ROBUST OFF-LINE SIGNATURE VERIFICATION USING COMPRESSION NETWORKS AND POSITIONAL CUTTINGS

José F. Vélez, Ángel Sánchez and A. Belén Moreno
Escuela Superior de Ciencias Experimentales y Tecnología
Universidad Rey Juan Carlos
c/ Tulipán, s/n
28933 - Móstoles (Madrid), SPAIN

Abstract - A novel robust technique for the off-line signature verification problem in practical real conditions is presented. The technique is based on the use of compression neural networks, and in the automatic generation of the training set from only one signature for each writer. Our proposal incorporates a new kind of acceptance/rejection rule, which is based on the similarity between subimages or positional cuttings of a test signature and the corresponding representation stored in the class compression network. Experimental results show that the proposed technique reduces significantly the False Acceptation Rate (FAR).

1. Introduction.

Signatures are a special case of handwriting subject to intrapersonal variation and interpersonal differences. This variability makes necessary to analyse signatures as complete images and not as collection of letters and words [4]. As human signatures provide secure means for authentication and authorisation in legal and banking documents, the need of research in efficient automatic solutions for the involved signature recognition and verification problems has increased [13]. In the signature recognition problem a given signature is searched in a database to establish the signer's identity. Signature verification problem is concerned to determinate if a particular signature belongs to a person, to decide if it is authentic or a forgery. Techniques for solving the verification problem can be classified as on-line and off-line [9]. In the first ones, data are obtained using an electronic tablet and other devices and in the second ones, images of the signatures written on a paper are obtained using a scanner or a camera and not any dynamic information is available.

In this paper, we present a new off-line technique for the signature verification problem. The technique is based on the use of a compression neural network for testing each signature class which corresponds to each writer in the database. When verifying a signature, compression networks are used in combination with a method that compares the internal representation (stored in the network) of the signature class with subimages or cuttings which form a partition of the tested signature.

Many approaches for the automatic verification problem have been reported in the literature [1][2][6][9]. In general, proposed techniques use either global, statistical, geometric and local features (or a combination of different features) extracted from the signature images [1][11]. We propose a holistic off-line verification approach, requiring neither from any signature pre-processing nor any feature extraction stage [3]. Our approach considers the practical involved problems and requirements problems in automatic verification systems, such as: lack of training samples, variability of signature patterns (due to intra-personal or inter-personal variations), presence of noise, and efficient performance.

The paper is organised as follows. Section 2 illustrates the practical off-line signature verification problems. In section 3, an overview of the proposed verification scheme is presented. Section 4 describes how the training stage is performed using compression networks. Section 5 outlines the devised positional signature cuttings method for the verification stage. Our signature database and verification experiments are presented in Section 6. Finally, conclusions are resumed in Section 7.

2. Practical off-line signature verification problems.

Real problems involved in off-line signature verification can be classified in two main categories: those related to the extraction of the signature from the document and those derived from the characteristics of the verification task. The first category includes problems due to the need of segmenting the signature from the image document. Some examples are the different positions of signatures inside the document, presence of Gaussian noise caused by document scanning, existence of texture and logotypes in the document background (i.e. in back cheques), and the possibility of presence or even superposition of stamps and typed text mixed with the signature, among others. Figure 1 shows some examples of situations where the signature segmentation becomes a difficult task.

(a) (b) (c) (c)

Figure 1. Signature image examples presenting (a) noised background, (b) textured background, (c) stamp superposition and (d) typed text.

Among the problems related to the verification process, we mainly observe the lack of sufficient signature samples from each writer for training up the system. This difficulty is a very realistic problem due to the private nature of signatures. It would also be desirable to collect signatures from a writer in different periods of time to capture the intrapersonal signature variations [6]. Moreover, with an insufficient number of global training samples, the determination of interpersonal signature differences among writers could result unreliable. In this paper, a method for the automatic generation of additional training samples from only one original writer's signature is proposed. The additional signature patterns are used to train a compression network for each writer.

Other related signature verification problems are the scalability of the system (when adding new writers), the acceptable response time of the system when automatically processing a large amount of documents, and how to consider FAR (False Acceptance Rate) and FRR (False Rejection Rate) values [3]. We have analysed and considered these real practical problems when designing and implementing the proposed off-line signature verification method.

3. Off-line signature verification approach.

This section sketches the architecture of our signature verification system which is organised in two main stages: training and verification, respectively. Compression networks, which are used as essential components of both stages, are shortly first described.

