MVA2011 IAPR Conference on Machine Vision Applications, June 13-15, 2011, Nara, JAPAN

Off-Line Signature Verification Using Two Step Transitional Features

E. N. Zois, A. Nassiopoulos
R.D.T.L, T.E.I of Athens
Ag. Spiridonos Str, Greece
{ezois, nassiop}@teiath.gr

K. Tselios, E. Siores
CMRI, University of Bolton,
Bolton BL3 5AB, UK
kt1cmr,E.Siores@bolton.ac.uk

G. Economou
Electron. Lab, Physics Dept.
University of Patras, Greece
economou@upatras.gr

Abstract

In this work, a new approach for off-line signature recognition and verification is presented and described. A subset of the line, concave and convex family of curvature features is used to represent the signatures. Two major constraints are applied to the feature extraction algorithm in order to model the two step transitional probabilities of the signature pixels. Segmentation of the signature trace is enabled using a window which is centred upon the centre of mass of the thinned image. Partitioning of the image leads to a multidimensional feature vector which provides useful spatial details of the acquired handwritten image. The classification protocol followed in this work relies on a hard margin support vector machine. Our method was applied to two databases, the first taken from the literature while the second created by the authors. In order to provide comparable results for the first stage signature verification system, we have applied an already published feature extraction method while keeping the same classification protocol. Primary evaluation schemes on both corpuses provide very encouraging verification results for the Average Error.

1. Introduction

Biometrics, which refers to a person's identity determination, becomes more and more a key aspect of today's security applications. Recently, biometry has noteworthy improvement in efficiency and reliability [1], especially with the evolvement of Internet and dedicated hardware. Among the various types of biometrics, handwriting can be used to formulate one, two or many class pattern recognition problems by using handwritten signatures and/or words-text in common and secure transactions. Signature verification systems can be categorized as on-line or off-line according to the acquisition instrument that is used in order to capture the handwritten sample [1]. For offline systems, image processing techniques in conjunction with pattern recognition algorithms are applied in order to address the various stages of the signature verification system [2].

A short literature review reveals that recent trends in feature selection for offline signature verification are based on grey level information and supplementary texture gray level information [3]. Another approach considers curvature of the most important segments and introduces a graphometric feature set [4]. Contour features have been used also to code and represent the directional properties of the signature contours [5]. Another interesting issue is that feature used in the analysis of writer verification and identification tasks could be employed in order to examine the signature image as a textural signal. Then, textural features could be used in order to represent the feature space [6]. A similar approach codes the probabilistic transitions of signature pixels inside a window, while it explores the relation between the local handwriting strokes and the global shape of the signature [7]. Structural approaches can be also applied by using Chain Code based features [6]. In recent works, texture features have been used in order to supply additional insight to the signature image or handwriting patterns [8]

In this paper we propose an approach for verifying off-line handwritten signatures by means of exploiting the relative pixel distribution along predetermined paths. For this purpose, we have used a subset of the well described curvature features. For each signature pixel inside the segmented and partitioned sub-images statistics are calculated for 15 curvature and line paths inside a confined chessboard distance of two. In order to obtain comparable results, the method is applied to an already used signature database [9] along with a new one which has been created in our institution for verification purposes. Support Vector Machine classifiers (SVM) have been enabled in order to design and realize the classification stage. This paper examines the random forgery case, which could be found appropriate to be applied to a first stage signature verification system. In order to provide comparable results we have chosen to compare our approach with an already published one [3]. The Average Error (AvE) is computed in various SVM configurations by means of measuring the false acceptance rate (FAR) and the false rejection rate (FRR). Finally, it is shown that our proposal provide better results in terms of efficiency.

This paper is organized as follows: Section 2 presents the two databases along with the pre-processing steps that have been employed. Section 3 describes the feature vector extraction method, in order to produce the final feature vector. Section 4 discusses the designing, training and evaluation phases of the classifier. In section 5 the experimental results are reviewed, evaluated, compared, and discussed. Finally, in section 6 the conclusions are drawn.

