Intelligent Automation and Soft Computing, Vol. 7, No. 3, pp. 161-162, 1998

ON-LINE CHARACTER ANALYSIS AND RECOGNITION WITH FUZZY NEURAL NETWORKS

E. Gómez Sánchez*, Y.A. Dimitriadis*, M. Sánchez-Reyes Más*, P. Sánchez García*, J.M. Cano Izquierdo, J. López Coronado****

**Department of Signal Theory, Telecommunications and Telematics Engineering,*
School of Telecommunications Engineering, University of Valladolid
Camino Viejo del Cementerio s/n, Valladolid, 47011, Spain
***Department of Systems Engineering and Automatic Control,*
School of Industrial Engineering, University of Cartagena
Campus Muralla del Mar, 30202, Spain

ABSTRACT—A new recognition system based on a neuro-fuzzy system, called FasArt, is proposed in this paper. Satisfactory results were obtained using the *train_r01_v02* UNIPEN dataset, together with a comparison with the recognition rates achieved by independent human testers. Two methods for segmenting handwritten components into strokes are proposed, with better experimental results for the method based on biological models of handwriting, in terms of consistency and network complexity. A systematic experimental study of different codification schemes is also described, based on Shannon entropy and clustering maps. Finally, some steps towards the construction of an allograph lexicon are shown, that exploit the generation of fuzzy-rules by FasArt architecture.

Key Words: FasArt, neuro-fuzzy systems, UNIPEN, handwriting segmentation, codification and recognition, pre-allographs

1. INTRODUCTION

Handwriting has been studied for a long time, because of its widespread use as a means for human-to-human communication. Its generation and recognition aspects served as a paradigm in order to study models and generic techniques, besides the need to solve the specific engineering problem. As an example, we mention the studies on the human motor system, that use handwriting as an exemplar of fine movements [21]. On the other hand, its variability, due to differences in cultures, writer styles, alphabets or fonts, make it an interesting problem in the field of pattern recognition [27]. Its two-way relationship with the humans (generation and perception) requires an involvement of knowledge and techniques, more related to psychology, biology, and neural modeling. Therefore, a fusion of work in handwriting generation models, neural and fuzzy systems can be of help for the study of this problem.

As far as engineering methods for on-line handwriting recognition are concerned, a great number of them has been proposed in the literature [27]. However, it was found very difficult to perform a quantitative comparative analysis. This problem was caused by important differences in the experimental conditions, such as number of writers, type and size of learning and test sets, data format, etc., that reflected a proprietary status of data used in experiments. This deficiency that restricted the advance of this research and development field was tackled by the on-going world-wide UNIPEN project [14], that has already collected over 5 million characters provided by 40 organizations in a common human-readable format. A long time devoted to data debugging and documentation has been based on the distribution of successive learning sets to the project participants, aiming to perform a common benchmark in the near future. The use of UNIPEN data by the research community will permit an analysis of the relative advantages of the proposed methods. Then, reliable methods for handwriting recognition can be used by the industry in order to re-launch efficient pen-computers, in the line of Apple Newton and eMate, or other Personal Digital Assistants. This type of mobile, small (even hand-held) computers, equipped with infrared or other wireless communication interfaces can provide a very interesting solution for the paradigm of ubiquitous, nomadic computing [18].

Important applications of pen-computers include access to Web-based information servers or integration in a flexible, virtual classroom [29].

In this paper, we propose and experimentally study an on-line handwriting recognition system using UNIPEN data. In section 2, we pay attention to the preprocessing stage and especially to character segmentation. This step is necessary because our system considers on-line handwritten specimen as a signal with dynamic information, that can be split into components (between two successive pen-lifts), which are in turn divided to elemental movements, called strokes. Following models of handwriting generation that have been reported in the literature [22], we propose a biology-inspired segmentation method and compare it with another method developed by us, that is based on geometric features.

In the following section, we study the process of finding a feature set, that would be sufficiently discriminant for the strokes, as well as for the components and characters formed by them. Our analysis is based on Shannon entropy and on clustering maps, together with systematic methods that provide an ordered list of candidate features, such as RC [26] or genetic algorithms.

The core of our classification, i.e. the system based on the FasArt neuro-fuzzy architecture, is described briefly in section 4. Such an architecture, motivated by studies on human cognition that resulted in the original ART architectures [9][7], is consistent with our general approach based on human modeling. Furthermore, its compliance to the plasticity-stability dilemma [13], its formulation as a fuzzy logic system, and our previous experience on similar architectures [11] contributed to the adoption of this specific classification scheme.

The experimental results on the UNIPEN second version of the first training release, distributed officially as *train_r01_v02*, are presented in section 5. Based on this experimental study, we can draw conclusions on the relative merits of the proposed segmentation techniques, codification schemes, network complexity, real-time processing characteristics, etc. Similarly, we can take a look on its main error sources and compare its performance with human subjects that were exposed to the same learning and test sets. Finally, we present some steps towards the construction of an allograph lexicon [19], starting from to the fuzzy rules produced by our neuro-fuzzy classifier.

