Neural Comput & Applic (2000)9:297–305

Neural
Computing
& Applications

A Set of Neural Tools for Human-Computer Interactions: Application to the Handwritten Character Recognition, and Visual Speech Recognition Problems

G. Vaucher, A. R. Baig and R. Séguier

SUPÉLEC, Signal Processing and Neural Networks Team, Cesson Sévigné, France

This paper presents a new technique of data coding and an associated set of homogenous processing tools for the development of Human Computer Interactions (HCI). The proposed technique facilitates the fusion of different sensorial modalities and simplifies the implementations. The coding takes into account the spatio-temporal nature of the signals to be processed in the framework of a sparse representation of data. Neural networks adapted to such a representation of data are proposed to perform the recognition tasks. Their development is illustrated by two examples: one of on-line handwritten character recognition; and the other of visual speech recognition.

Keywords: Human-machine interaction; Lipreading; On-line handwritten character recognition; Spatio-temporal coding; Spatio-temporal neural networks; Spatio-temporal patterns; Spiking neurons; Visual speech recognition

1. Introduction

The present means of communication between the humans and machines are more and more questioned and criticised. The pocket computer, which has a certain interest for the general public, has sizes such that the utilisation of a keyboard or mouse is unsuitable. Moreover, a broader expansion of the use of computers is harnessed by the fact that they impose the burden of understand and learning the functioning of the tool on the user. Ideally, it should be the task of the computer to adapt itself to the user.

Realisation of HCI, which permits someone to command all the power of a machine while using means of communication which are more natural, is a challenge which occupies a large community of scientists and technicians. What we propose is to deal with the problem by adopting a unified approach at the level of the shaping of the data, as well as at the level of its processing.

After having identified many characteristics common to HCI, this paper presents a coding and a mode of homogenised processing, adapted to the signals thus characterised. To illustrate this approach, two examples of applications are presented.

Correspondence and offprint requests to: Correspondence and offprint requests to: G. Vaucher, Supélec, Signal Processing and Neural Networks Team, B.P. 28, 35511 Cesson Sévigné cedex, France. Email: gilles.vaucher@supelec.fr

2. User Interfaces

2.1. Characteristics of Signals in HCI

Variety of signals to be processed. At present there exist many means of operational inputs for achieving an intuitive HCI. For example, the stylus for writing and drawing, the microphone for speech and sound, the touch screen and the TouchPad for the sense of touch, and video for the interpretation of gestures, recognition of facial expression, and analysis of the movements of the eyes, head and lips.

Characteristics of the signals. The main characteristic which all these signals have in common is the

enormous quantity of information they carry. Video is for the moment the leader in generating high dimensional data, but it is progressively being joined by the other systems of acquisition due to the development of micro-technologies. As an example, we can cite graphic tablets which are now capable of producing the level of pressure of the stylus, and even, for more elaborate versions, its speed, acceleration and orientation.

The second characteristic shared by these types of signals reside in the fact that the relevant information is contained in the correlation between the spatial and temporal information. These correlations must be done and analysed to permit a machine to extract the essence of the user's message. These Spatio-Temporal Patterns (STP) may be simple: like the case of a rigid object (spatial correlation) which moves in space (temporal correlation of the object throughout the duration of a sequence of images). These may equally be more elaborate: the raising of an eyebrow or a smile are gestures which may be interpreted as spatial forms which evolve over time.

Moreover, it is often necessary to produce a fusion of different sources of signals. Such an operation brings an improvement in robustness; for instance, the speech recognition is improved when we enhance the audio modality by the video signal of the lips' movement. This fusion is sometimes indispensable for understanding the complete message, for example in the case of an operator giving a command of the type *put the object there* to a system equipped with a microphone and a touch screen.

2.2. Coding and Homogenous Processing

Bio-inspiration. It is natural, in the framework of HCI, to investigate the manner in which our brain processes the different sensorial modalities. In simple terms, we may say that it seems the sensors (ears, eyes, skin) convert the set of acquired information into electrical impulses (isolated impulses, train of impulses, etc.).

Moreover, after transcoding, it seems that, independently of the origin of the signal, the same type of units process the information: the *neuron*, if we may permit ourselves to group under the same name all the neuronal cells which may have very different properties.

Our proposition: A unified approach for the coding and processing of information. The common characteristics of the processed signals in HCI naturally point towards a common processing. For this reason, and also because the brain seems to adopt the same strategy, we propose to follow a unified approach, both at the level of the coding of the information as well as in its processing.

The sparse coding [1] could be the means utilised by the brain for analysing and memorising the information in a fashion which is robust and efficient [2], and much work has been done in developing algorithms which permit the coding of signals in such a manner [3,4]; links with Independent Component Analyses and wavelets have been highlighted [5]. The sparse coding is therefore the coding which we have retained to work on.

