An artificial neural networks architecture for handwritten signature authentication

Hubert Cardot, Marinette Revenu, Bernard Victorri, Marie-Josephe Revillet

1993

An artificial neural networks architecture for handwritten signature authentication

Hubert Cardot (1), Marinette Revenu (1), Bernard Victorri (2), Marie-Josèphe Revillet (3)

(1) LAIAC (Laboratoire d'Algorithmique et d'Intelligence Artificielle de Caen)
ISMRA, 6 bd. du Maréchal Juin, 14050 Caen Cedex, France, Tel. : +33 31 45 27 19
(2) ELSAP, Université de Caen, 14032 Caen Cedex, France
(3) SEPT, 42 rue des Coutures, 14000 Caen, France

I. Introduction

I.1. Presentation of the problem

It is frequently asked to individuals to prove their identity when writing official documents. This is done to avoid the use of someone else's signature and also to avoid that someone disowns a document that he has previously acknowledged. Texts are often typed, so it is not possible to authenticate these documents from handwriting. However, it is customary to append a mark authenticating the author of the document, thus showing that he agrees with the text of the document. Nowadays this mark is generally a handwritten signature, so it is interesting to devise an automatic and reliable system for the authentication of handwritten signatures appended on the numerous documents which are produced daily.

The difficulty of the signature authentication problem is linked to the high number of writers, to the diversity of signatures to store, and also to the important variations between signatures from the same writer [Sabourin 90]. The authentication problem is different from the identification problem because the latter consists in determining the writer from his signature. In the authentication case, we know the writer who is supposed to have signed, as his name is written on the document, for example a check. So it is possible to access in a database to the signatures given by the writer to be used as reference signatures. Then, the authentication process consists in comparing the signature to the reference ones in order to judge if the supposed writer is really the author of the tested signature.

The signature authentication can be used in several applications ; let us now focus on the verification of checks from the French Post Office.

Our goal is to detect rough forgeries, which are signatures written by someone who is not imitating a genuine signature. Those rough forgeries are the most commonly found forgeries. Systems based on dynamic information (duration, speed of the signing, ...) are able to detect good imitations. In our application however, this dynamic information is lost because the image of the check contains only static information.

Without major modifications, the authentication module of our system can be used by authentication systems based on other types of data such as digital fingerprints or dynamic information about the signatures.

I.2. Use of neural networks

Signature authentication achieved by human experts is a difficult task to model. To solve this problem, Neural Networks (NNs) seem more appropriate than symbolic methods :
- The learning and the generalisation abilities of NNs should be helpful to cope with the diversity and the variations of signatures.
- Once the learning is achieved, the response of a NN to an input is extremely fast; whereas learning can be done off-line, it is during the exploitation phase (treating a flow of checks) that rapidity is necessary.
- It is possible to compare two images with NNs, which is difficult and long with classical methods.
- Though it is not presently included in our system, we can do continuous learning : to follow the evolution of the signatures with time, it is possible to regularly do again the learning of the NNs with new signatures.

II. Handwritten signature authentication

II.1. Handwritten signatures

Two types of signatures are distinguished:

- American type, which are cursive signatures. It is possible in this case to recognize characters which form the signatures, therefore allowing to eliminate more easily rough forgeries.
- European type, which are graphical signatures that must be processed globally.

Two kinds of information can be extracted from signatures :

- static information, generally acquired with a CCD camera, constitutes the image of the signature. The acquisition can be done off-line, that is to say after the signature appending.
- dynamic information, acquired by a digitalisation device, contains the speed of the signing, its duration, its acceleration, and the lifting of the pen (tip of the pen no longer touching the support). It needs an on-line acquisition device while the writer is signing.

We can also define pseudo-dynamic information corresponding to dynamic information obtained from static information : fluctuations of the line width give an indication about the pressure fluctuations exerted on the pen.

Our system is working on European type signatures and only use the static information.

II.2. Description of a signature authentication system

II.2.1. General view

A signature authentication system is divided into two modules (fig. 1) :