2. CHARACTER SEGMENTATION

The objective of segmentation is to divide complex handwriting pieces into simpler ones, in order to reduce input pattern variability and thus simplify the classifier structure. In this work initial data are isolated characters composed of one or more components, which are to be segmented into strokes. Segmentation methods can be derived from handwriting generation theories as well as heuristic studies of the geometric features in handwritten specimen:

According to Plamondon [21][22], handwriting is generated by a sequence of superimposed simple movements, each with a delta-lognormal velocity profile. Segmentation procedures should then divide the velocity profile of a component into simpler delta-lognormal profiles, each corresponding to one stroke.

It can be observed that the first part (with a form of ‘c’) of the trajectory of several characters is similar, e.g. in ‘a’, ‘c’, ‘d’, ‘g’, ‘o’ and ‘q’. It should therefore be reasonable to expect that any character could be built from a finite base of strokes. Then segmentation should aim to discover such a stroke base, assuming that character allographs correspond to sequences of strokes picked up from the aforementioned base. Allographs are further discussed in section 6.

Besides the extensive use of this generic approach in oriental character recognition, biology-inspired segmentation methods based on minima in velocity profiles (see figure 1f) have been proposed in the literature [24], [17].These algorithms require preprocessing of real data, due to signal quantization, inaccurate sampling rates or human trembling that may introduce irregularities.

In this paper, we propose and compare two methods based either on geometric features or on biological models of handwriting generation.

In the **first method** (**M1**), it is necessary to preserve geometric features of handwriting, and therefore all repeated points are deleted, instead of performing a low-pass filtering. This produces a signal without zero values but with minima in the same locations as in the original velocity signal. These minima are considered as candidates for segmentation points, and later iteratively evaluated in order to delete bad segmentation points. A candidate point C_i is deleted if:

- the angle between $\overline{C_{i-i}C_i}$ and $\overline{C_iC_{i+1}}$ does not correspond to a significant change in trajectory direction, or

- the length of the segment $\overline{C_{i-i}C_i}$ or $\overline{C_iC_{i+1}}$ is smaller than a threshold.

This two-phase segmentation (selection of candidate segmentation points and iterative selection of the geometrically significant ones) is based on [2]. It eliminates velocity minima produced by spatial quantization (discrete sampling of space on the electronic pad surface), selects the best segmentation point in a region with successive changes in trajectory direction, and finally discards spurious minima and too short artifacts in the extremes of a component. Additional enhancements implemented in our system include adaptivity for the angle and length thresholds, as well as mechanisms to avoid proliferation of strokes in curved parts of a component [12].

The **second method** (**M2**), proposed in this paper, is inspired on handwriting generation theories [21][22]. It tries to eliminate perturbations in the velocity signal with the minimum modifications of the character geometry. For this purpose, a FIR low-pass filter was applied to the velocity signal, with a cut-off frequency of 10 Hz [28] and a window length of $f/10$, where f is the sampling frequency of the digitizer pad.

After this preprocessing stage, angle $\theta(t) \approx \arctan(\Delta y(t)/\Delta y(t))$, angular velocity $\theta'(t) \approx \Delta\theta(t)$ and linear velocity $v(t)$ signals are obtained. This information is jointly exploited in order to obtain the segmentation points:

- Initial candidates are determined according to singularities of the angle signal, which are crosses with 0, 90^o and –90^o. Crosses with 0 will determine peaks (due to sudden changes in orientation), loops (in which there is a transition to a different angular region), and crosses with horizontal base lines. Also, crosses with 90^o or –90^o can be used to detect crosses with vertical base lines. These candidates can be seen in figures 1a and 1b for a sample of letter 'a'.
- In this step, we look for extrema of the angular velocity signal in the vicinity of the candidate points chosen in the previous step. This second criterion is not used independently, because of its extreme sensibility to possible perturbations that might have passed through the low-pass filter stage. The output of this stage is shown in figures 1c and 1d for the same as above character.
- Finally, minima in linear velocity next to remaining candidates determine the final set of segmentation points. The use of this information, which is theoretically dual to extrema in angular velocity, permits us to reduce the number of local singularities that were previously accepted as segmentation points. In figures 1e and 1f the final segmentation points are shown for our example.
- Besides this basic segmentation procedure, we can use any available *pen-up* information, in order to delete small strokes that are due to a too early contact of the pen with the pad or a too late withdrawal, in the beginning or in the end of the component respectively. Information of the linear velocity profile can also be used to delete small pulses (ornamental strokes), as shown in figures 1g and 1h.

Several criteria can be used in order to evaluate the relative merits of the proposed segmentation techniques: **Simplicity**, in order not to pose obstacles to a real time processing of the global system; **Consistency**, i.e. similar characters should be segmented to similar strokes, independently of small variations; **Limited number of fundamental strokes**, i.e. the obtained strokes should be able to be grouped in a small number of clusters, thus reducing the size of the stroke base; **Small number of strokes per character**, in order to reduce the length and consequently the variability of stroke sequences that represent the characters; and finally, **Independence with respect to experimental conditions**, as an indicator of the method robustness.

Our experimental results on the UNIPEN data, permitted us to make the following observations:

- Both methods are suitable for real-time processing, since they consume a time in the range of 6 msec/character, measured with Linux *time* utility in a Pentium 120 MHz-based PC. The second method (**M2**) is slightly slower.
- The second method (**M2**) is more consistent, although both of them provide satisfactory results. The main reason is the combination of several complementary criteria in method **M2**, while the method **M1** is extremely sensitive for shapes of intermediate curvature. This fact is specially present in upper case letter segmentation, for which method **M2** is clearly better, as shown in section 5.
- The size of the stroke base is lower for method **M2** (further study is made in the following sections, that include clustering and classification stages).
- The number of strokes per character was found to be satisfactory, since it was over 6, only in less than of 1% of the characters, with a mean close to 3 strokes/character. Moreover, method **M2**

typically provides a number of strokes m_c±1, where m_c is the average number of strokes for the samples of character *c*.