Following such a coding of the signals, we have developed a set of neural tools which permit recognition of patterns containing spatial and temporal information. This approach, which unifies, with the help of a common representation, the data obtained from different sensors (microphone, video camera, stylus touch screen) makes their fusion easier. Moreover, it simplifies the implementations. It leads to a perspective of the development of the system under the form of a limited program code, functioning in real time; which is essential for certain systems (eg. PDAs: Personal Digital Assistants).

3. STANN

The tools we are proposing make up a set of neural networks [6–11]. We called them STANN (Spatio-Temporal Artificial Neural Networks). Their main characteristic is the processing of STP expressed in the form of impulses. This set of tools belongs to the domain of work done by several teams for the introduction of a time factor in the neural networks field [8,10,12,13]. The STANN combines the potential of classical artificial neural networks with the assets of spiking neurons, thus combining the algebraic properties of the McCulloch & Pitts neuron and the capabilities of the biological model of Rall for the recognition of elementary sequences [14].

3.1. Coding of Information

The processed information is represented by sequences of impulses. It is presented to each calculating unit in the form of a vector whose components are complex numbers. The process of converting a train of impulses into vectors is done in the following manner: each impulse I is characterised by its amplitude a_{I_0} and the temporal delay d_I which separates the current instant of time from the date at which the impulse was received by the calculating

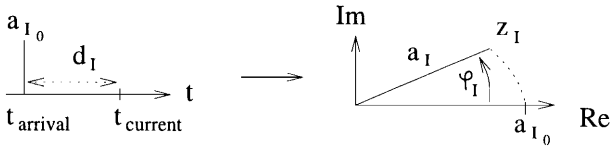

Fig. 1. Impulse coding with complex numbers.

unit. A complex number z_I is associated with the impulse I. It contains the delay d_I as its phase φ_I, and the current amplitude a_I as its module. a_I is determined from the initial amplitude of the impulse a_{I_0} by introducing an attenuation with the passage of time (short term memory) as shown in Fig. 1:

$$z_I = a_I e^{i\varphi_I}$$

with

$$\begin{cases} a_I = a_{I_0}\, e^{-\mu_S d_I} \\ \varphi_I = \arctan(\mu_T d_I) \end{cases}$$

where μ_S and μ_T are two constants which permits us to control the characteristics of the short-term memory of the model.

The composition of the impulses, at the level of each unit, is done input by input such that it becomes a complex vector whose number of components is equal to the number of inputs (Fig. 2). A sequence of impulses is thus coded by a vector in C^n, which is provided with the canonical hermitian metric, therefore we can calculate the distance between sequences.

3.2. Units and Architectures

Several types of units have been developed. The simplest are composed of a single neuron, and the others are constituted of bigger networks. These are all autonomous units which communicate in an asynchronous manner with each other by means of exchanges of impulses. The units have a continuous dynamic; however, taking into account the impulse character of the exchanges between the units, an event-driven implementation seems to be more efficient. The training algorithms specific to each of these types of units have been developed.

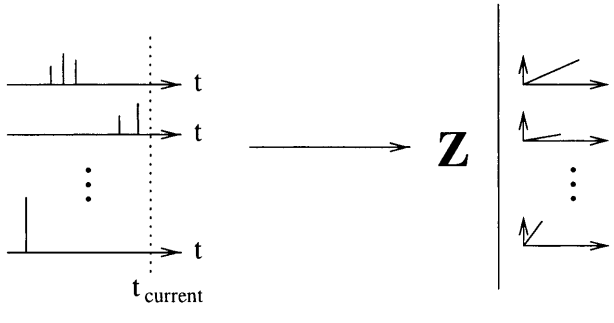

Fig. 2. A sequence of impulses is coded as a complex vector.

Simple units. Two types of simple units have been developed to-date. They are composed of a unique neuron which at each instant of time compares the vector coding the input sequence (short-term memory) with a weight vector (long-term memory). The weight vector is also complex valued, and it codes the sequence which the neuron must recognise. Two types of comparison between sequences (from which the two different types of units ensue) are retained; one is based on the hermitian scalar product

$$v = X^T \overline{W}$$

X being the input sequence and W the weight sequence. The other is based on the calculation of hermitian distance between the two sequences:

$$\begin{aligned} \mathrm{d}(X,W) &= \|X - W\| \\ &= \sqrt{(X-W)^T\, \overline{(X-W)}} \end{aligned}$$

In the first case, we get from the phase of the potential v the date at which the unit may produce an output, and from the module we obtain the amplitude of the output. This module may be compared, if desired, with a threshold.

In the second case, an impulse of unit amplitude is produced when the distance d becomes inferior to a given distance.

