bar{y}}{\sigma_y}\right]^q I(x,y) \quad (9)$$

where $L \times M$ is the size of the image $I(x, y)$, $(\bar{x}, \bar{y})$ is the centroid, and σ_x and σ_y are the standard deviation of the image along x and y directions, respectively. The moments $m_{30}, m_{40}, m_{50}, m_{60}, m_{70}, m_{80}$ and $m_{03}, m_{04}, m_{05}, m_{06}, m_{07}, m_{08}$ are used in the proposed writer verification system. These moments are invariant under translation and magnification but not under rotation.

2.3. Gray Level Co-occurrence Matrix Features

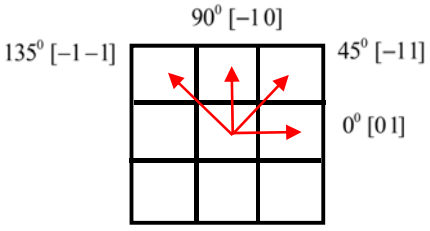

Fig. 3. The relationships between pixel of interest and its neighbors specified in both distance and angle.

The gray level co-occurrence matrix (GLCM), also called gray-tone spatial-dependence matrix [11, 12]. It is a method that extracts 2nd order statistical texture features because it records the value changes between neighboring pixels. For an image $I(x, y)$ with width W_x and height H_y, the gray tone is quantized to N_g levels. Let $L_x = \{1, 2, \ldots, W_x\}$, $L_y = \{1, 2, \ldots, H_y\}$ be the horizontal and the vertical spatial domain, respectively, and $G = \{1, 2, \ldots, N_g\}$ be the quantized gray tones. The image can be mapped from $H_y \times W_x \to G^2$ with the function defined as calculating the frequencies of gray-tone transfers between adjacent cells.

As shown in Fig. 3, There are 4 matrices can be generated by accumulating the neighboring gray-tone relations along the 4 directions. Their formulas are

$$\begin{aligned}
P(i,j,d,0^\circ) &= \#\{((k,l),(m,n)) \in (L_y \times L_x)\times(L_y \times L_x) \mid \\
&\quad m-k = 0, n-l = d, I(k,l) = i, I(m,n) = j\}\\
P(i,j,d,45^\circ) &= \#\{((k,l),(m,n)) \in (L_y \times L_x)\times(L_y \times L_x) \mid \\
&\quad m-k = -d, n-l = d, I(k,l) = i, I(m,n) = j\}\\
P(i,j,d,90^\circ) &= \#\{((k,l),(m,n)) \in (L_y \times L_x)\times(L_y \times L_x) \mid \\
&\quad m-k = -d, n-l = 0, I(k,l) = i, I(m,n) = j\}\\
P(i,j,d,135^\circ) &= \#\{((k,l),(m,n)) \in (L_y \times L_x)\times(L_y \times L_x) \mid \\
&\quad m-k = -d, n-l = -d, I(k,l) = i, I(m,n) = j\}
\end{aligned} \quad (10)$$

where # denotes the number of elements in the set. Though the original GLCM proposed by Haralick was symmetric, we implemented the non-symmetric form here.

In our implementation, G is set to be 8 and only values in the range of [50, 254] are equally quantized. So there are 4 gray level co-occurrence matrices generated for each input image. Since there are 4 directions of GLCM and 4 kinds of measures, we have 16 textural measures in total.

As suggested in [11], several classical feature measures can be extracted from the GLCM, including

(i) Texture homogeneity

$$H = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1}\left(P(i,j)\right)^2. \quad (11)$$

(ii) Texture contrast

$$C = \sum_{n=0}^{G-1}\left(\sum_{i=0}^{G-1}\sum_{j=i\pm n} n^2 P(i,j) \right) . \quad (12)$$

(iii) Texture entropy

$$E = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1} P(i,j)\log\big(P(i,j)\big) . \quad (13)$$

(iv) Texture correlation.

$$O = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1} \frac{ijP(i,j)-(m_i m_j)}{\sigma_i \sigma_j} \quad (14)$$

where m_i and m_j are the means and σ_i and σ_j are the standard deviations of the i^{th} row and the j^{th} column, respectively.

2.4. Adaptive Method for Classification

There are two choices for classification, SVM and the weighted squared Euclidean distance (WSED). The SVM has good performance if the training data is sufficient and balanced. However, in real-life applications of writer verification, the training data (or reference handwritings) are usually not enough and confined to the genuine cases since it is difficult to collect possible forgeries. So we apply the classifier of the WSED in our system.