- an acquisition module
- an authentication module

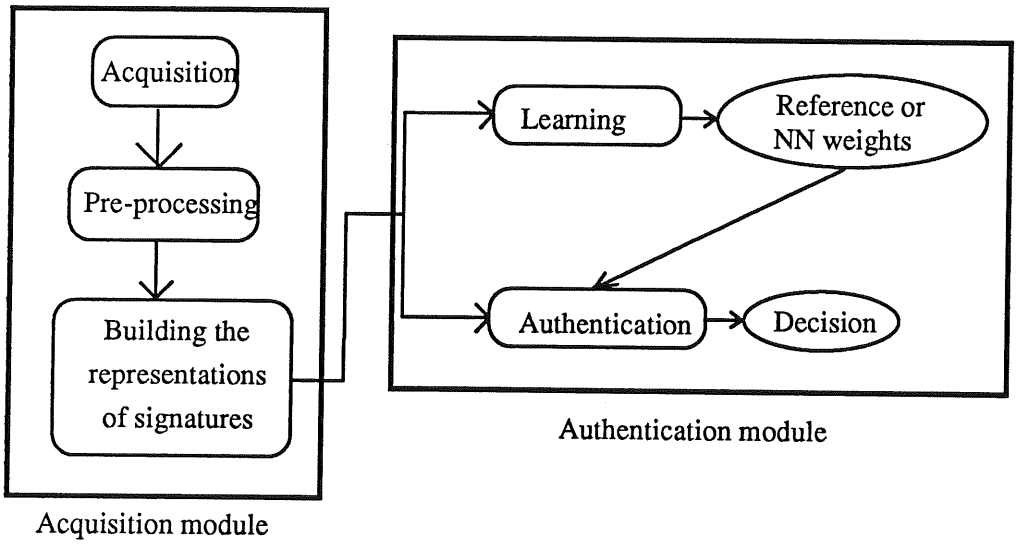

figure 1 : A signature authentication system

The acquisition module starts with the digitalisation of the image of the check. For that purpose, either a 2D-CCD camera or a 1D-CCD camera in front of which the check passes by, can be used. Then, a pre-processing step occurs to improve the image quality and eliminate what does not belong to the signature. Finally, parameters needed by the authentication module are extracted : geometrical parameters, outline and image of the signature in our case.

The authentication module uses, as its input, data extracted from the check by the acquisition module. Two phases are distinguished.

II.2.2. Learning phase

In systems based on statistical methods, this phase consists in building the reference associated to each signatory. This reference can be made of several signatures, or of an average signature calculated from the reference signatures.

In the case of neural methods, the learning phase consists in showing signatures to the NN and in modifying its weights according to its responses. Thus, the reference is "contained" in the weights of the network, which is all we have to store. One of the interests of this method stems to the fact that the amount of data about each signatory does not increase according to the number of reference signatures used.

II.2.3. Exploitation phase

In systems based on statistical methods, during the exploitation phase, the presented signature is compared with the reference of the supposed signatory and the system has to take a decision either to accept or to reject the signature according to the result of this comparison.

In systems based on NNs, the decision is taken according to the response of the NNs.

II.2.4. Evaluation of an authentication system

To evaluate the performances of an authentication system, two rates are generally computed : the false rejection rate (FRR) and the false acceptance rate (FAR).

We can consider the signature authentication problem as a two classes partitioning problem. For a given signatory, one of the two classes is formed by his signatures and the other one by all the others' signatures.

In the ideal case, these two classes are separable by an hyper-surface in the space of the signature representations (figure 2).

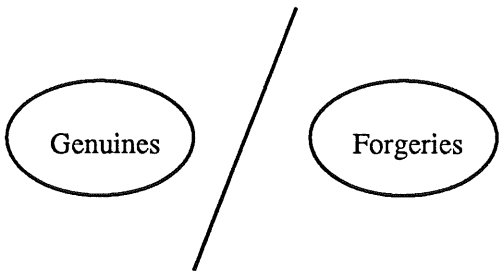

figure 2 : Separable classes

The authentication problem is mainly to find the form and position of this hyper-surface. Generally, authentication systems use a parameter, called the decision threshold in order to modify the hyper-surface. Thus, if the two classes are separable, there exists a value for this decision threshold that nullyfies the two error rates FRR and FAR (figure 3).

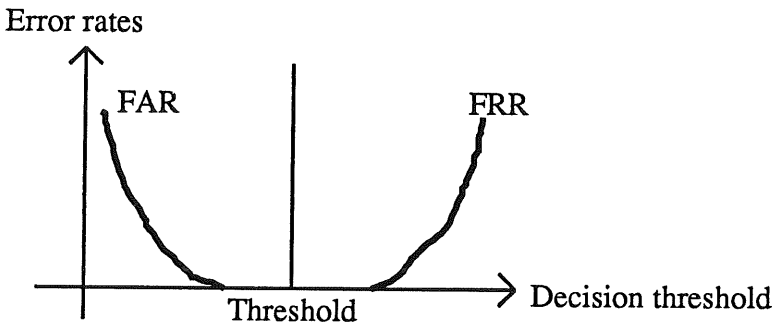

figure 3 : Choice of a decision threshold that nullyfies the error rates

Practically, with most of the representations used, we cannot obtain disjoint classes (figure 4).

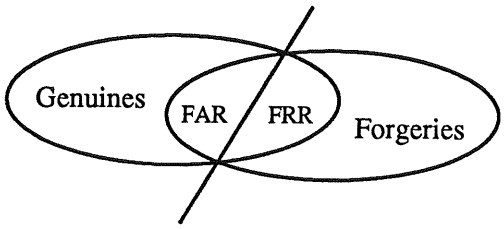

figure 4 : Non separable classes

Then the choice of the decision threshold follows one of the following criteria (figure 5) :

- minimising the average of FAR and FRR.
- keeping one of the two rates below a desired rate (for instance FAR below 1 %).