The simple units in network. Several architectures of networks were studied using the simple units presented above. They are derived from classical architectures in the domain of artificial neural networks. They are:

1. ST-WTA [15]: this is a Spatio-Temporal (ST) adaptation of the Winner Takes All model, for which the training is done in an analytic manner by imposing on each weight vector of each unit the following constraints:
   - It should correspond to the prototype sequence to be recognised.
   - It should be normalised and orthogonalised, in the output space, with all the other weight vectors of the network.
2. ST-Kmeans [6,7]: this is a transposition of the K-means algorithm [16–19] to ST coding in the complex domain, thus permitting an unsupervised vector quantification in the space of sequences

to be done. The training leads to a partitioning of the space of input sequences into regions, each being represented by a vector of reference.
3. ST-RCE [6,7]: this is a supervised model in which the first of the two layers of the evolutive architecture of Reilly, Cooper and Elbaum [20] divides the space of the input sequences by hyperspheres; the second layer regroups the preceding hyperspheres by class in order to fulfil the tasks of recognition.

Large units. Units containing several neurons in a network have also been conceived. They profit from the possibilities of non-linear separation offered by the classical artificial neural networks. Each of the neural parameters is a complex value, with the ST interpretation given above. However, here the interactions between the internal neurons are purely analytic. Only the inputs of the unit are furnished with impulses (short-term memory at the level of inputs); the outputs generate impulses as a function of the thresholds of decision. For some applications (e.g. classification), a process of decision of type WTA is added at the level of the output, so that only one of them produces an impulse during the recognition task. Here again, several classical architectures were transposed for a ST exploitation of data:

1. ST-MLP [9,21]: this model derives from a version extended to complex numbers of the Multilayer Perceptron [22], for which the back-propagation algorithm stays valid by taking certain precautions.
2. ST-RBF [8]: it is conceived from the commun Radial Basis Functions (RBF) model, but operates in an input hermitian space; the second layer of neurons works with real or complex values, according to the application (classification or function approximation).
3. ST-Kohonen [23]: this is a ST version of the self-organising map of Kohonen, which makes possible the extraction of topological links in a sequence space.

3.3. Application

For a given task of HCI recognition, once the raw signals have been processed by the sparse coder, the systems which we use are formed of two classical steps: the first ensures the extraction of characteristics; the second is charged with the task of classification. The extraction step is aimed at making the task of the classifier easier and more robust. The classifier associates a label to a sequence of impulses. For example, in online handwritten character recognition, a chain of elementary lines is recognised as the traced letter.

Extraction of characteristics. The procedure consists of analysing the sequences to be recognised as a train of elementary sub-sequences. Hence, this step aims at extracting an alphabet of sub-sequences with which it is possible to compose the global sequences (Fig. 3). The extraction may eventually be done in several stages; certain chains of sub-sequences hence constituting a dictionary which serves as the basis for the elaboration of a meta-dictionary (and so on), until we are able to get the list of elements which act as the basis for the classification of the global sequences to be recognised.

For each level of the decomposition, a STANN (composed of a single layer of ST neurons) ensures the automatic extraction, by unsupervised training, of the sequences of elementary impulses representative of a certain temporal window *TW*.[1] During relaxation, when a sequence is recognised by a neuron, it produces an impulse. The succession of impulses produced by the STANN is at its turn processed by the following STANN, which works on a temporal window *TW′* of a duration larger than the *TW*, and so on until the classification of the global sequences may be done. The STANNs used here are the ST-Kmeans and ST-Kohonen, according to the application. Usually, they are both tried and the best combination is retained.

Classification. Once a sequence is broken into sub-sequences, the next step is to associate their corresponding class to the sequences of impulses produced by the last STANN of the previous step.

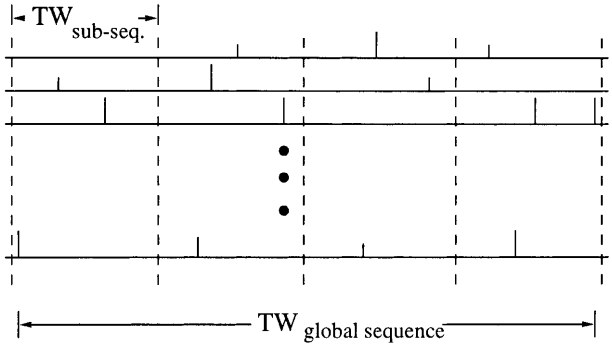

Fig. 3. Global sequences are analysed as compositions of elementary sub-sequences.

[1] *TW* corresponds to the duration of the short-term memory of the neurons. A precise definition is given in Baig [24].

Another STANN is therefore added to the system. It is a model functioning with a supervised training. Both ST-RCE and ST-MLP should be tried. The performance in terms of recognition and implementation will decide which is the best.

During elaboration of the system, the improvement in performance of the implementation is done by limiting the number of layers in the extraction of characteristics step. Moreover, we sometimes need to compare the classification and implementation performance of the following approaches:

- Experiments with none or at least one layer of extraction of characteristics in step 1 and a ST-MLP in step 2.
- Experiments based on a system having several layers of extraction of characteristics, followed by a simple ST-RCE at the output.

4. Examples

To illustrate the technique which we have presented above, two example applications are proposed. The first concerns online handwritten character recognition, and the second is about visual speech recognition, commonly known as lipreading.