In the method of the WSED, only genuine training data are required as reference handwritings for verification. The threshold for determination depends only on the distance values of genuine training data. We proposed a weighted version based on the observation of feature importance. Let the feature vector for each image c be

$$f^{(c)} = [\underset{i=1,\ldots,48}{f^{(c)}_{\log Gabor,i}}, \underset{j=1,\ldots,23}{f^{(c)}_{moment,j}}, \underset{k=1,\ldots,8}{f^{(c)}_{\text{GLCM,k}}}] . \quad (15)$$

Then, for each feature type, we calculate the following distances for genuine training data

$$\begin{aligned} d(m,n)_{\log Gabor} &= \sum_{i=1}^{48}(f^{(m)}_{\log Gabor,i} - f^{(n)}_{\log Gabor,i})^2 \\ d(m,n)_{moment} &= \sum_{j=1}^{23}(f^{(m)}_{\text{moment,j}} - f^{(n)}_{\text{moment,j}})^2 \\ d(m,n)_{\text{GLCM}} &= \sum_{k=1}^{8}(f^{(m)}_{\text{GLCM,k}} - f^{(n)}_{\text{GLCM,k}})^2 \end{aligned} \quad (16)$$

where $m, n \in$ {genuine training data}. Based on the idea that a good feature should vary more for different labels and vary less when the labels are the same, the weighting for each type of feature is defined as:

$$\begin{aligned} w_{\log Gabor} &= \frac{1/std(\text{d}_{\log Gabor})}{\frac{1}{std(\text{d}_{\log Gabor})}+\frac{1}{std(\text{d}_{moment})}+\frac{1}{std(\text{d}_{GLCM})}} \\ w_{moment} &= \frac{1/std(\text{d}_{moment})}{\frac{1}{std(\text{d}_{\log Gabor})}+\frac{1}{std(\text{d}_{moment})}+\frac{1}{std(\text{d}_{GLCM})}} \\ w_{GLCM} &= \frac{1/std(\text{d}_{GLCM})}{\frac{1}{std(\text{d}_{\log Gabor})}+\frac{1}{std(\text{d}_{moment})}+\frac{1}{std(\text{d}_{GLCM})}} \end{aligned} . \quad (17)$$

Then, the total WSED is evaluated as

$$d'(m,n) = \sum_{i=1}^{48} \text{w}_{\log Gabor}(f^{(m)}_{\log Gabor,i} - f^{(n)}_{\log Gabor,i})^2 + \sum_{j=1}^{23} \text{w}_{moment}(f^{(m)}_{\text{moment,j}} - f^{(n)}_{\text{moment,j}})^2 + \sum_{k=1}^{8} \text{w}_{GLCM}(f^{(m)}_{\text{GLCM,k}} - f^{(n)}_{\text{GLCM,k}})^2 \quad (18)$$

and threshold for determination T is

$$T = mean(d') + 2std(d') . \quad (19)$$

Then, for every testing image u, the score is calculated as the average of the WSEDs between it and every genuine training image n:

$$score_u = mean(d'(u,n)) . \quad (20)$$

If the score is no larger than T, it is verified as genuine; otherwise forgery.

$$\begin{cases} L_{predict,u} = 1 \text{ (true)}, & if\ score_u \le T \\ L_{predict,u} = -1 \text{ (false)}, & otherwise \end{cases} \quad (21)$$

3. SIMULATIONS

In this study, a Chinese handwriting database for writer verification is required. However, there is no public writer verification database for Chinese scripts until now. Therefore, a Chinese handwriting database based on forgery for writer verification is constructed here.

The database was constructed by 10 volunteers. They were asked to be forgery targets and to write down the assigned contents for 50 times each. The assigned content consists of Chinese characters: 6 characters are the names (the name of a person repeated twice), four characters are the numbers written in Chinese, and two characters are complicated Chinese characters for observing how character complexity affects our results. Then, the forgeries were produced by making other 25 individuals to mimic the assigned contents twice.

For each target set, we got 50 pieces of genuine data along with 25×2 = 50 forgeries. Since our verification system is character-wise, there are (50+50)×12 = 1200 characters per target and 1200 × 10 = 12000 characters in sum. Totally, there are 10 + 25 = 35 writers involved.

Table 1. Comparison of the performances for genuine and forgery script determination.