3.1. Compression networks.

An interesting aspect of back-propagation networks is that during learning process, the hidden layers build an internal representation of the inputs that is useful to produce the output [10]. Fleming and Cottrell [5] used a two-stage neural network with the same number of neurones for input and output layers, and fewer units for the hidden layer. This forces the network to encode the inputs in a smaller dimensional space retaining most of the relevant information in an equivalent way as the Principal Component Analysis (PCA) method. This class of networks are known as compression networks. A formal description of them can be found in Valentin et al [14], where they investigated the application of compression networks to face recognition. Valentin used the representation formed in the hidden neurones of this network as input to a single perceptron used as a classification network.

An important property of compression networks is that they can act as auto-associative or content addressable memories [8][14]. This means that these networks are able to acceptably reconstruct a degraded image pattern when a noise image is given as input or to complete an incomplete image input pattern [12]. The quality of the results will depend on the number of hidden units of the compression network. This property is used in this work to precisely determine the position of each signature cutting resulting from the partition of an original test signature used during the verification stage.

3.2. Architecture of proposed signature verification system.

Figure 2 presents the complete off-line signature verification architecture represented as an activity UML diagram. It is divided in two components or stages: learning and verification, respectively. To simplify the representation of the training stage, we have only drawn the process corresponding to one writer (who originally signs once), which produces as result a trained compression network. This process is repeated for each one of the *m* writers, and in total we generate *m* corresponding trained compression networks. Next two sections respectively describe with more detail how training and verification stages are performed in our method.

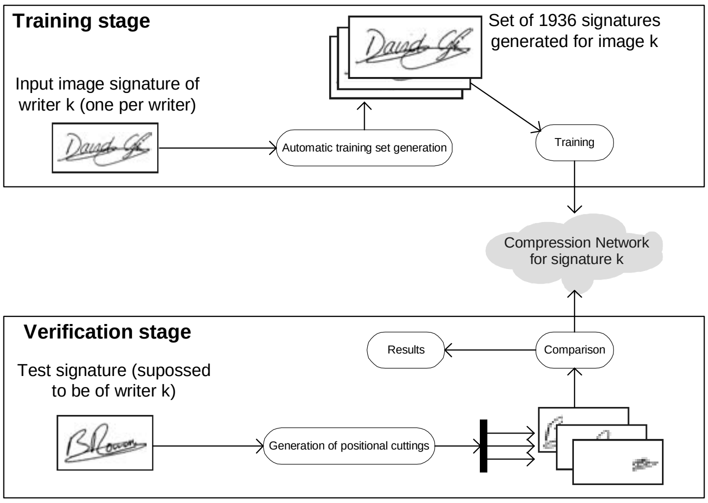

Figure 2. Activity UML diagram of stages in the proposed signature verification method.

4. Training stage: automatic generation of training samples.

There are inevitable variations in the signature patterns produced by the same person (intrapersonal variability) [4][6]. This problem is usually solved by capturing a large amount of different signatures of each individual for the training set. However, it can result difficult or even non feasible to get multiple signatures from each individual in a real problem (i.e. banking documents [13]). That is why we have automatically generated a complete training set for each writer using only one corresponding signature.

The complete signature pattern set used to train all the different compression networks (each one created for a writer) was automatically produced using script-programming. The created scripts transform the only “real” signature of each writer to produce an associated training set of new similar synthetic signatures presenting slight variations. The same task is performed for the rest of the writers. Transformations applied on the signature patterns were the following: rotations (in the range of ±15º), scalings (in the range of ±20%), horizontal and vertical displacements (in the range of ±20%), and different types of noise additions. These transformations, which try to emulate the intrapersonal variability into the signatures of each writer, were applied in different combinations for each one of the “original” signatures. Thus, each of the original signatures generated 1936 synthetic signature patterns.

These patterns were normalized to fit a rectangle of n pixels, where n is the closest number to 800 that defines a rectangle proportional to the bounding box of the original signature. In this way, the training set for each compression network (one per writer) was built. The number of hidden units of each compression network was experimentally defined and set to 20 (other network configurations for the network were tested with worse results). In this way, compression networks with n inputs, 20 hidden units, and n outputs were trained. Weights of neurons were initialised in the range of -0.0001 and 0.0001, the defined activation function was identity, and the backpropagation algorithm with a learning rate of 0.0001 was used. Figure 3 shows the evolution of Mean Square Error (MSE) value with the number of training iterations of the compression network for the class (writer) 28. Note that, as predicted by Baldi [2], the curve has not local minimums and seems to have only a global minimum.