4.1. On-line Handwritten Character Recognition

The drawing of a character is done with a stylus on a digitising tablet, which sends a series of elementary displacements to the computer. This series is first translated in the form of a vectored sequence of impulses. Then, the recognition task is done in two steps: (1) an initial layer of ST neurons, which have the task of detecting certain primitives (lines) in the strokes; and (2) a ST-MLP which recognises the characters. While proceeding thus, we get rid of all kinds of preprocessing which are necessary for most classic connectionist methods: filtering, normalisation, resampling, etc. [25,26].

The digitising tablet provides us with coordinates of the elementary displacement of the stylus. Quite a few authors transform these stylus elementary displacement values into a movement in a given direction out of a limited set. We have adopted the same technique, which is inspired from the 8-topology of Freeman [27], but we made it simpler in translating every displacement of the stylus into a couple of movements, each corresponding to one of the four basic directions (Fig. 4). Taking these movements as impulses, the stroke of a character produces a train of impulses, each one having an orientation (left, right, up, down), an amplitude (absolute value of the projected movement) and a date (at which the elementary displacement occurs) (Fig. 5).

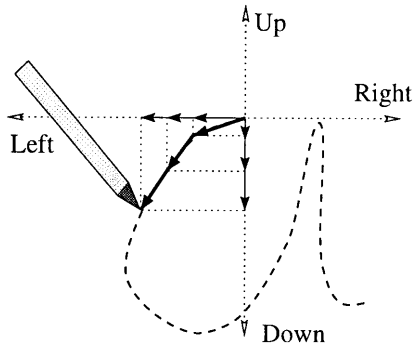

Fig. 4. Decomposition in four directions.

The first layer is made up of simple ST neurons, of the first type, presented in Section 3.2 (*Simple units*). They have the role of accumulating spikes in the same direction. If the potential of a given neuron accumulates enough successive spikes, and attains a certain threshold, the neuron produces a spike at its output. This impulse represents the identification in the stroke of a character of a line with a certain length in a given direction. Since every neuron can detect only a particular length, we have used three neurons (with different thresholds) for any direction to be able to detect three different line lengths in all directions (Fig. 6). This separation between small and large lines facilitates the recognition task by a better identification of the local characteristics of the characters (it is so, for example, in the case of loops: Fig. 7).

The second module has to recognise one of the 26 characters of the alphabet from the corresponding spikes of identified lines in the stroke of the character drawn. In this step, a ST-MLP is used (Fig. 8). To do the training and the tests, we used a multi-scripter database, developed by the IRISA[2], made up of the handwriting of 14 people.

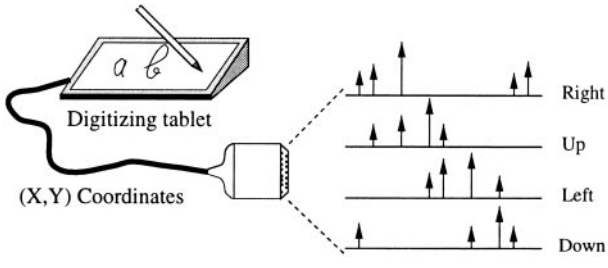

Fig. 5. Movement acquisition in the form of spikes.

[2] Institut de Recherche en Informatique et Systèmes Aléatoires, Rennes (35), France.

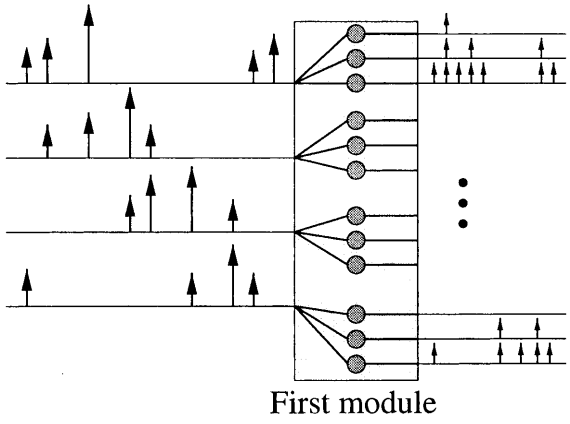

Fig. 6. Identification of basic lines.

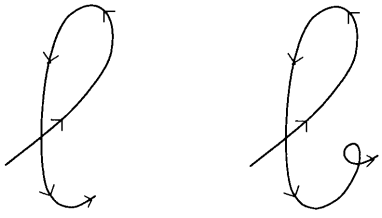

Fig. 7. Importance of loops in the recognition task.