Method	FAR (%)	FRR(%)	P(%)	R(%)	Accuracy (%)
Proposed	**11.15**	**3.48**	**90.16**	**96.52**	**92.68**
SFH [13]	20.28	19.57	80.59	80.43	80.07
CFC [14]	25.31	20.83	76.35	79.17	76.93
Gabor [15]	14.45	6.01	87.27	93.99	89.77
LBP [16]	14.95	8.53	86.50	91.47	88.26
LDP [16]	15.88	9.78	85.62	90.22	87.17
LDerivP [16]	13.87	8.13	87.37	91.87	89.00
Statistics [17]	21.58	7.76	82.19	92.24	85.33
DoG+SIFT+ loc [6]	17.59	7.75	84.51	92.25	87.33

The complete results are listed in Table 1, which records the false acceptance rate (FAR), false rejection rate (FRR), precision (Precision), recall (Recall) and accuracy (Accuracy) of each method. Table 1 shows that the proposed algorithm has smaller FAR and FRR and higher precision, recall, and accuracy than those of other methods, which prove that the proposed algorithm has high ability to distinguish whether a Chinese script is genuine or forgery.

Simulation results show that our proposed method with SVM classification can reach an accuracy of 92.68% on average provided 50 training images (25 genuine +25 forgery), which towers other methods including Gabor features, the local binary pattern (LBP) [16], the local directional pattern (LDP) [16], the local derived pattern (LDerivP) [16], the stroke fragment histogram (SFH) [13], the curve fragment code (CFC) [14], and the method where the difference of Gaussian (DoG) and the SIFT are applied.

To testify our adaptive WSED classification, the experiment is set up differently here. For each character, the training data is not 50 pieces of handwriting (25 genuine + 25 forgery) anymore. Though the testing data still remains 50 pieces (25 genuine + 25 forgery) for fair comparison, several combinations of training data would be examined. These combinations are set to be unbalanced and would decrease in amount. To sustain the reliability, every character still goes through 5 times of verification processes and then the average accuracy is obtained.

There are 5 kinds of combinations to be tested: 25 genuine +12 forgery, 12 genuine + 6 forgery, 8 genuine + 4 forgery, 6 genuine + 3 forgery and 4 genuine + 2 forgery. Note that in each combination, though some forgery data is provided, the adaptive WSED method only requires the genuine training data. On the contrary, the typical SVM method requires both genuine and forgery training data for classification.

Table 2. Comparing the performances of the SVM and the proposed classifier of the WSED when the amount of forgery training data is less.

Combination	P(%)		R(%)		Accuracy(%)	
Method	**SVM**	**WSED**	**SVM**	**WSED**	**SVM**	**WSED**
24 genuine +12 forgery	83.88	78.91	98.15	97.41	88.94	83.74
12 genuine + 6 forgery	78.39	80.20	96.85	94.88	84.05	83.69
6 genuine + 3 forgery	72.82	83.32	95.77	87.73	78.48	83.18
12 genuine+ 4 forgery	74.22	80.38	97.64	94.20	80.46	83.80
12 genuine + 2 forgery	68.12	81.49	98.74	93.71	74.61	84.40

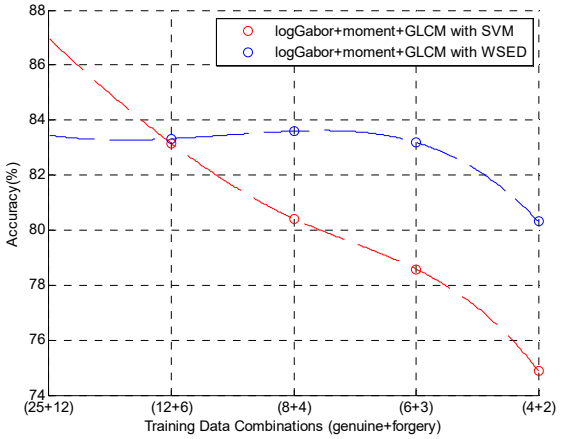

Fig. 4. The verification accuracy of different combinations. The curve is generated by cubic spline interpolation from the accuracy provided by Table 2.

Fig. 4 shows the accuracy for each combination and Table 2 compares the statistical results with SVM and WSED classification.

From Fig. 4, it is observed that WSED classification lead to a 4% lower accuracy at first when the training data is 25 genuine and 12 forgery. However, as the size of the training data decreases, the accuracy of SVM also decreases prominently. When the training data is reduced to 6 genuine and 3 forgery, the accuracy of SVM already drops about 8% while the accuracy of WSED only drops 0.2%. Besides, for the combination of 4 genuine and 2 forgery, FAR of SVM increases to 44.35% while WSED separates some of its increases to FRR, which is more balanced for FP/FN. Thus, we can prove from the simulations that the adaptive WSED classification is applicable for our proposed system when the training data is inadequate or in short of forgery references.