- The geometric method (**M1**) is more robust with respect to variations of the handwriting speed. Method **M2** depends on the way the character is drawn, and therefore has problems with calligraphic or extremely fast handwriting.

Globally, both methods were found to have a satisfactory performance, which will be further evaluated in the following sections.

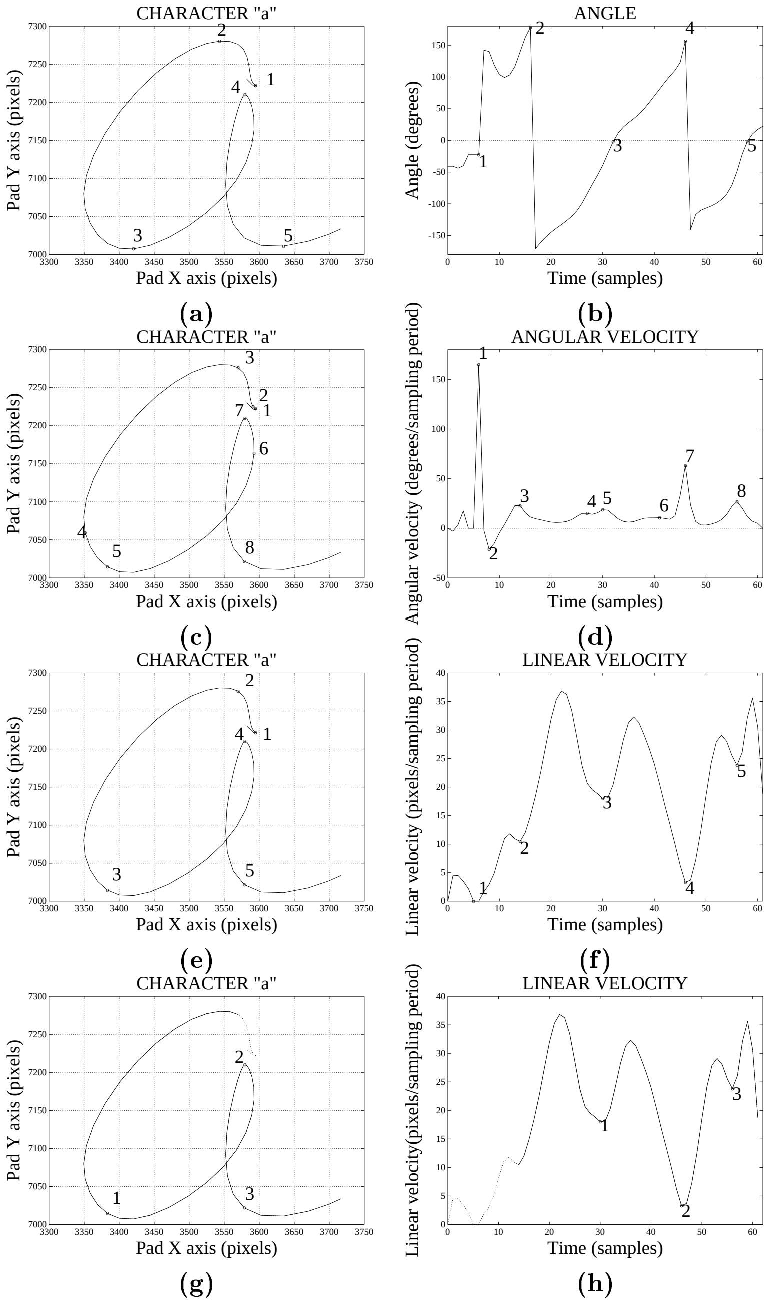

Figure 1. Successive steps of the proposed segmentation method based on biological models (M2). Character 'a' with the corresponding segmentation points and the associated signal information is shown. (a,b) Singularities in angle signal are selected as segmentation candidate points. (c,d) Extrema in angular velocity are found in the vicinity of candidates of previous step. (e,f) Minima in linear

velocity next to candidates of previous steps are selected as final segmentation points. (g,h) *Pen-up* information is used to delete some strokes (dotted lines)

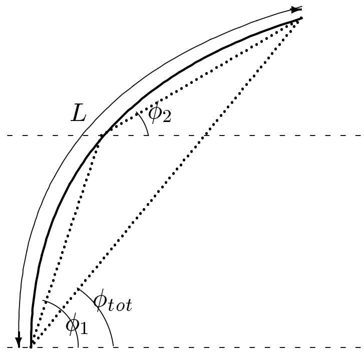

Figure 2. Geometric features used in the selected codification. ϕ_1 and ϕ_2 are the phase points of the stroke, ϕ_{tot} the total phase, and *L* its length.