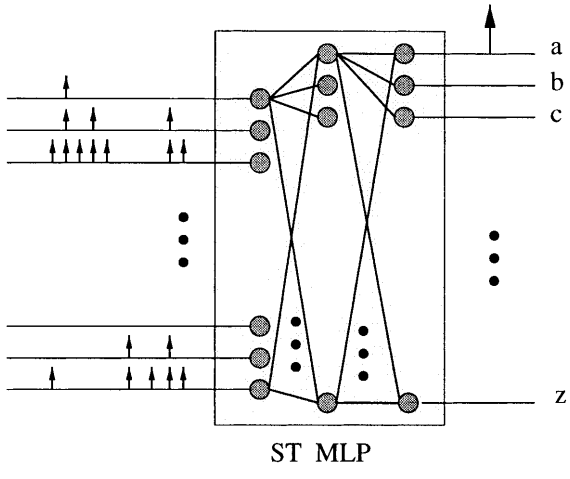

Fig. 8. Recognition of characters.

In this phase of the test, the quality of recognition is good but inhomogeneous (between 62% and 100% of recognition, according to the letters in question). The confusion observed between certain letters, like *h* and *k* or *m* and *n*, corresponds to that made by the human expert, and it can only be solved by adding contextual information. The recognition of words is thus the goal we are currently working on.

For more details on this application, see Mozayyani [8] or Mozayyani and Vaucher [21].

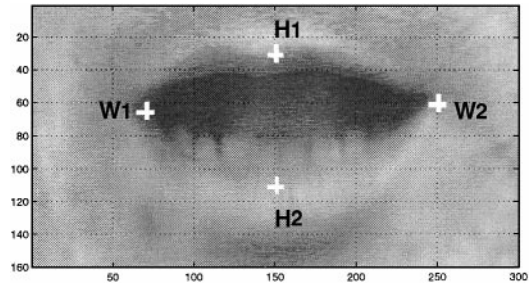

Fig. 9. A grey level image.

4.2. Visual Speech Recognition

We have also applied our approach to a Visual Speech Recognition (VSR) task of French digit recognition. In a continuous effort to built better man-machine interfaces, many good audio speech recognition systems have been developed in the past two decades. However, their performance decreases rapidly in the presence of noise and cross-talk. To overcome this defect, researchers have been exploring the incorporation of a visual signal. Some of the VSR systems developed to-date are discussed elsewhere [28–32]. Most of them are either intrusive (colouring lips, head mounted cameras [29], marker points on the face [32], etc.), or need quite heavy preprocessing.

Our system consists of a preprocessing module and two ST neural architectures in cascade. The preprocessing module implements simple and low CPU cost techniques to automatically follow the movement of certain points on the lips of a speaker. From the movement of these points, which is ST by nature, speech recognition is carried out by the neural architectures without using the audio modality.

Images of the mouth region are acquired by a video system (Fig. 9). The images are taken in a common office room, in natural lighting conditions. The RGB colour images from the video camera are projected into YUV colour coordinates [33]. For further processing, we use only the coordinate V, because the contrast between the lips and the mouth interior in this coordinate is quite large (Fig. 10).

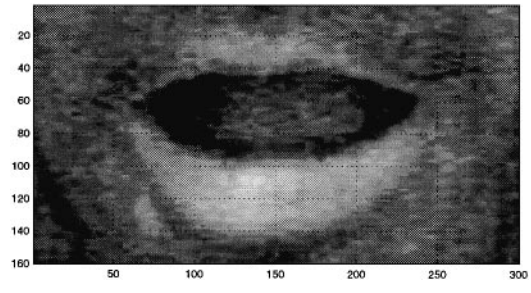

Fig. 10. Image of V coordinate in the YUV colour space.

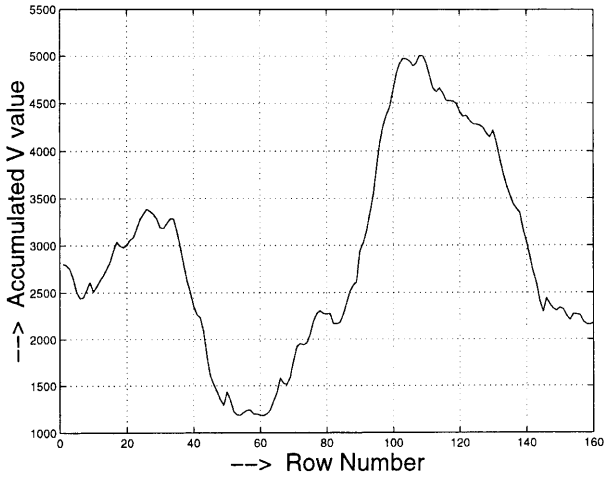

Fig. 11. Accumulated V values for each row.

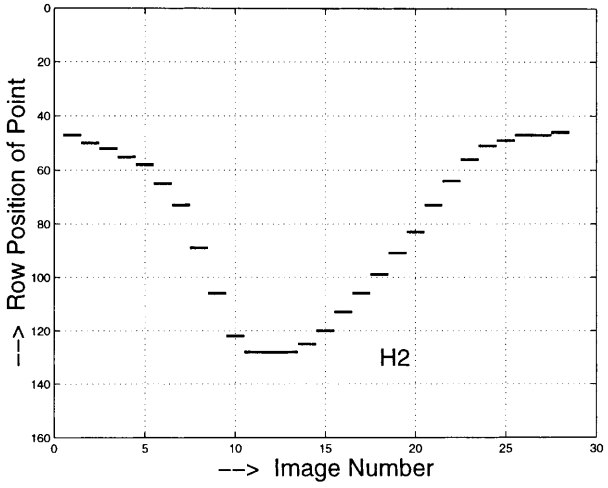

Fig. 13. Trace of the point H2 for a complete sequence.