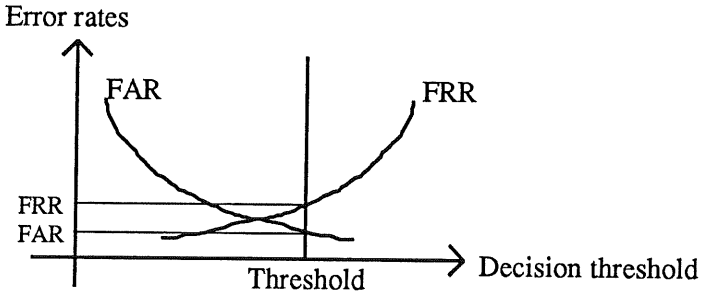

figure 5 : Choice of a decision threshold based on a criterion

II.2.5. Global view of our authentication system

Our authentication system takes up the different elements from the general authentication system (see § I.2.1). Figure 6 gives an overview of the system.

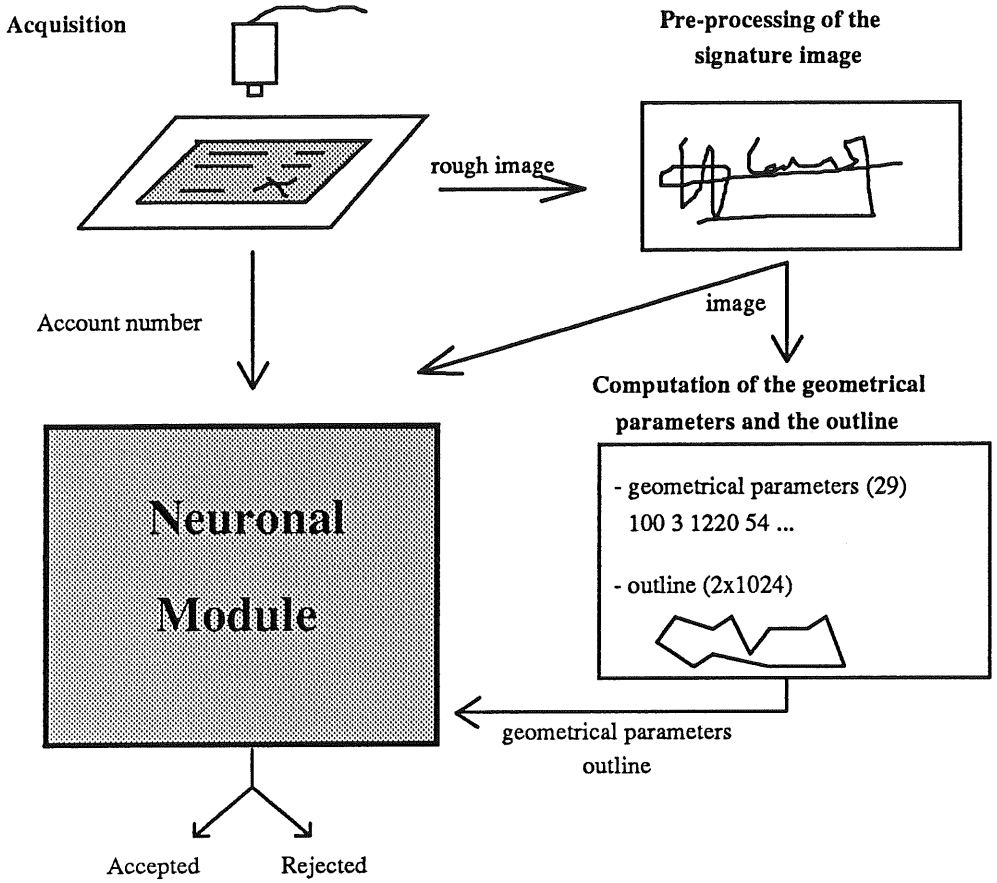

figure 6 : Our authentication system

II.3. Signature representation

The image of a signature represents an important amount of data which is difficult to process globally. That is the reason why it is common, as in other image interpretation systems, to extract parameters that should be informative, discriminating and stable for each signatory.

It is possible to use the pixel image itself. The idea is to take data as rough as possible, because the processing which aims at extracting significant information also removes useful information. However, we cannot use the image without making some pre-processing to reduce the amount of data, which is too large to be processed straight away. For example, our signature image are coded with 1024x512 pixels on 256 grey levels, occupying 512Kb.