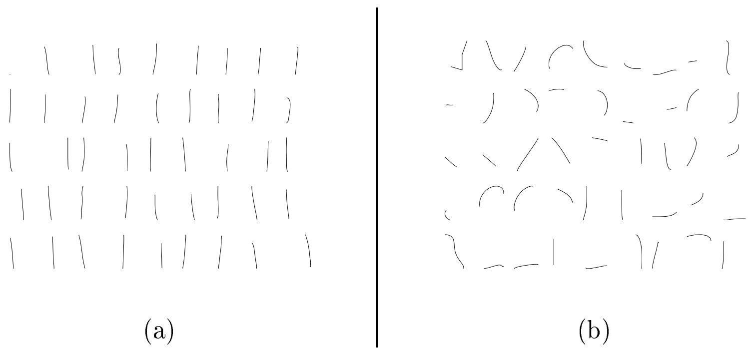

Figure 3. Clustering maps for the same data set, with different features used for codification. (a) Two phases, total phase, sequential feature and last *y* coordinate. (b) Main direction of the stroke using Karhunen-Loéve axis, and mean *x* and *y*. Clustering was made with the unsupervised module of FasArt (see section 4), for the upper case letters after segmentation based on biological models (method M2).

3. FEATURE EXTRACTION

Feature extraction or codification is necessary in order to cluster strokes within our effort to build a base of character primitives and classify stroke sequences that correspond to complete characters. Then features should be discriminant, geometrically significant and consistent for each character class. In this work we built a small set of features extracted from the literature [17] and from our knowledge about the problem. Several feature sets were evaluated, that come from our initial feature pool. The well-known task of feature extraction was handled through the use of two techniques, i.e. Shannon entropy [25] and clustering maps.

Shannon entropy [25] of a continuous variable x can be defined by

$$H = -\sum_{i=1}^{N} P_i \log_2 P_i \tag{1}$$

where P_i is the probability of the i-th of the N equal partitions made in the interval of variation of variable x.

Shannon entropy can be understood as the quantity of information provided by one variable. In the case of feature extraction for pattern classification, we are interested in having high values, i.e. variables with high entropy are more discriminant. However, high Shannon entropy does not guarantee that features contribute to a geometrically significant feature set.

Table I shows Shannon entropy values for the selected features. All features are geometric (see figure 2), except that mentioned as "Main orientation of a stroke using K-L transform", which corresponds to the K-L transform of the direction of the stroke. Using the unsupervised module of the neural network proposed in section 4, clustering maps of the strokes were obtained, for two feature sets that contain the most discriminant features according to Shannon entropy.

In figure 3 we can clearly observe that only one feature set produced geometrically significant clustering maps, that are more appropriate for our final classification goal.

For the final feature selection a systematic method, called regularity criterion (RC) [26], was employed. This method seeks the minimum error index, while progressively including features in the final set. Although a voting strategy was additionally used, there was no warranty that we could avoid local minima. In order to reduce such a risk, a genetic algorithm was alternatively used, seeking for a minimum in an objective function (the same classification error index used in RC method) by pseudo-randomly varying (*mutating*) the feature set. However this method produced similar results to the use of RC. Finally, RC method was run on several data subsets and a voting strategy applied to determine the optimum feature set. Such a process permitted us to conclude, e.g. that a sequential feature, indicating whether a stroke is in the beginning or in the end of a component, clearly contributes to a better classification. On the other hand, it was found that some features sets are more appropriate for digits and upper case letters, where variation is much less than that found in lower case letters. However, we decided to use the same feature set for all types of characters, that consisted in length L, two phase points (ϕ_1 and ϕ_2) and total phase ϕ_{tot} of the stroke, the end y coordinate (normalized) and a discrete sequential feature that informed if the stroke started and/or ended component (see figure 2).

Table I
Normalized Shannon entropy for some codifications

Features	Entropy	Features	Entropy
Length	0.7623	Final y coordinate	0.9446
Two phase points	0.9297 0.9393	Minimum y coordinate	0.7310
Total phase	0.9238	Main orientation of stroke using K-L transform	0.9588 0.85780
Area	0.5988	x and y mean	0.9803 0.9702

4. FASART: FUZZY ADAPTIVE SYSTEM ART-BASED

Fuzzy sets theory based on pioneer work by Zadeh [30], introduces an alternative to cope with vague information through a formulation close to that used in natural language. It has been used in different areas including pattern recognition and classification [3].

In the past few years two criteria have been followed for the definition of fuzzy logic systems: on one hand heuristic methods usually based on ideas coming from statistical classification [1]; on the other, embedding of fuzzy logic into neural networks in order to have fuzzy systems with learning capabilities [15]. Within the architectures of Adaptive Resonance Theory, the Fuzzy ART model [9] introduces fuzzy operators. Thus, a classifying module was achieved, that pretends to provide a self-organizing classification, based on categories of fuzzy nature. Similarly, a supervised architecture, Fuzzy ARTMAP [7], is built using Fuzzy ART modules within the original ARTMAP architecture [8]. In ARTMAP-based architectures, supervised models are defined on the base of the incremental learning of the relations that appear between the outputs of two self-organizing modules. The introduction of fuzzy logic should permit the definition of categories as fuzzy sets, with their associated membership functions. However, Fuzzy ARTMAP does not include a formal definition of the categories as fuzzy sets, and therefore cannot be properly defined as fuzzy logic systems [5]. For a review of the main principles of ART architectures and a detailed discussion of their treatment of Fuzzy Logic, see [23], and [20] for a detailed study of the STORE memory model.