2. The Database

2.1. Database formation

In order to evaluate our approach, we have used two signature corpuses. The first one (designated hereafter as CORPUS1) is a signature database which is available from the Internet [9]. It is composed of 20 sets of different signatures. Each set is constituted of 24 genuine and 24 forgery signatures. The second signature database (designated hereafter as CORPUS2) has been created at our laboratory (not specified due to blind review rules) under the FrameWork of an Education and Initial Vocational Training Program of our Country. It is made of 69 sets of different signatures. Each set consists of 105 genuine and 21 forgery signatures. For both databases, we can see from inspection that the database contains signatures of various styles i.e from clear and tided to cursive and oriental. The forgery samples also, represent various levels of imitation, ranging from simple freehand up to skilled.

2.2. Pre-processing

For every image, in both databases, a pre-processing step is applied whose objective is to provide an enhanced image with maximized amount of utilized information. The pre-processing stage includes threshold of the original handwritten image using Otsu's method [2]. The resulted binary image is further processed by thinning algorithms [10] in order to provide a one pixel wide signature trace, which is considered to be insensitive to pen parameters changes like size, colour and style [1]. Finally, the bounding rectangle of the image is produced.

Next, an alignment is carried out for every bounded image. This is an attempt to gather the intrapersonal useful information from all the samples of a writer inside a region that is considered to be the one that contains the most useful handwriting information. In this work, we have used the estimated coordinates of the centre of mass $\overline{x}$ and $\overline{y}$ for each image. Accordingly, we consider:

$$m_{pq} = \sum_i \sum_j i^p j^q \text{ , where (i,j) } \in \textit{ thinned image} \quad (1)$$

which are the geometric moments of the image and,

$$\overline{x} = \frac{m_{10}}{m_{00}}, \quad \overline{y} = \frac{m_{01}}{m_{00}} \quad (2)$$

are the respective values of the centre of mass. Next, the definition of the most informative window (MIW) of the signature image is described. It is the signature sub-region, inside the bounded image, centred at $\overline{x}$ and $\overline{y}$ parameters while its length and width is confined within an $\pm$ 80% of the minimum distance of $\overline{x}$ or $\overline{y}$ from the bounding rectangle. Figure 1 illustrates the pre-processing algorithm.

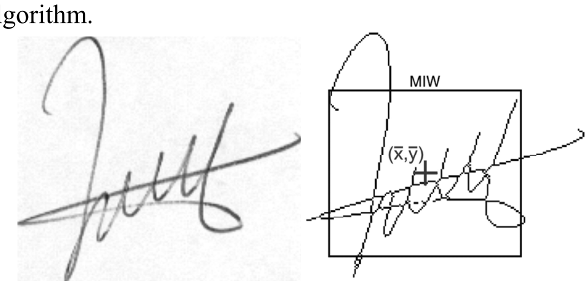

Figure 1. Pre-processing steps: The left is the original image, while the right displays the thinned image along with the MIW content.

3. Feature Extraction

The derived information from the MIW image is further processed in order to transform it to a feature vector, which describes the whole signature image by summarizing local line features like orientation and curvature. In addition, partitioning of the MIW onto sub-blocks provides essential local information by quantifying and expanding in a detailed perspective. It is known that chain coding describes the boundaries of an object. In its simplest form, eight in all, sequences of two pixels are examined, thus coding the succession of different orientations on the image grid. When sequences of three successive pixels are examined, line, convex and concave curvature features are generated [2, page 441].

For example, the direction (01) designates that there are three consecutive pixels on a run beginning from the first, then moving to an eastern direction (0) and finally terminating to a north-eastern direction (1). In our method we do not utilize the features' order of appearance. As a result, the corresponding features which can be defined uniquely, beginning from a central pixel to another one, which has a chess-board distance equal to 2, into a 5×5 window are merely 22. Furthermore, the symmetry of the 5×5 window confine further the number of convex and concave features to 11. This subset of curvature features is expanded with the inclusion of features (00), (11), (22), (33), which mainly describe fundamental line segments of slope 0, 45, 90, 135, constitutes our feature space. It is easy to see that these 15 features can be derived using only a 3×5 window mask as figure 2 shows.