As visual parameters, relevant for tracking the lips' movement, we have chosen four points on the lips. For finding the position of these points we make an accumulation of the V values for each row and each column of the image. The points H1 and H2 (resp. W1 and W2) are localised on the negative and positive slopes of the accumulation curve of the rows on the y-axis (resp. columns on the x-axis), hence we pass a derivative filter on these curves, and search for the minimum and maximum points (Figs 11 and 12).

The coordinates of H1 and H2 (resp. W1 and W2) on the y-axis (resp. x-axis) are traced throughout the sequence of images (Fig. 13), and the information in these sequences is converted into impulse vectors. For this purpose, only the strongest rate of changes in the two opposite directions are retained for each point tracked. A derivative filter applied on the curve permits us to generate two signals of impulses per point (Fig. 14). This gives us an eight-component impulse vector for each image series, which serves as input to our STANN (Fig. 15).

For the experiments, we have chosen a digit recognition task with French digits: *Zéro*, *Un*, etc. A spoken digit contains one or more basic lip movements (sub-sequences of a complete sequence), and differs from another spoken digit in the order and presence (or absence) of these sub-sequences. Accordingly, we have broken the classification process of the impulse vector sequences into two parts. In the first part, we determine the sub-sequences present in a complete sequence. In the second part, we classify the complete sequence according to the presence in the right order of these sub-sequences (Fig. 16). The impulse vectors, produced by the preprocessing subsystem, are the input of the first STANN. The simple ST artificial neurons of this module are of the second type given in subsection

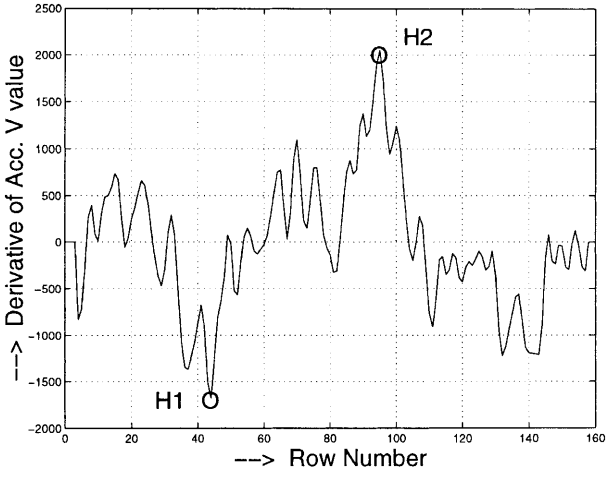

Fig. 12. The accumulation curve after the derivate filter.

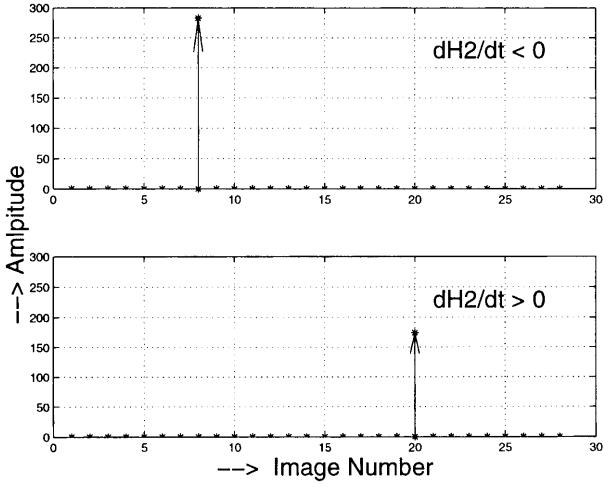

Fig. 14. Impulse plot of H2 for a complete sequence.

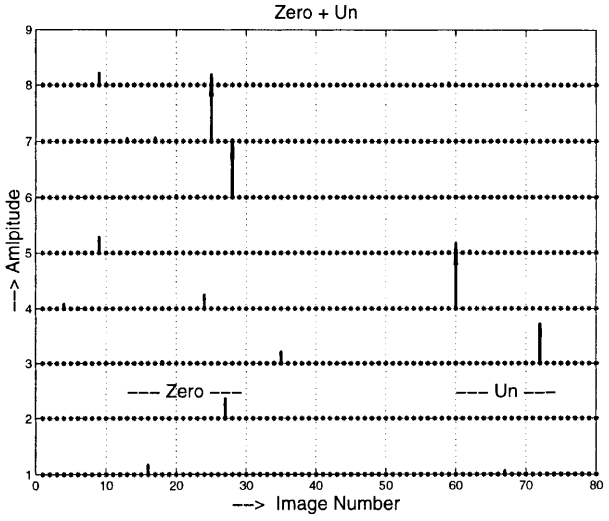

Fig. 15. Two consecutive sequences: *Zéro – Un*.