A new neuro-fuzzy model, called FasArt [5], formally relates neural activation with fuzzy set membership functions, permitting its dual interpretation as fuzzy logic system and as neural network. For this purpose, the activation function of Fuzzy ART [9] is redefined by an expression which can also be interpreted as the triangular fuzzy membership function described by the weights of the associated category. For each input dimension i=1,...M the activation η_{R_k} of unit k is determined as in figure 4, where w_{ki} and w_{ki}^c are weights present in Fuzzy ARTMAP architecture, and c_{ki} are newly introduced weights representing the center of the fuzzy membership function. A new design parameter γ permits us to control the degree of fuzziness of the fuzzy sets (higher γ would lead to less fuzzy sets). The total activation is given by:

$$\eta_{R_k} = \prod_{i=1}^{M} \eta_{ki}(I_i) \tag{2}$$

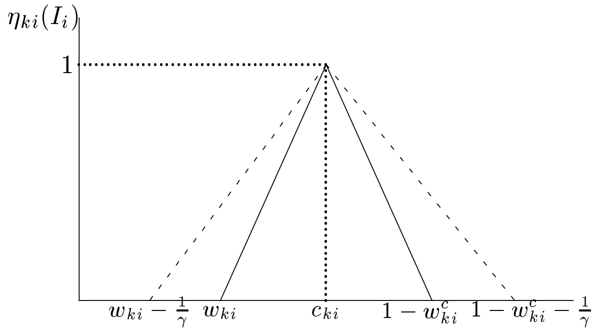

Figure 4. Neural activation/fuzzy membership function in FasArt, where the fuzzy degree of the membership function can be varied with parameter γ.

With these changes, FasArt shares with Fuzzy ARTMAP the ARTMAP-based architecture, and therefore the main advantages of such models are preserved, i.e. fast learning, compliance to the stability-plasticity dilemma, and capability for incremental learning in test phase. Moreover, the introduction of a duality between activation function and fuzzy membership function permits the interpretation of an ARTMAP-based architecture as a fuzzy logic system, allowing us to take advantage of general theorems, that guarantee FasArt capability to approximate arbitrary functions [5].

Being FasArt a fuzzy logic system with self-organizing features, i.e. capabilities to learn from examples, fuzzy rules can be extracted from FasArt weights. Actual output is obtained after a defuzzification process, in the general case of identifying a continuous function. However, in character recognition problems output is simply a label and defuzzification is not necessary. The process of classifying a character can be summarized as follows:

1. Present the features vector corresponding to one stroke.
2. Cluster this vector with ART_a, and storing the resulting cluster in the STORE.
3. Repeat this presentation, clustering and storing for the rest of the strokes.
4. Find through the inter-ART map the category in ART_b most related to the sequence of strokes in the STORE.
5. Obtain the label corresponding to this category. Labels of other activated categories can be obtained with smaller confidence degree.

Then, an ordered list is obtained reflecting the confidence degree, that a certain input is related to the examples that generated the categories (fuzzy sets). This feature can be used in a post-processing module, e.g. using a syllabic dictionary [16] in order to correctly predict complete words.

For the on-line handwritten character recognition studied in this paper, the original FasArt structure has been modified, on the basis that we adopt a stroke-based model for classification; after unsupervised module a sequence of clusters must be associated with the character label. For this purpose we use the STORE model [4], based on studies of human memory [13]. The introduction of this module makes necessary to modify inter ART reset scheme, in order not to reclassify all clusters in the sequence but only the less confident.

Besides, a statistical module using edit distance between input sequence and prototypes has been used to validate FasArt prediction. This element makes use of the sequence of clusters produced by FasArt unsupervised module, a database of the sequences that appeared during training, and a distance module. Edit distance is calculated between the input sequence of clusters and the stored sequences, each with an associated label, as a weighted average of substitution distance (if both sequences have the same length), insertion and deletion distance (if one sequence is one item longer). Here substitution distance was considered more important.

The label that minimizes such distance would be the correct prediction. However, when this edit distance was used, we found experimentally that often several labels provide the minimum distance. Therefore, this statistical module is not suitable for building a classifier, but is useful to validate FasArt prediction, i.e. if FasArt prediction is among those which provide minimum distance, prediction is accepted, and otherwise an inter-ART reset is forced.

5. EXPERIMENTAL RESULTS FOR CHARACTER RECOGNITION

As we have already mentioned in the preceding sections, all experimental results were obtained using UNIPEN second version of the first training release, distributed as *train_r01_v02*. From this set we excluded those files with the sampling rate of the pad not documented, since this element is necessary for the proposed segmentation method based on biological models (method **M2**). A total of 3838 digits, 12198 lower case letters and 4248 upper case letters were divided randomly into training and test sets, assuming that all the writer provide samples to both sets.

Recognition experiments were carried out with both segmentation methods, using a feature set of two phases, total phase, length, sequential feature and last y coordinate (totaling 6 features), and the neuro-fuzzy classifier based on FasArt. For training ρ=0.7, β=β_c=0.8 in ART_a, while during the test stage ρ was relaxed to 0.7. These parameters were tuned heuristically, although it was experimentally found that ρ, β and β_c values could be safely varied in [0.5,1] without significant decrease of recognition rate, as long as ρ value in the training stage was greater to that used in the test stage. Classification results after four training cycles are shown in table II.