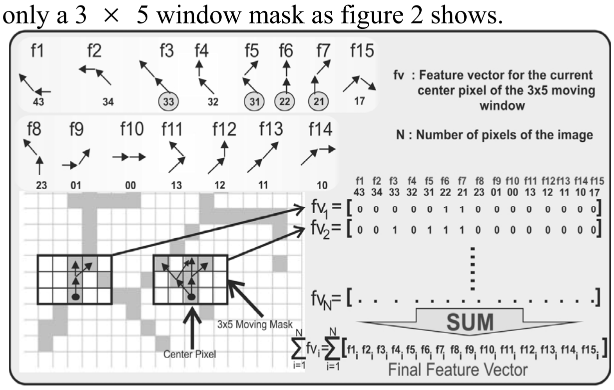

Figure 2. Feature extraction method. Detail of a signature image. The feature vectors and the feature components produced by the mask.

For each signature pixel that is part of the one pixel wide trace of the MIW image, a rectangular grid mask is applied whose dimensions are 3×5. The mask aligns each aforementioned pixel with the {3, 3} coordinate, thus enabling 15 potential 2-step paths from the central pixel to any pixel bounded to possess a chess-board distance of 2. For each signature pixel, the paths which are included in the feature set are marked and a counter updates the corresponding features found. Finally, the feature components are normalized by their total sum in order to provide a probabilistic expression.

Local handwriting variations and fine detail of the signature are emphasized by means of partitioning the MIW image to sub-regions. In this work, the MIW has been divided to 4 equal sub-windows, leading to a feature dimensionality of 60. The feature extraction method handles, in a statistical manner, the directional transition probability between neighbouring pixels.

4. The Classification Scheme

4.1. Classifier Selection

A hard margin classifier support vector machine is employed in this section in order to implement the classification scheme. The approximation of the linear learned decision function of the hard margin classifier for a two class problem is provided below:

$$\hat{f}(\bar{x}) = \operatorname{sgn}\left(\sum_{i=1}^{l} a_i^* y_i k(\bar{x}_i, \bar{x}) - b^*\right) \quad (3)$$

where: $i = 1, \ldots, l$ is the number of training samples, a_i^* are the corresponding Lagrangian multipliers of each training sample subject to appropriate constraints, $k(\bar{x}_i, \bar{x})$ is the kernel function which maps the input space into a feature space with higher dimensionality, b^* is the offset, $\bar{x}_i$ is the current training vector, $\bar{x}$ is the input feature vector and finally $\bar{y}_i$ is the label or the class that belongs to the current training vector $\bar{x}_i$. For the target function we have that $f : \mathbb{R}^n \to \{+1, -1\}$ where n denote the dimensions of the input feature space R and f is the label function. The labeled separated training vectors are given below:

$$D = \{(\bar{x}_1, y_1), (\bar{x}_2, y_2), \ldots, (\bar{x}_l, y_l)\} \subseteq (\mathbb{R}^n, \{+1, -1\}) \quad (4)$$

where $y_i = f(\bar{x}_i)$.

Gaussian Radial Base Functions have been selected for mapping the non linear separated input feature space into a linear one , with optimal value for the standard deviation parameter σ = 6. Optimal value has been estimated with trial an error method.

4.2. Evaluation Protocol

In order to evaluate the first stage of a signature verification system, for the purposes of this paper, our protocol evaluates the random forgery case. For each writer a separate model is being created. The realization of the training set for the genuine class, for each writer, is accomplished using their genuine samples. These, were selected randomly by using the hold-out validation method in order to avoid over-fitting of the training samples by limiting the training data noise in the evaluation process. During the hold-out method the available data set is divided into two subsets. The first subset is used for training while the other one is reserved for testing. In our case, the hold out method was implemented for both signature databases. The experimental protocol for the two databases is presented here: In case of CORPUS1, the representation of the genuine class, among the 24 samples of each writer, is made by using 4, 6, 8, 10, 12 and 14 samples for training which leads to 20, 18, 16, 14, 12 and 10 samples for testing respectively. For the CORPUS2 database, the representation of the genuine class from the 105 samples of each writer and the hold out method use 5, 10, 15, 20, 25, 30 samples for training and 100, 95, 90, 85, 80, 75, 70 samples for testing respectively.