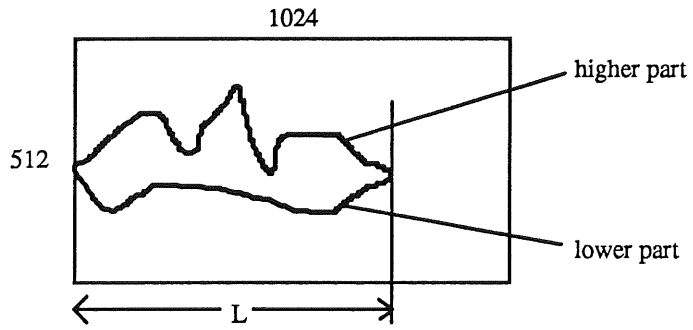

figure 7 : Outline diagram

The signature outline (figure 7), corresponding to the extreme vertical points, is interesting because, although it is made of much fewer data compared to the whole image, it keeps the global appearance of the signature ignoring local details. It cannot be used on its own because it is not informative for some signatures (figure 8). Moreover, the outline is quite sensitive to the orientation of the signature. In our system, we use the outline only for a first rough comparison.

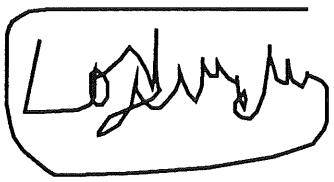

figure 8 : Signature with a non-informative outline

Geometrical parameters can be extracted either from the image or from the outline of the signature. They are interesting because they constitute a more concentrated information making them easier to manipulate and compare. Moreover, some geometrical parameters are invariant to rotation and magnification, which is particularly convenient. The main difficulty is to choose them so that they contain enough discriminative information and that they remain stable in spite of local or global distortions.

A significant study has been done by the SEPT (French Telecom research centre) [Revillet 91] to select twenty nine geometrical parameters, taking into account among others : size of the signature, orientation of the strokes and inertia moments. They constitute the most discriminant source of information in our system.

Getting inspiration from handwritten characters recognition, we could also consider using the number and the position of loops and intersections. But, for the European signature type, these characteristics are not stable as we can see on figure 9. Frequently a stroke underlines or crosses the signatures and the position of this stroke completely modifies the number of loops and intersections.

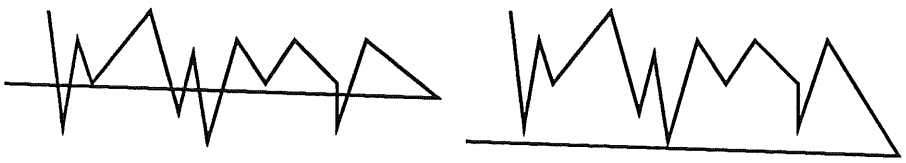

figure 9 : Instability of the number of loops

II.4. The database

Our system is a model of a real-scale system which should be able to work on 300 000 persons. In order to develop and test our neuronal architecture, a large representative database is needed. To constitute our database, 6000 checks have been digitalized.

The image of the checks is binarized by an automatic and adaptive thresholding method. For our application, the right lower quarter of the image, which contains the signature, is extracted. A rotation is applied so that the axis of inertia of the signature be horizontal, and a window is computed around the signature. The signature is often surcharged: banks are applying stamps on the checks, the machine that automatically fills in the check amount and the date does it sometimes on the signature, some signatures are written on the surrounding inscriptions (date, address, numbers), and finally, for legal reasons, all the checks we used have been stamped.

Looking at the signatures after these operations shows that this automatic pre-processing is not flawless : there are problems with some signatures : noise sensitive windows including parasitic information when the signature touches them, rotation sometimes too sensitive to signature deformations.

III. Description of the authentication module

III.1. Architecture of the authentication module

III.1.1. Structure of the authentication module

Our system is made up of three levels (figure 10). The first one is formed by two NNs of the non-supervised Kohonen map type. These two NNs use respectively as input, geometrical parameters and outline. Their function is to classify signatures belonging to the signatories of a same set (for instance : a postal check centre) into signature classes. The second level is also formed of two NNs, that are multi-layer networks using a error gradient backpropagation learning algorithm. One uses as input geometrical parameters and the other the image of the signature. These two NNs are specific to each signatory, they bring a major contribution to the authentication module decision. The third level is formed only by one NN of the backpropagation type. It takes as input the outputs of the previous NNs and takes the final decision whether to accept or reject the signature under examination.