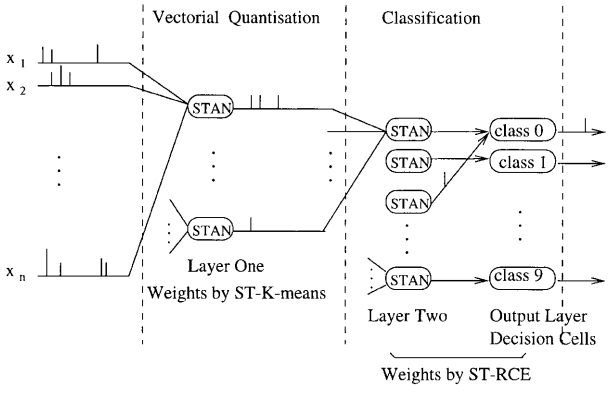

Fig. 16. Two consecutive Spatio-Temporal Artificial Neural Networks.

3.2 (*Simple units*). They have their weights and thresholds set by the ST-Kmeans technique.

The second module is an ST-RCE architecture. The number of neurons required and their weights and thresholds are determined by training, according to an adaptation of the RCE algorithm [20]. The output comprises 10 units, one for each digit.

On the recordings of one speaker (male, white, beardless), the best results obtained in recognition are 77.6% correct on our testing base. Among the video-only results reported by others, there is none in similar conditions which is more than 60% correct. In Luettin [30], the result reported on the same task as ours is around 58%, but on a multi-speaker database. So, our results are quite encouraging, even if a real benchmarking was not followed, since there is no such publicly available database.

In the future, we intend to extend our VSR system to a multi-speaker task. We also wish to make it completely autonomous by incorporating a face [34] and lip-tracking module; our purpose is to have a global system functioning in real-time.

For more details on this application, see Baig et al. [6,7,24].

2. Conclusion

Following a bio-inspired approach, we have developed a family of models of neural networks which processes STPs coded in a unified way. The interest in such models is shown on two applications: first on a handwritten character recognition task; and then on a lipreading problem, both within the context of HCI.

The performances obtained by our handwritten recognition system in a task of isolated character analysis are comparable with those of a human operator (cf. end of Section 4.1). The absence of rather heavy preprocessing, as used in other works [35,36] for respectively removing tremors and duplicated points in the first case, and adding intermediate points while oversampling in the second case, reduces the size of the system, which is an asset for integration into a system such as a PDA.

In the absence of any benchmark in the field of lipreading, in Section 4.2 we compared the 77% recognition rate of our system to the 58% rate of Luettin [30]. This system, which was selected because it appeared to us to be the closest to ours at the level of the task realised,[3] uses Hidden Markov Models (HMM) to carry out the recognition, which leads to rather heavy processing overhead. In our case, a few seconds on a PC (Pentium III–550 MHz) is enough for the complete chain from acquisition to classification to carry out the whole process, even though the code is not optimised.

Consequently, beyond the rates of recognition obtained by the two systems presented above,[4] we would wish to underline, in the context of real-time processing, the performance of the proposed spiking systems; and to highlight the potentialities offered by such systems as regards multi-sensory fusion for the development of multi-modal man-machine interfaces. To illustrate this, we are currently working on voice recognition system based on both audio and video signals. Each source would be converted into the form of impulses, first processed separately and then jointly by a multi-network architecture

[3] An inventory of the performance of 39 systems is proposed in Baig [24].

[4] Since our experiments in these application domains are still in progress, we expect to be able to improve the capacities of the systems presented.

made up of STANNs. Finally, with an aim of having an environment adapted to implement these architectures in real-time, the development of a dedicated dispatcher of impulses is in progress; it will ensure an efficient scheduling in an asynchronous mode of the autonomous units which are STANNs.

Acknowledgements. This work was carried out at *Supélec*, Rennes (35), France. We would like to thank Nasser Mozayyani for his contribution in the conception of the on-line handwritten character recognition system presented above.