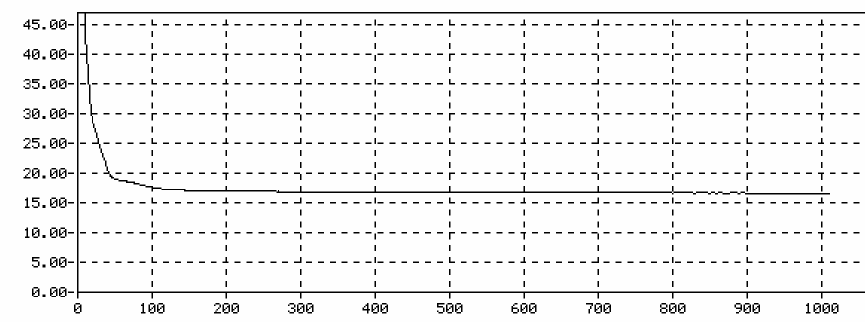

Figure 3. MSE evolution for training the compression network of writer 28.

5. Verification stage using positional signature cuttings.

Kim and Lee [7] suggested a rejection rule based on the comparison of the MSE value measured between the input and the output of the compression network. If the error is smaller than a defined threshold value, the pattern is accepted; otherwise, it is rejected. These authors have successfully applied this rule to neural networks in a speech recognition domain.

In order to apply the threshold rejection method for our signature verification problem, several experiments were carried out. However, it has been experimentally checked that in about 25% of the patterns, MSE for test signatures is lower with false signatures (forgeries) than with genuine signatures. This fact invalidates the application of the threshold rejection method to our signature verification problem.

The previous result makes sense. As compression network acts as an autoassociative memory. When a signature is presented to the network, it searches the most similar learned pattern, and this result is returned as output by the network. Suppose that the compression network is trained with signatures of writer i, for $i \in \{1..m\}$. Now suppose that signatures s_1 of writer i and s_2 of a different writer are input to the compression network. Now, consider that the network returns as results the representations t_1 and t_2, respectively. Even though s_2 and t_2 were completely different and s_1 and t_1 were quite similar, the MSE_1 value between s_1 and t_1 could be greater than the MSE_2 value between s_2 and t_2. For example, this could happen when s_1 and t_1 have a "real" size (i.e. the area of the convex hull corresponding to the signature) of n_1 pixels and s_2 and t_2 have a size of n_2 pixels, and $n_1 >> n_2$. In addition, it could occur that s_2 with t_2 randomly have a high percent of common pixels and s_1 with t_1 has only a little number of common pixels. This happen if the strokes of signature s_1 was quite wider than the strokes of signature t_1. Performed experiments confirm the previous hypothesis. Consequently, it makes impossible to find a MSE threshold value to distinguish between genuine signatures and forgeries.

As compression networks have the capability to act as autoassociative memories, when incomplete images are presented to the network, it fills the missing parts. This characteristic could be used to construct a robust system for the problem of signature verification. To do that, the pattern of the signature is partitioned into n vertical subimages or cuttings. Then, each of these cuttings is presented to the network in different positions of the input (filling the rest of the input signature with –1 values). MSE values between the input and the output are measured when comparing the input and output for a pattern. Intuitively, if both the presented signature to the compression network and the signature used to train the network belongs to the same writer, it is highly probable that, the position of the cutting at the input where the error is minimum matches the correct position of the cutting. Otherwise, if the signature belongs to a different writer, the position of the minimum error will be random. This experiment repeated for the n component cuttings of a signature, as schematically shown in Figure 4, can be used to measure the confidence in a test signature as result of its verification.

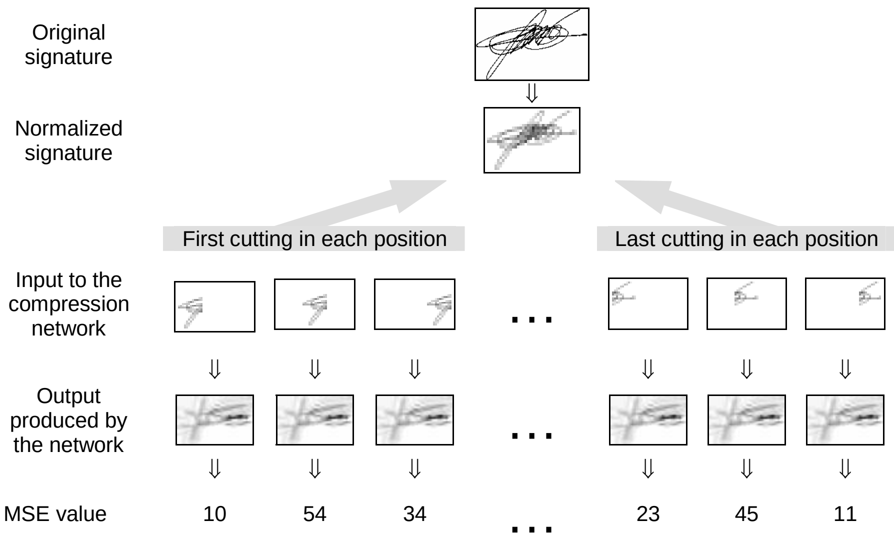

Figure 4. Sketch of the positional cuttings method applied to the verification stage.