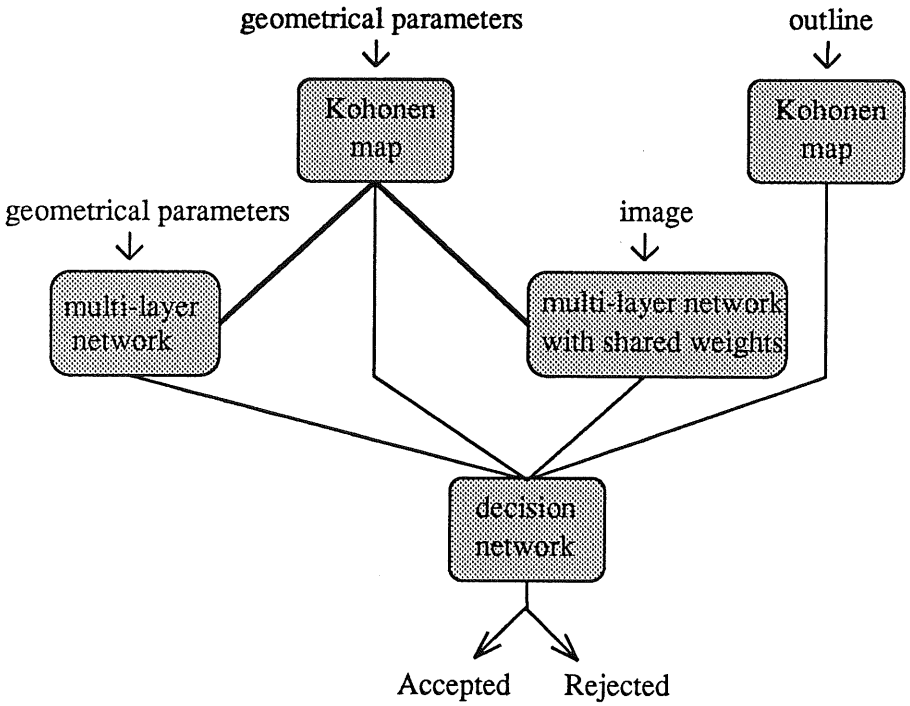

figure 10 : The three levels of the authentication module

The links represent the cooperation existing between the NNs. They are all used during the learning phase and only the links connected to the decision NN are also used during the exploitation phase.

III.1.2. Building the learning database

Some examples of genuines and forgeries must be shown to the two NNs of the second level during the learning phase. As these NNs are personal to each signatory, for the genuines we can use the reference signatures. But, for the forgeries we have to find some representative examples among the signatures of other persons. Several solutions can be considered :

A first solution consists in taking a few signatures from all the other persons. It is not a good one because the learning phase becomes too long when dealing with a real-scale database. Moreover, it would suppose to do again the learning for all the persons when a new signatory is added to the database.

A second solution consists in presenting to the NNs random values instead of the geometrical parameters of false signatures ; we have tested this solution because it simplifies the test protocols as the learning can be done independently from the rest of the database. But it gives poor results because NNs have to learn to differentiate the genuines in the whole space, the dimension of which equals the number of inputs to the NN. Actually, the whole space is much larger than the area where the genuine signatures can effectively be found, although they can be very different from one another and it is important to limit the inputs of the NNs to the existing signatures only.

In the third solution, we take as false signatures for a given person a sample of true signatures from other persons. These false signatures are chosen randomly among the reference signatures of the other signatories. Results are better than with the second solution but we noticed some differences when the random generator was initialised with different values. Globally, on all signatories, the results are about the same, but when analysing more precisely the results for a given person, significant variations can be noticed. We concluded that the choice of false signatures has some influence on the learning of the NNs, and that a random choice, although being a good solution, is certainly not the best one.

The question we have to answer is : which false signatures should be learned, for a given person, in order to optimize the leaning phase ? We noticed an improvement of the results when we reduced the space of signatures from randomly-made signatures to real ones. So we continued in this way and tried to reduce again the space of signatures used for learning: the idea was to present to the NNs, false signatures that resemble the genuine ones of a given person. Figure 11 summarizes how the whole space of representations was gradually reduced.

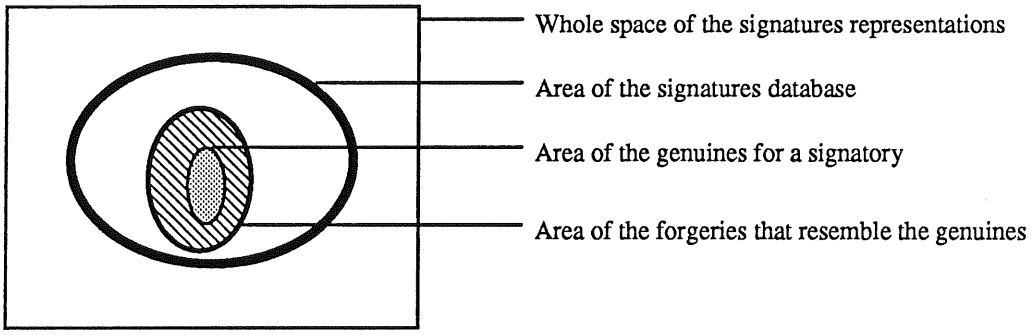

figure 11 : Splitting up of the representation space

To determine whether a false signature is close to a genuine signature of a given person, we use a Kohonen map ; it classifies signatures into a pre-defined number of classes. The false signatures used to learn the weights of the NN associated to each signatory are chosen among the false signatures that have been classified in the same class as most of the genuine signatures of that person.

III.1.3. Cooperation within the authentication module

Two types of cooperation [Boutkhil, Cardot et al. 92] are used in our authentication module :
The first one, of the "modulation" type, establishes cooperation between NNs of the first level and NNs of the second level. The former perform a rough classification of the signatures of the learning base, in order to improve the learning of the latter.
The second type of cooperation of the so-called "associative" type, is achieved by a third level which takes at its inputs the outputs of the NNs from the two first levels, and takes the decision whether to accept or reject the signature. Details about this NN are given in § III.4.