In section 2, it was argued that segmentation based on biological models (method **M2**) is more consistent than geometric (method **M1**), specially for open curves, like those mostly found in upper case letters. In this sense it is expected that with segmentation method **M2** the number of clusters created after training should be smaller and therefore higher classification rates should be achieved. This is experimentally confirmed as shown in table II. The average number of clusters for all types of characters was reduced in 12%, being the most significant reduction (over 15%) in the case of upper case letters.

It is important to assess error causes in order to estimate data inaccuracies and to eventually improve some of the modules. The main detected error causes were the following, as illustrated in figure 5:

- *Erroneously labeled data* (as judged by human recognition), since not all characters have been checked by human recognizers in this training release. This is the case of figure 5a, where correct label should have been 'C', as predicted by the proposed system, instead of 'I'.
- *Ambiguous data* as the characters shown in figure 5b, for which even a human does not clearly recognize a 'B'.
- *Segmentation errors*, that may vary, ignore or enhance geometric variations of the character. Figure 5c is an example of such an error.
- *Insufficient feature set*, that may not include information enough to classify the patterns, as in figure 5d, where two phases are not enough to distinguish a straight line from a very open curve.

A reasonable performance limit to the proposed system can be obtained when compared to human recognition, since this can let us estimate how many bad labeled, ambiguous or unrecognizable data are in the test sets. For human recognition three volunteers not related to this research were employed. Initially the training patterns were shown in a console, six patterns at a time, with the corresponding labels. They were given free time to study each screen. Afterwards, they were shown one test pattern per screen and required to label it or simply give an '*I do not know*' answer. The human average classification rate is shown in table II. Compared to our previous experiments [18] on digit recognition by humans without training, a 6% improvement can be found due to training on characters from the same writers found in the test set, since there exists a great variability of writers and cultures found in UNIPEN data sets. Two of the humans involved in both experiments explicitly confirmed this issue. Additionally, it can be seen that recognition of lower case letters was a much more difficult task for both humans and the proposed method.

Table II
Recognition rates for the different test sets after four training cycles, using the geometric segmentation (M1), the method based on biological models (M2), and those achieved by human testers.

	Set	Correct	Wrong	Unclassified
Geometric segmentation (M1)	Digits	85.39	7.64	6.96
	Upper case letters	66.67	14.74	18.60
	Lower case letters	59.57	29.36	11.07
Biological models segmentation (M2)	Digits	82.52	9.23	8.24
	Upper case letters	76.39	12.37	11.24
	Lower case letters	58.92	32.92	7.14
Human recognition	Digits	96.17	2.02	1.81
	Upper case letters	94.35	4.71	0.94
	Lower case letters	78.79	15.15	6.06

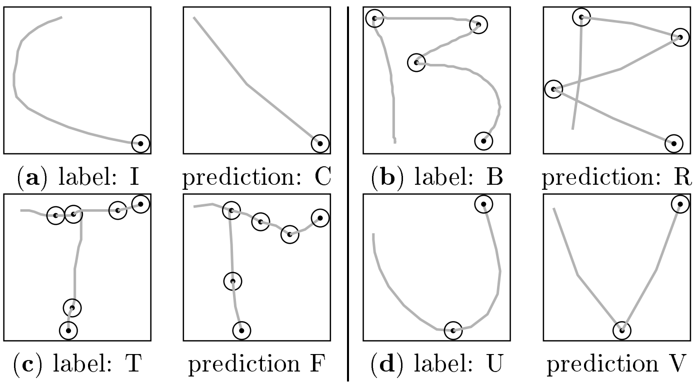

Figure 5. Typical classification errors. The original character and FasArt internal representation are shown with their labels. (a) Badly labeled data. (b) Ambiguous data. (c) Badly segmented data. (d) Character features not retained by codification.

One important issue for developing applications based on on-line handwriting recognition in a pen-computer is real time performance. In this sense, processing time required by the proposed system is low enough (on a Pentium 120MHz PC, measured with Linux *time*, 276 ms/char for the whole test process, including 7 ms/char for segmentation, 5ms/char for feature extraction, and 265 ms/char for the actual test). We should remind that handwriting is a low velocity process, especially as compared to the speed of 0.28 sec/char achieved by the proposed system. Here we can emphasize the low training time, due to the fact that learning is guided by pattern matching instead of error minimization, as opposed to backpropagation-based solutions for which a much higher training time is expected [6].

5. PRE-ALLOGRAPHS: TOWARDS AN ALLOGRAPH LEXICON

The building of an allograph lexicon has been studied [19] as an instrument for syntactic recognition and script description. If a stroke-based model is used, a character x composed of n strokes s_i has one or several allographs, i.e. different forms of being written:

$$\text{IF } s_1 \text{ IS } c_1 \text{ AND ... AND } s_n \text{ IS } c_n \text{ THEN } x \text{ IS } l$$

where stroke s_i belongs to cluster c_i and character x has a label l.