4. CONCLUSION

In this work, a writer verification system based on global features and adaptive classification method is proposed. The framework can be decomposed into feature extraction and classification. For feature extraction, log-Gabor features, advanced moments and features from gray level co-occurrence matrices (GLCM) are combined. Log-Gabor features analyze the textures of the handwriting images from different scales and orientations. Advanced moments include Hu, affine and Tsirikolias-Mertzios moments provide invariance properties under translation, rotation or scaling. Homogeneity, contrast, entropy and correlation calculated from GLCM are also helpful measures. The combination makes the feature vector representing each image discriminative and thus appropriate for writer verification. For classification, the CNN and the SVM outperform other techniques when the training data is adequate and stable. However, an adaptive classification based on the weighted squared Euclidean distance measure is proposed for real applications. This classification method requires no forgery training data and can provide a more robust result when the training data is insufficient. When it is reduced to only 4 pieces of genuine training data, the system could still keep an average accuracy of 80.31%.

REFERENCES

[1] C. Shen, X. G. Ruan, and T. L. Mao, "Writer identification using Gabor wavelet," in *IEEE World Congress on Intelligent Control and Automation*, vol. 3, pp. 2061-2064, 2002.

[2] A. Bensefia, A. Nosary, T. Paquet, and L. Heutte, "Writer identification by writer's invariants," in *Int. Workshop on Frontiers in Handwriting Recognition*, pp. 274-279, 2002.

[3] M. B. Yilmaz, B. Yanikoglu, C. Tirkaz, and A. Kholmatov, "Offline signature verification using classifier combination of HOG and LBP features," in *IEEE Int. Joint Conf. Biometrics*, pp. 1-7, Oct. 2011.

[4] R. Jain and D. Doermann. "Offline writer identification using K-adjacent segments," in *IEEE Int. Conf. Document Analysis and Recognition*, pp. 769-773, 2011.

[5] M. Sreeraj and S. M. Idicula. "The effect of SIFT features as content descriptors in the context of automatic writer identification in Malayalam language," in *IEEE Int. Congress Ultra Modern Telecommunications and Control Systems and Workshops*, pp. 613-617, 2012.

[6] X. Wu, Y. Tang, and W. Bu, "Offline text-independent writer identification based on scale invariant feature transform," *IEEE Trans. Inf. Foren. Sec.*, vol. 9, issue 3, pp. 526-536, 2014.

[7] Z. Y. He and Y. Y. Tang. "Chinese handwriting-based writer identification by texture analysis," in *IEEE Int. Conf. Machine Learning and Cybernetics*, vol. 6. pp. 3488-3491, 2004.

[8] J. Coetzer, B. M. Herbst, and J. A. du Preez. "Offline signature verification using the discrete radon transform and a hidden Markov model," *EURASIP J. Applied Signal Processing*, vol. 4, pp. 559-571, 2004.

[9] S. Fiel and R. Sablatnig, "Writer identification and writer retrieval using the fisher vector on visual vocabularies," in *IEEE Int. Conf. Document Analysis and Recognition*, pp. 545-549, 2013.

[10] Z. Yin, P. Yin, F. Sun, and H. Wu, "A writer recognition approach based on SVM," in *IEEE IMACS Multi-conference on Computational Engineering in Systems Applications*, vol. 1, pp. 581-586, 2006.

[11] J. F. Vargas, M. A. Ferrer, C. M. Travieso, and J. B. Alonso, "Off-line signature verification based on grey level information using texture features," *Pattern Recognition*, vol. 44, issue 2, pp. 375-385, 2011.

[12] R. M. Haralick, "Statistical and structural approaches to texture," *Proceedings of the IEEE*, vol. 67, issue 5, pp. 786-804, 1979.

[13] Y. Tang, X. Wu, and W. Bu, "Offline text-independent writer identification using stroke fragment and contour based features," *IEEE Int. Conf. Biometrics,* pp. 1-6, 2013.

[14] G. Ghiasi and R. Safabakhsh. "Offline text-independent writer identification using codebook and efficient code extraction methods," *Image and Vision Computing*, vol. 31, issue 5, pp. 379-391, 2013.

[15] M. H. Sigari, M. R. Pourshahabi, and H. R. Pourreza. "Offline handwritten signature identification and verification using multi-resolution Gabor wavelet," *Int. J. Biometric and Bioinformatics*, vol. 5, pp. 234-248, 2011.

[16] M. A. Ferrer, J. F. Vargas, A. Morales, and A. Ordóñez, "Robustness of offline signature verification based on gray level features," *IEEE Trans. Inf. Foren. Sec.*, vol. 7, issue 3, pp. 966-977, 2012.

[17] Y. C. Chim, A. A. Kassim, and Y. Ibrahim. "Character recognition using statistical moments," *Image and Vision Computing*, vol. 17, issue 3, pp. 299-307, 1999.