III.2. The first level of our authentication module

The goal of this layer is to perform a first classification, using the geometrical parameters and the outline of the signature. We built one NN for the parameters, and one for the outline. Both NNs work on the whole set of signatories, as we aim at classifying each signature in comparison with all the others.

As the number of signatories can be very large, it is impossible to have as many classes as signatories, which would have enabled supervised learning. So, we had to limit the number of classes of each NN (not necessarily the same number). In the case of a real-scale application, this number of classes is much smaller than the number of signatories : we could notice that several persons share the same class but a signatory can also be classified in more than one class.

We cannot a priori forecast which signatories are to be found in the same class. That is the reason why it is difficult to use a supervised learning method, in which we would tell the NN, for each reference signature, which class it is to choose. So, we decided for a type of NN enabling non-supervised learning. In that case, during the learning phase, the NN has to group into the same class all the signatures, it judges close to one another.

To implement this scheme, we chose the most current type of NN that enable unsupervised learning: the Kohonen auto-organisation network [Kohonen 84], which is now described. In our system, the Kohonen NN has as many input cells as the number of components of the data vector, i.e. 29 for geometrical parameters and 400 for the outline (400 equals twice the width of the outline). The number of output cells is chosen by experimentation. As the map is square, this number can be 16, 25, 36, 49 etc... The higher the number, the worst the classification rate of signatures into the class(es) of their signatory, but a good classification gives in that case a better information.

How the outputs of the Kohonen NNs are used as inputs for the third level must now be precised. To achieve this, we took the activation value of the node having the highest value, multiplied by a factor of 1 if the associated class is the most frequent among the reference signatures, 1/2 for the second most-frequent class, 1/3 for the next and so on, following the number of classes that can be associated to one signatory, and 0 for the other classes. So, we get a value that is really in proportion to the way the Kohonen map estimates the similarity of the signature with the classes associated to the signatory.

As we have seen it before, the classification done in this level helps for the choice of the forgeries used for the training of the NNs of the second level.

III.3. The second level of our authentication module

III.3.1. Multi-layer network working on the geometrical parameters

This network (fig. 12) has an input layer of 29 cells corresponding to the 29 geometrical parameters, an output layer with a unique cell and a variable number of hidden layers. The learning phase is done using the error gradient backpropagation algorithm [Rumelhart 86]. The connections between cells of a lower layer towards cells of the next higher layer are complete. The transfer function is a sigmoïd.

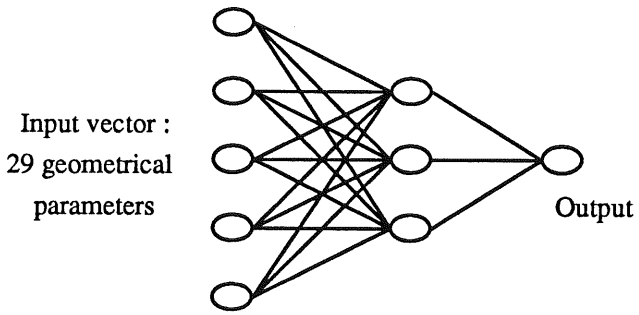

figure 12: The multi-layer network working with the geometrical parameters

The final structure of this NN was decided, following the results obtained when using it independently from the other NNs of the authentication module. We got the best results with one hidden layer. This can seem surprising when you know that hidden layers enable to delimit classes more precisely. Actually, hidden layers improve "by heart" learning of the signatures but diminish generalization capabilities, which is particularly interesting, for it enables the NN to get correct answers, even when it has never learnt the input signature.

III.3.2. Multi-layer shared weights NN (MSWNN) working on the image

In traditional multi-layer NNs, all the neurons of a layer are completely connected to the neurons of the next layer and at each connection is associated a weight. In the case of multi-layer shared weights NNs [Le Cun 89], we only keep local connections to a neuron (fig. 13) and we use shared weights, i.e. weights common to several connections.

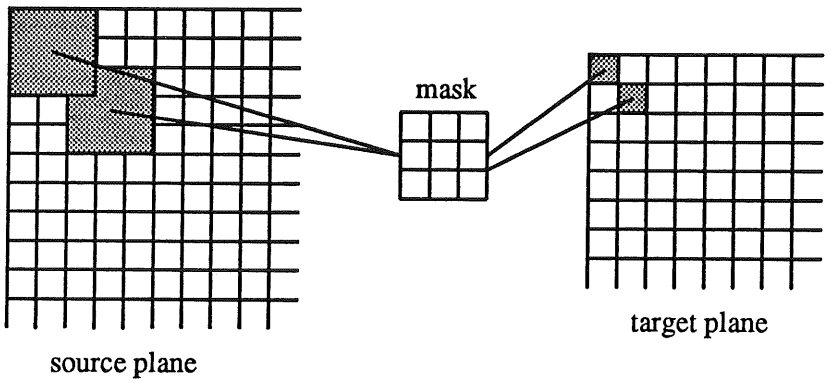

fig. 13: Principles of shared weights

To increase the NN efficiency, we used several planes for each layer (fig. 14). Each plane of a layer "sees" the same data and thus does the same work. Each plane has a mask, whose initial values of weights are chosen randomly at the beginning of the learning phase. The initial values of masks being different, each plane will converge towards a different "view" of the data, thus extracting different features.