The signature cuttings must have enough size to let compression network to reconstruct with fidelity the signature. Moreover, the bounding box of a signature has a width/height ratio greater than one. It means that signature variability along the vertical direction is minor than in the horizontal direction. This makes vertical cuttings to be more robust than horizontal ones to reconstruct signatures in our method.

Multiple confidence parameters have been evaluated. A sketch of the proposed algorithm to measure the confidence on a signature as result of the verification stage follows:

```
error_ratio = MinimunErrorForCutting / ErrorOfCorrectPositions;
ascending_acumulate = 0;
for (int cutting_cont = 0; cutting_cont < number_of_cuttings-1; cutting_cont++){
     if (MinimunErrorPosition(cutting_cont) < MinimunErrorPosition(cutting_cont+1)
              ascending_acumulate += 1;
     else if (MinimunErrorPosition(cutting_cont) > MinimunErrorPosition(cutting_cont+1)
              ascending_acumulate -= 1;
}
ascending_acumulate /= number_of_cuttings;
confidence = ascending_accumulate * error_ratio;
```

Note that this algorithm contains two main parameters:

- `ascending_accumulate` which represents the ordering of the cuttings using the computed MSE value (between 0 and 1), and
- `error_ratio` which measures the difference between the errors obtained for the correct position of the cuttings and the error obtained for the position that minimizes the MSE value.

The product of these two factors is normalized between 0 and 1 and it is used as a signature confidence measure. It must be remarked that this value is adimensional and let us to define a global confidence threshold.

6. Experimental results.

This section describes our signature database and the performed experiments.

6.1. Signature database.

Due to the lack of reference signature databases, we have created our own database (http://web.madritel.es/personales3/jvelezserrano/firmas). The inexistence of referenced common signature databases makes unfeasible the experimental comparison of our method with other existing methods. The database consists of 28 individuals, with 4 signatures for each writer (for testing purposes of our method, 3 original and 1 forged signatures are available for a few individuals). Patterns correspond to different kinds of writing styles, and were scanned with a resolution of 300 dpi and 256 grey levels, and stored in BMP format.

Figure **5** shows some sample signatures corresponding to four different writers in the database.

Figure 5. Some signatures in our database used for the experiments.

6.2. Experiments.

Three compression networks were built for three different signatures (one per writer). Each network has been tested with 108 signatures (104 false signatures and 4 true signatures for each case). Table 1 presents the results of using the Kim and Lee's threshold rejection rule [7], described in Section 5. A different threshold was selected for each network. The threshold value was aposteriori chosen for a correct classification of all the true patterns.

	Number of Signatures	Success classification	False Rejection Rate (FRR)	False Acceptance Rate (FAR)
Number of signatures	336	252	0	84
Percentage	100%	75.0%	0.0%	25.0%

Table 1. Verification results produced using Kim and Lee's threshold rejection rule.

Figure 6 shows the results of our proposed positional cuttings method. FRR and FAR results are represented as a function of the proposed confidence measure. These results were obtained using overlapped cuttings for each signature. It can be observed in Figure 6 that the threshold of 15% for the confidence measure produces good results. Table 2 particularises Figure 6 for this threshold value. Improved verification results achieved by our method can be compared with those presented in Table 1.

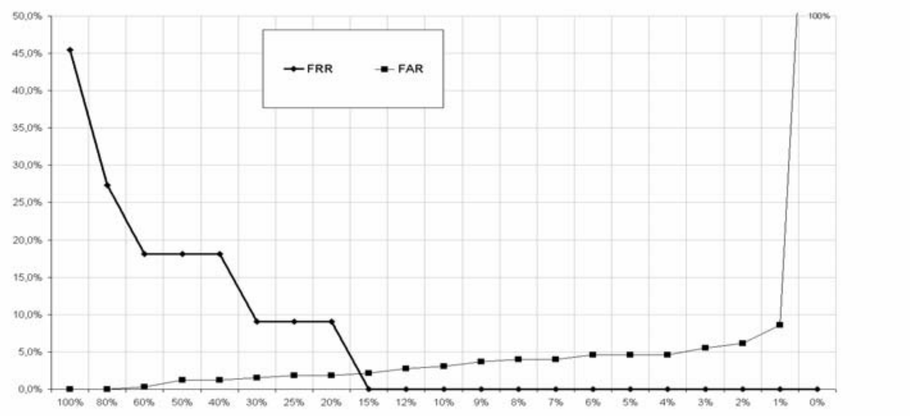

Figure 6. Verification results of the proposed method.