Such rules can be obtained by our neuro-fuzzy system FasArt, since each stroke is classified in a cluster by ART_a and the whole character belongs to a fuzzy set with label l, with a certain membership function value. However, extraction of a reduced set of rules is difficult when the number of clusters becomes high, because of the supervision method employed in FasArt or Fuzzy ARTMAP [10]. In order to reduce network complexity, exploit the classifier generalization capabilities and simplify the process of rule fusion [10] to create allographs, reconstruction of the prototype from FasArt c_i weights is made, and unsupervised classification of these prototypes is carried out with an unsupervised module of FasArt, leading to a great reduction in the number of clusters. A typical example of this methodology is the reduction of 567 clusters with supervised training on upper case letters to 166 unsupervised clusters.

Therefore, we use an extended definition of allograph (called pre-allograph) in the following way:

$$\text{IF}\begin{Bmatrix} s_1 & \text{IS} & c_{1_1} \\ s_1 & \text{IS} & \vdots \\ s_1 & \text{IS} & c_{1_{p1}} \end{Bmatrix} \text{AND} \ldots \text{AND} \begin{Bmatrix} s_n & \text{IS} & c_{n_1} \\ s_n & \text{IS} & \vdots \\ s_n & \text{IS} & c_{n_{p_n}} \end{Bmatrix} \text{THEN } x \text{ IS } l$$

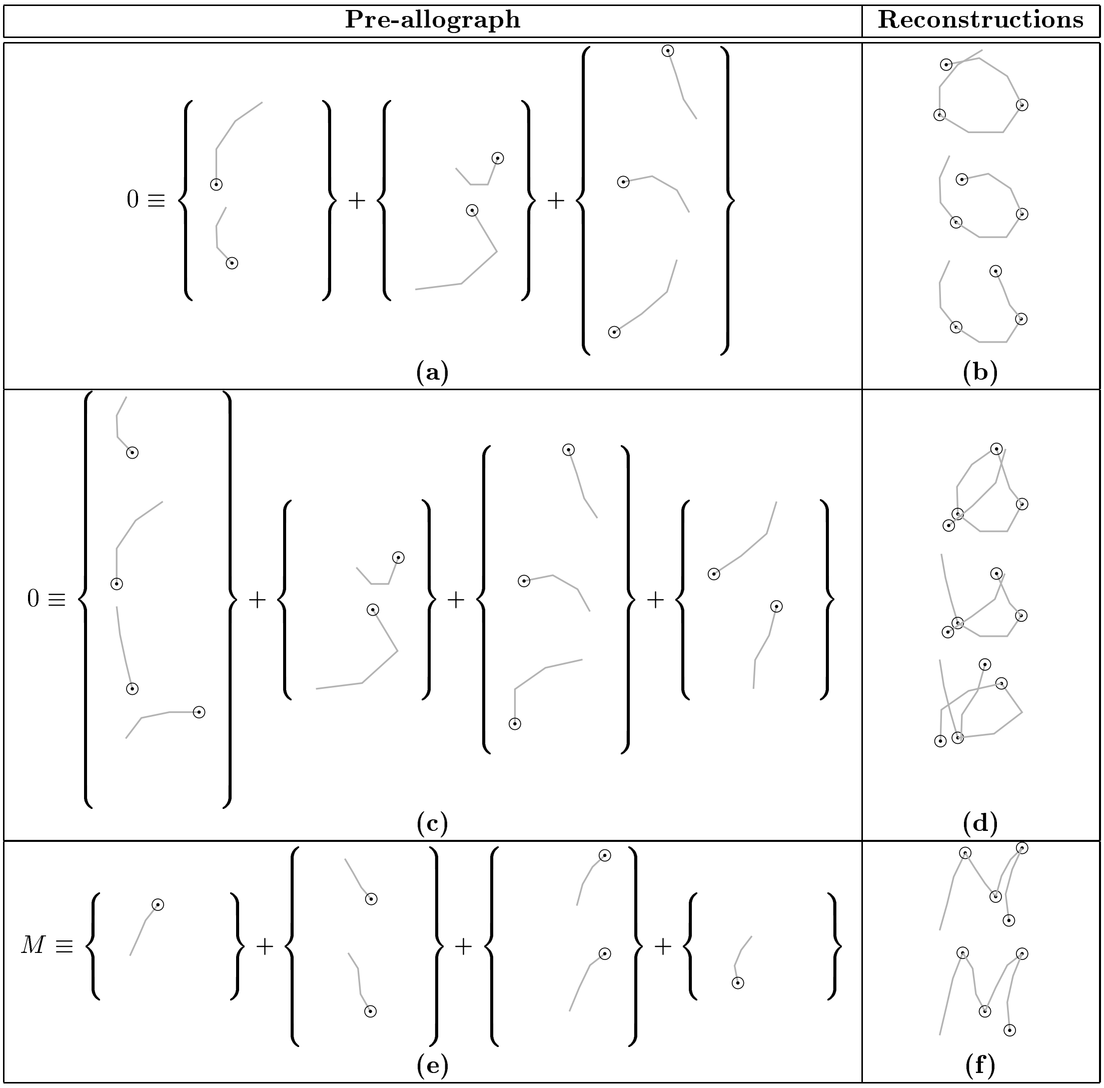

Figure 6. Typical pre-allographs with strokes reconstructed from FasArt c_i weights. This reconstruction is based on the weights c_{ki}, and thus simplifies the fact that clusters are fuzzy sets. (a,b) Pre-allograph for three-stroke '0' and typical reconstructions from this pre-allograph. (c,d) Pre-allograph and typical reconstructions for a four-stroke '0'. (e,f) Pre-allographs and typical reconstructions for a four-stroke 'M'. Pre-allograph for a three-stroke 'N' is formed by the first three stokes of 'M' pre-allograph.