Figure 14 shows the final structure of our MSWNN which contains 2 planes for the first hidden layer, i.e. 2 masks 3x3, and 5 planes for the second layer, i.e. 5 masks 5x5.

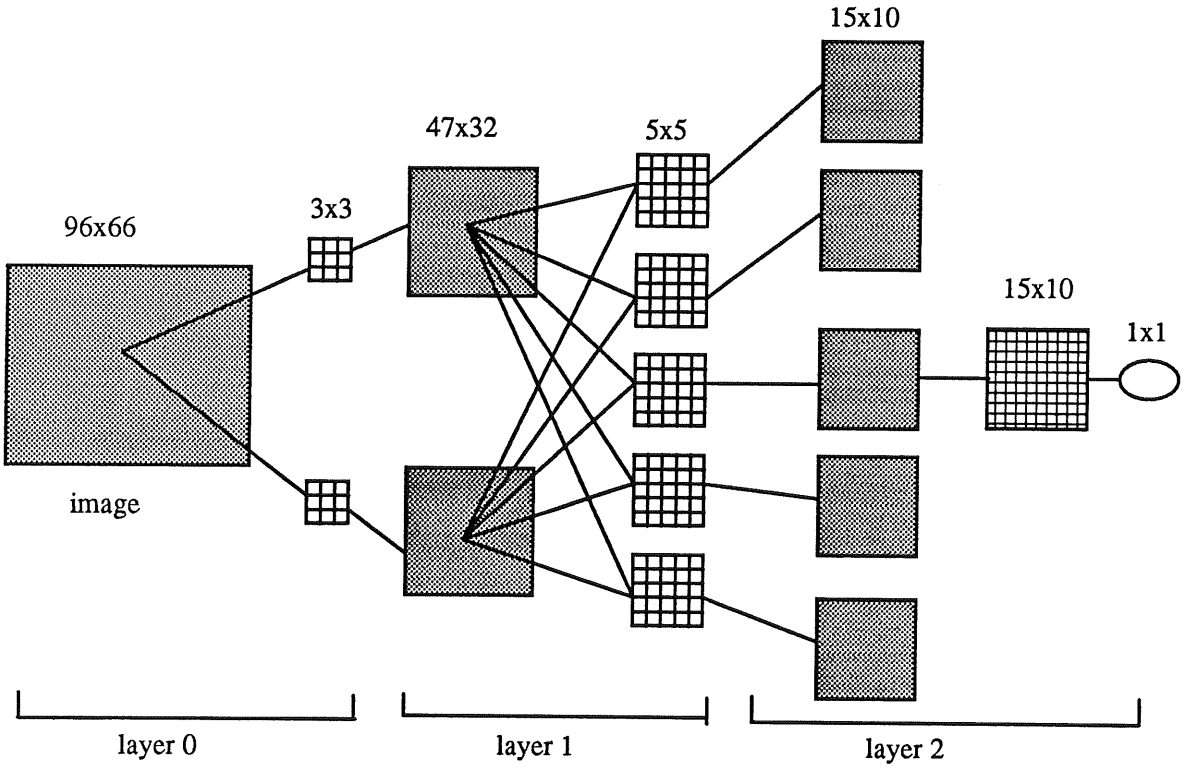

fig. 14: Architecture of our MSWNN

III.4. The third level of our authentication module

In a first experimentation [Cardot et al. 91], the decision phase only consisted in merging the boolean (true - false) outputs of each NN. Thanks to the size of our database (6000 signatures), we could then improve our decision strategy by introducing a multi-layer NN, whose output is a real number between 0 and 1, proportional to the similarity between the tested signatures and reference signatures. It comes to the same as to ask an expert to give his advice : he could answer something like: "I think the signature is not very resembling". Then, for the final decision, the answers of experts are weighted by a "confidence coefficient", corresponding to the fact that the answer of some experts plays a more important part in the final decision than the answer of others.