	Number of signatures	Success classification	False Rejection Rate (FRR)	False Acceptance Rate (FAR)
Number of signatures	336	329	0	7
Percentage	100%	97.9%	0%	2.1%

Table 2. Verification results of the proposed method for a threshold value of 15%.

7. Conclusions and future work.

Related problems with the off-line signature verification task in real conditions have been analysed. A new off-line signature verification method has also been proposed. It is based in three main components: use of compression networks, automatic training pattern set generation and a positional cuttings criterion for confidence analysis of a signature. Our approach can simultaneously handle the practical signature analysis problems previously described. Particularly, the method only needs one signature in order to generate the training set. However, this advantage makes difficult the comparison of our method with other existing methods which use more than one signature for the learning stage. Moreover, it is not conditioned by horizontal displacement, because it does not depend on the signature starting point for the analysis. Other important features of our proposal are the non-influence of signature vertical displacements in the verification task, and the robustness in presence of noise due to the compression network capabilities. In addition, the approach is completely scalable because it allows adding new signatures (classes) to the system, training a new compression network for each new signature without modifying the rest of available trained networks. Finally, our method does not need from any human assistance in the training stage because compression networks ensure the existence of a global minimum existence and the inexistence of local minimum values.

One possible improvement to our method is the application of Hidden Markov Models (HMM) for a probabilistic interpretation of the signature verification confidence measure. The proposed off-line signature verification approach has been tested with a relatively small set of signatures. We will expand our database and will apply the technique to such a new situation.

References.

[1] R. Bajaj and S. Chaudhury, "Signature Verification using Multiple Neural Classifiers", **Pattern Recognition**, Vol. 30, no. 1, pp. 1-7, January 1997.

[2] P. Baldi y K. Hornik, Neural networks and principal components analysis: learning from examples without local minima", **Neural Networks,** nº 2, pp. 53-58, 1989.

[3] R.O. Duda, P.E. Hart and D. G. Storck, **Pattern Classification**, 2nd Edition, NY: John Wiley & Sons, 2001.

[4] B. Fang et al, "Off-line signature verification with generated training samples", **IEE Proc.-Vis. Image and Signal Processing**, Vol. 149, no. 2, pp. 85-90, April 2002.

[5] M.K. Fleming and G.W. Cottrell, "Categorizacion of faces using unsupervised feature extraction", **Proc. Int. J. Conf. on Neural Networks II**, pp.65-70, 1990.

[6] E.J. Justino, F. Bortolozzi and R. Sabourin, "The Interpersonal and Intrapersonal Variability Influences of Off-Line Signature Verification using HMM", **Proc. XV Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2002)**, Fortaleza (Brazil), October 2002.

[7] D.S. Kim and S.Y. Lee, "Intelligent judge neural network for speech recognition", **Neural Processing Letters**, Vol. 1, 1994.

[8] T. Kohonen, **Associative Memory: A System Theoretic Approach**, Springer, Berlin, 1977.

[9] F. Leclerc and R. Plamondon, "Automatic Signature Verification: The State of the Art", **Intl. J. Pattern Recognition and Artificial Intelligence**, Vol. 8, no. 3, pp. 643-660, March 1994.

[10] C.G. Looney, **Pattern Recognition using Neural Networks: Theory and Algorithms for Engineers and Scientists**, Oxford University Press, 1997.

[11] Y. Muzukami, M. Yoshimura, H. Miike and Y. Yoshimura, "And off-line signature verification system using an extracted displacement function", **Pattern Recognition Letters**, Vol. 23, pp. 1569-1577, 2002.

[12] O'Toole et al, "Low-dimensional representation of faces in higher dimensions of the face space", **J. Opt. Soc. Am. A**, Vol. 10, no. 3, pp. 405-411, March 1993.

[13] C.Y. Suen, Q. Lu and L. Lam, "Automatic recognition of handwritten data on cheques – Fact or fiction?", **Pattern Recognition Letters**, Vol. 20, pp. 1287-1295, 1999.

[14] D. Valentin, H. Abdi, A.J. O'Toole and G.W. Cottrell, "Connectionist Models of Face Processing: A Survey", **Pattern Recognition**, Vol. 27, no. 9, pp. 1209-1230, September 1994.