The realization of the training set for the forgery class follows the protocol which is described below. For the CORPUS1, one sample from each of the other 19 genuine writers (19 samples total) has been used for the creation of the training set. All remaining samples of the 19 genuine writers have been used for testing according to the hold out method. For the CORPUS2, one sample from each of the other 68 genuine writers (68 samples totally) has been used for the modelling of the forger class while the rest of all the samples of the 68 genuine writers employed for testing according to the hold out method.

Solid and comparable results, for the training and testing procedure, have been extracted by repeating the procedure six times. This is because the hold out procedure introduces a distortion on the calculation of the error rates since an 'unfortunate' split between the train and test samples may occur. In the evaluation process False Rejection Rate (FRR) and False Acceptance Rate (FAR) error were considered as the critical parameters of the overall system efficiency. Finally the average error was calculated according the below form for the overall performance evaluation of the proposed verification system.

$$\text{AvE} = 0.5(\text{FAR} + \text{FRR}) \quad (5)$$

A primary indication of the significance of the method can be viewed by comparing our results with those which are derived by an already published work [3]. They have proposed a texture-based feature which uses fusion of rotation invariant local binary patterns (LBP) operators and gray level co-occurrence matrices (GLCM). Accordingly, the implementation of feature extraction method has been followed to the letter, in order to obtain their feature space representation. This stage provides the ability to benchmark our method with already published techniques.

5. Experimental Results

The performance of our proposed method is provided in Tables 1 and 2 for the corresponding corpuses and in the case of random forgeries. Commenting out the results, on behalf of our method, we can note the extreme low probability of false alarm (FAR), for both corpuses, which lies at the vicinity of 0.5%. This outcome holds for every classification protocol that has been applied to the corpuses, according to the previously exposed discussion. It is worth of attention that a probability of 0.01% for the FAR rate is achieved. This is a strong indication that our method provides a powerful discrimination for the FAR case. For the FRR measure we can observe that, the corresponding probability of error, decreases as the number of added genuine training samples into the classifier, is increasing, which is a presumable result. It is noted also that, augmenting the set of genuine training samples causes an increase of the FAR error. This is expected due to the reason that, there is always a trade off between the FAR and the FRR parameters. Concluding, it seems that as the model absorbs the intrapersonal variability by adding more genuine samples, it also allows the FAR to increase by embracing more forgery samples [11].

Comparison of our work with the method provided by the literature [3] shows that, ours seems to provide encouraging results to a substantial number of classification schemes, in terms of both FAR, FRR and AvE. This assumption is partially valid in the CORPUS1 database.

This could be explained by noting that CORPUS1 has a smaller amount of samples than CORPUS2. Additionally, we can allege that the Average Error in the first three cases of CORPUS1 is quite large, so actually, no one can actually debate about the differences. The comparable results are presented in Tables 3 and 4.

Table 1. CORPUS 1 Proposed method results

# Training / Testing Samples		Results		
Genuine	*Forger*	*FRR*	*FAR*	*AvE*
4 / 20	19 / 437	37.47%	0.03%	18.75%
6 / 18	19 / 437	28.61%	0.02%	14.32%
8 / 16	19 / 437	11.29%	0.03%	5.80%
10 / 14	19 / 437	12.45%	0.80%	6.63%
12 / 12	19 / 437	4.58%	0.76%	2.67%
14 / 10	19 / 437	6.57%	1.22%	3.90%

Table 2. CORPUS 2 Proposed method results

# Training / Testing Samples		Results		
Genuine	*Forger*	*FRR*	*FAR*	*AvE*
5 / 100	68 / 7072	57.47%	0.01%	28.74%
10 / 95	68 / 7072	21.14%	0.11%	10.63%
15 / 90	68 / 7072	9.10%	0.33%	4.71%
20 / 85	68 / 7072	5.54%	0.70%	3.05%
25 / 80	68 / 7072	5.00%	0.73%	2.87%
30 / 75	68 / 7072	3.30%	1.10%	2.20%