References

1. Foldiák P, Young MP. Sparse coding in the primate cortex. In: Arbib MA (ed), The Handbook of Brain Theory and Neural Networks. MIT Press, 1995; 895–898
2. Field DJ. Visual coding, redundancy, and feature detection. In: Arbib MA (ed), The Handbook of Brain Theory and Neural Networks. MIT Press, 1995; 1012–1016
3. Lappalainen H. A computationally efficient algorithm for finding sparse codes. http://www.cis.hut.fi/harri/dityo/dityo.html, May 1996
4. Meunier C, Nadal JP. Sparsely coded neural networks. In: Arbib MA (ed), The Handbook of Brain Theory and Neural Networks. MIT Press, 1995; 899–901
5. Hyvarinen A, Oja E, Hoyer P, Hurri J. Image feature extraction by sparse coding and independent component analysis. Int Conf on Pattern Recognition (ICPR'98) 1998; 1268–1273
6. Baig AR, Séguier R, Vaucher G. Image sequence analysis using a spatio-temporal coding for automatic lipreading. Int Conf on Image Analysis and Process September 1999
7. Baig AR, Séguier R, Vaucher G. A spatio-temporal neural network applied to visual speech recognition. Int Conf on Artificial Neural Networks September 1999
8. Mozayyani N. Introduction d'un codage spatio-temporel dans les architectures classiques de réseaux de neurones artificiels. PhD thesis, Supélec, July 1998
9. Mozayyani N, Baig AR, Vaucher G. A fully-neural solution for on-line handwritten character recognition. IEEE Int Joint Conf on Neural Networks. Alaska, May 1998
10. Vaucher G. A la recherche d'une algèbre neuronale spatio-temporelle. PhD thesis, Supélec, 1996
11. Vaucher G. An algebra for recognition of spatio-temporal forms. Euro Symposium on Artificial Neural Networks April 1997; 231–236
12. Chappelier JC, Grumbach A. Time in neural networks. SIGART Bulletin, July 1994; 5(3): 3–11
13. Maass W, Bishop C. Pulsed Neural Networks. MIT Press, 1998
14. Rall W. Core conductor theory and cable properties. In: Geiger SR, Kandel ER, Brookhart JM, Mountcastle VB (eds), Handbook of Physiology: The Nervous System. American Physiological Society, 1977; 1: 39–97
15. Vaucher G. An algebraic interpretation of PSP composition. BioSystems 1998; 48(1–3): 241–246
16. Duda R, Hart P. Pattern Classification and Scene Analysis. Wiley, New York, 1973
17. Gersho A, Gray R. Vector Quantisation and Signal Compression. Kluwer, 1992
18. Lebart L, Morineau A, Piron M. Statistique exploratoire multidimensionelle. Dunod, 1997
19. MacQueen J. Some methods for classification and analysis of multivariate observations. Proc 5th Berkeley Symposium on Math Stat and Prob 1967; 1
20. Reilly D, Cooper L, Elbaum C. A neural model for category learning. Biological Cybernetics 1982; 45
21. Mozayyani N, Vaucher G. A spatio-temporal perceptron for on-line handwritten character recognition. Int Conf on Artificial Neural Networks October 1997; 325–330
22. Masters T. Signal and Image Processing with Neural Networks. Wiley, 1994
23. Mozayyani N, Alanou V, Dreyfus J-F, Vaucher G. A spatio-temporal data-coding applied to kohonen maps. Int Conf on Artificial Neural Networks. EC2 et Compagnie, October 1995; 2: 75–79
24. Baig AR. Une approache méthodologique de l'utilisation des STAN appliquée à la reconnaissance visuelle de la parole. PhD thesis, Supélec, April 2000
25. Lecolinet E, Baret O. Cursive word recognition: Methods and strategies. In: Impedovos (ed), Fundamentals in Handwriting Recognition. Vol 124 of F: Computer and Systems Sciences. Springer-Verlag, 1993; 235–263
26. Manke S, Finke M, Waibel A. NPen^{++}: A writer independent, large vocabulary on-line cursive handwritten recognition system. ICDAR, Third Int Conf on Document Analysis and Recognition 1995; 1: 403–408
27. Belaïd A, Belaïd Y. Reconnaissance des formes: Méthodes et Application. InterEdition, 1992
28. Duchnowski P. See me, hear me: Integrating automatic speech recognition and lipreading. Proc of Int Conf on Spoken Language Processing 1994.
29. Goldschen A. Continuous automatic speech recognition by lipreading. PhD Dissertation, George Washington University, 1993
30. Luettin J. Visual speech and speaker recognition. PhD Dissertation, University of Sheffield, 1997
31. Petajan ED. Automatic lipreading to enhance speech recognition. Proc IEEE Communications Society Global Telecom Conf 1984
32. Stork D, Wolff G, Levine E. Neural network lipreading system for improved speech recognition. Int Joint Conf on Neural Networks 1992
33. Pratt WK. Digital Image Processing. Wiley, 1991
34. Séguier R, LeGlaunec A, Loriferne B. Human faces detection and tracking in video sequence. Proc 7th Portuguese Conf on Pattern Recognition 1995
35. Alimi A, Ghorbel O. The analysis of error in an on-line recognition system of Arabic handwritten characters. ICDAR, Third International Conference on Document Analysis and Recognition 1995: 890–893
36. Schwenk H, Prévost L, Milgram M. Comparaison de la Distance Elastique et de la Distance Tangente en Reconnaissance de Caractères 'on line'. 10^{e} congrès RFIA, Reconnaissance des Formes et Intelligence Artificielle January 1996; 589–596