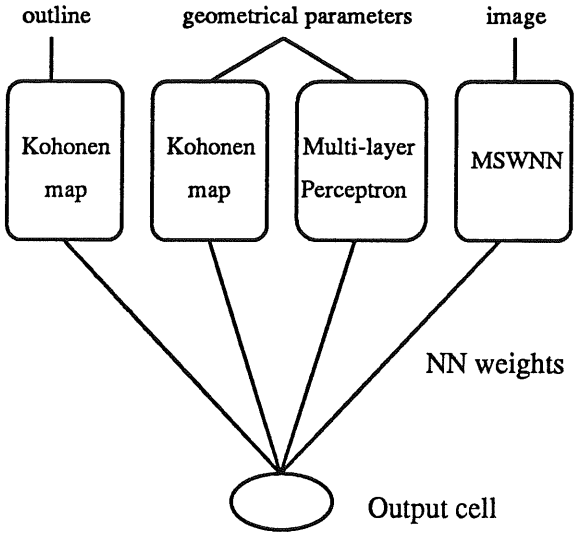

fig. 15: The decision NN

Figure 15 shows the fifth NN, which takes the final decision whether to accept or reject the signature. The weights of this NN are the same for all signatories. The NN takes as its inputs the outputs of the four former NNs and has only one output cell, the value of which is compared to a threshold, in order to take the decision.
Its has no hidden layer, and only 4 weights learnt by backpropagation on the all set of signatories. These weights can be interpreted as confidence coefficients towards the four former NNs.

Researchers are often deterred by using NNs because of the numerous parameters to tune. Here, our fifth NN is a means to avoid "manually" fixing confidence coefficient, leaving just the decision threshold to set. That threshold separates signatures judged as genuine from forgeries and is chosen in order to satisfy the initial objective, i.e. a 1% FAR.

Conclusion

We devised an architecture enabling the cooperation of various NNs using different representations of handwritten signatures : geometrical parameters, outline and image of the signature. A part of this architecture is specific to each signatory and another global to the set of signatories. Two types of cooperation are brought into play ; fusion of results obtained when working on the different representations of signatures, and improvement of the learning phase of multi-layer NNs, thanks to the selection of a sample of forgeries more closely resembling genuine signatures than randomly selected ones, thanks to a first classification performed by non-supervised Kohonen NNs.

Our research is directed towards an industrial application that can deal with millions of signatures from hundreds of thousands of signatories. This implies strong constraints such as the size of the data to be stored for each signatory ; our present system only needs about 300 values for each signatory, which is quite acceptable for a large-scale application.

Our results can be compared with others [Plamondon and Lorette 89], taking into account that our application has been developed and tested thanks to a large-scale database. This base was built by the digitalization of 6000 postal checks from more than 300 signatories. The use of real data and a large database enables to validate our system in almost normal conditions.

TVR	TFA
5 %	2 %

Results of the complete system

Finally, it will be possible to combine our authentication module with the module developed by the SEPT and based on statistical methods, because the decision level of our system can easily take into account results from other sources. The methods used by the SEPT being very different from our approach, we should improve the global performances of the system, although we use the same representations of signatures.

Acknowledgements

The study presented in this article was achieved in the ISMRA research team of the LAIAC laboratory (Laboratory of Algorithmics and Artificial Intelligence of Caen). This work was done thanks to a research contract between the ISMRA and the SEPT (France Telecom Research centre). It continues the thesis work done by F. Nouboud in 1988. Special thanks to Cristine Porquet who helped me with the English translation.

References

[Boutkhil, Cardot et al. 92] : L. Boutkhil, H. Cardot, F. Joublin, V. Lorquet, J.-D. Muller, O. Sarzeaud, S. Wacquant. "Structures et algorithmes neuromimétiques coopératifs pour la résolution de problèmes complexes". *Int. Conf. Neuro-Nîmes*, 1992.

[Cardot et al. 91] : H. Cardot, M. Revenu, B. Victorri, M.-J. Revillet. "Coopération de réseaux neuronaux pour l'authentification de signatures manuscrites". *Int. Conf. Neuro-Nîmes*, 1991.

[Kohonen 84] : T. Kohonen. "Self-Organization and Associative Memory". *Springer Series in Information Sciences*, vol 8, Springer Verlag, 1984.

[Le Cun 89] : Y. Le Cun. "Generalization and Network Design Strategies". *Connectionism in Perspective*, R. Pfeifer et al. (eds), Elsevier Science Publishers B.V. (North Holland), 1989.

[Plamondon and Lorette 89] : R. Plamondon, G. Lorette. "Automatic signature verification and writer identification - The state of the art". *Pattern Recognition*, Vol. 22, pp. 107-131, 1989.

[Revillet 91] : M.-J. Revillet. "Vérification de signatures sur chèques postaux". ICDAR 91, Saint-Malo, 30 sept. - 2 oct. 1991.

[Rumelhart 86] : D. Rumelhart, G. Hinton, R. Williams. "Learning representations by backpropagating errors". *Nature*, Vol. 323, No. 9, pp. 533-536, 1986.

[Sabourin 90] : R. Sabourin. "Une approche de type compréhension de scène appliquée au problème de la vérification automatique de l'identité par l'image de la signature manuscrite". *Thèse de l'Université de Montréal*, septembre 1990.