Table 3. CORPUS 1 Texture Based Feature results

# Training / Testing Samples		Results		
Genuine	*Forger*	*FRR*	*FAR*	*AvE*
4 / 20	19 / 437	35.72%	0.51%	18.12%
6 / 18	19 / 437	27.22%	0.54%	13.88%
8 / 16	19 / 437	10.00%	1.98%	6.00%
10 / 14	19 / 437	11.38%	2.48%	6.93%
12 / 12	19 / 437	3.33%	3.55%	3.44%
14 / 10	19 / 437	2.58%	4.14%	3.36%

Table 4. CORPUS 2 Texture Based Feature results

# Training / Testing Samples		Results		
Genuine	*Forger*	*FRR*	*FAR*	*AvE*
5 / 100	68 / 7072	72.93%	0.10%	36.52%
10 / 95	68 / 7072	40.00%	0.53%	20.28%
15 / 90	68 / 7072	18,43%	1.32%	9.88%
20 / 85	68 / 7072	9.73%	2.09%	5.91%
25 / 80	68 / 7072	7.56%	2.88%	5.22%
30 / 75	68 / 7072	6.26%	3.55%	4.91%

6. Conclusion

In this work an off-line signature verification method is proposed. The feature extraction method utilizes the summing of local line features like orientation and curvature on a curvature feature that is extracted from portions of a signature image. For each pixel of the one-pixel-wide signature trace, the paths which are included in the feature set are marked and a counter updates the corresponding active components found. In order to provide comparable results we have employed a widely used database plus a new database which was constructed by our team. In addition, we have chosen to compare our approach with an already published one in order to evaluate the strength of our method. For each writer a dedicated SVM classifier has been employed for the first stage of the verification stage. The experimental procedure provides low verification error rates which are comparable to other referenced works. Further research is directed towards the evaluation of our work against skilled samples and the enhancement of our proposed method by incorporating other texture and chain code based features. The proposed methodology feature extraction methodology has the advantage that the majority of the operations are carried on binary images using Boolean masks which leads to a feasible, fast hardware implementation.

References

[1] D. Impedovo, G. Pirlo, "Automatic signature verification: The state of the art", *IEEE transactions on Systems Man and Cybernetics*, vol 38, no. 5, pp. 609-635, 2008.

[2] S. Theodoridis, K. Koutroubas, "Pattern Recognition", *Academic Press,* 2009.

[3] J. F. Vargas, M. A. Ferrer, C. M. Travieso, J. B. Alonso : "Off-line signature verification based on grey level information using texture features," *Pattern Recognition* , vol. 44, no.2, pp. 375-385, 2011.

[4] D. Bertolini, L.S. Oliveira, E. Justino, R. Sabourin, "Reducing forgeries in writer-independent off-line signature verification through ensemble of classifiers", *Pattern Recognition,* Vol. 43 pp.387-396, 2010.

[5] A. Gilperez, F. Alonso-Fernandez, S. Pecharroman, J. Fierrez, J. Ortega- Garcia, "Off-line signature verification using contour features", *Proceedings of the International Conference on Frontiers in Handwriting Recognition*, ICFHR, 2008.

[6] I. Siddiqi, N. Vincent, "Combining Contour Based Orientation and Curvature Features for Writer Recognition", *Lecture Notes in Computer Science*, Volume 5702, pp. 245-252, 2009.

[7] E. N. Zois, A. A. Nassiopoulos, V. Anastassopoulos, "Signature verification based on line directionality", *IEEE SIPS 2004*, pp. 343 – 346, 2005.

[8] S. Baheerathan, F. Albregtsen and H. E. Danielsen, "New texture features based on the complexity curve", *Pattern Recognition*, vol. 32, pp. 605-618, 2002.

[9] K. Huang and H. Yan, "Off-line signature verification using structural feature correspondence", *Pattern Recognition*, vol. 35, pp. 2467-2477, 2002.

[10] L. Lam, S. W. Lee, and C. Y. Suen: "Thinning Methodologies - a Comprehensive Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 9, pp. 869-885, 1992.

[11] E. J. R. Justino , F. Bortolozzi, R. Sabourin, "A comparison of SVM and HMM classifiers in the off-line signature verification", *Pattern Recognition Letters* vol. 26, pp. 1377–1385, 2005.