Using this definition, all characters from our UNIPEN set could be described by a small set of *pre-allographs*. For example, in figure 6a we can see that a three-stroke digit '0' could be represented by three conditions joined by **AND**, where 2, 2, and 3 alternatives were necessary for each condition (stroke).

A detailed analysis of the pre-allographs permitted us to identify some interesting issues:

- Segmentation errors may produce different pre-allographs. Thus a character class may have pre-allographs of the type: (c_1,c_2,c_3) and (c_4,c_3), i.e. although they have a common stroke that belongs to the same cluster, their representation is different because of different segmentation.
- Different allographs have a common part, such as (c_1,c_2,c_3) for normal '0' and (c_1,c_2,c_3,c_4) for a computer '∅', where clusters (c_1,c_2,c_3) form a circumference and c_4 a slash. This is illustrated in figure 6a and 6c.
- Different characters with a part geometrically similar (e.g. 'M' and 'N'), will be represented by *pre-allographs* with a part in common. This is illustrated in figure 6e, where first three strokes of a 'M' pre-allograph form a 'N' pre-allograph.
- A great number of exemplars in the actual data can be described with these pre-allographs. With pre-allograph in figure 6a, 54% of the three-stroke '0' samples are described, and with figure 6c, 51% of the four-stroke '0' samples. Since 66% of '0' exemplars are segmented in three strokes and 15% in four strokes, with these two pre-allographs 44% of the '0' samples are described. The pre-allograph in figure 6e describes 69% of the four-stroke 'M' samples, which constitute 51% of the total.

6. CONCLUSIONS

In this paper a complete scheme for handwriting recognition is proposed. A special merit of this work is the fact that the data set came from the UNIPEN project. In it, various writers with different cultures and habits used different tablets in order to produce a common data bank for future benchmarks. The difficulty of character recognition for this set was confirmed by a test with tree human subjects, who reached a recognition rate of 94-96% for digits and upper case letters, while a rate of only 79% was achieved for the most difficult set of lower case letters. Our system achieved a rate of about 86% for digits, while for the highly variable lower case letters a percentage of around 59% was reached.

The general approach was inspired on theories of handwriting generation, as well as on neuro-fuzzy models that approach the processes of human perception and cognition. On one hand, we tried to exploit the decomposition of handwritten specimen into elemental strokes. On the other hand, recognition was based on FasArt architecture that was shown to represent a fuzzy logic system with capabilities of universal function approximation, while preserving all important advantages of Adaptive Resonance Theory models.

Two segmentation methods were proposed and experimentally studied, showing that the one inspired in biological models was more consistent and contributed in lower network complexity. Also, a systematic search was performed in order to find the best feature set, based on Shannon entropy and clustering maps. Experimental results confirmed a very good performance in terms of learning time. This can be expected from the characteristics of learning by matching of ART networks, and also from experimental results from the literature which report improved performance of ART networks in terms of training time when compared to other main-stream backpropagation neural networks. Furthermore, real-time characteristics for the test phase have been shown in our experiments.

Finally, we showed how the generation of fuzzy rules by FasArt permitted the construction of pre-allographs. On-going study with respect to rule fusion and reduction of network complexity, may conclude with the building of an allograph lexicon. Such a lexicon may contribute to the further study of the way characters are generated and recognized, as well as to their use in a recognizer of fuzzy-syntactic nature.

Acknowledgements

The authors would like to thank J.A. Gago, M. Araúzo, A. Grande and G.I. Sainz, as well as the other members of our research group, for their contributions to this paper, as well the volunteers for human recognition experiments for their kind collaboration. Partial funding of the research work on fuzzy-neural networks is due to the European projects BRITE Project BRE2-CT94-0976 "PSYCHO" and ESPRIT Project 22416 "MONNET". Special thanks for the financial support of the CICYT project TIC99-0446-C02-04.

REFERENCES

1. S. Abe and M. Lan, "A method for fuzzy rules extraction directly from numerical data and its application to pattern classification", *IEEE Transactions on Fuzzy Systems*, 3(1):18-28, February 1995.
2. A. Al-Emami and M. Usher, "On-line recognition of handwritten arabic characters", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):704-710, July 1990.
3. J. Bezdek and S. Pal, *Fuzzy Models for Pattern Recognition*, IEEE Press, Los Alamitos, CA, USA, 1992.
4. G. Bradski, G.A. Carpenter, S. Grossberg, and D.B. Rosen, "Working memory networks for learning temporal order, with applications to three-dimensional visual object recognition", *Neural Computation*, 4(2):270-286, March 1992.
5. J.M. Cano Izquierdo, Y.A. Dimitriadis, M.J. Araúzo Bravo, and J. López Coronado, "FasArt: A new neuro-fuzzy architecture for incremental learning in system identification", *Proc. of the IFAC World Congress, IFAC96*, San Francisco, CA, USA, volume F, pages 133-138, June 1996.
6. G.A. Carpenter, S. Grossberg, and K. Izuka, "Comparative performance measures of fuzzy ARTMAP learned vector quantization and back-propagation for handwritten character recognition", *Proc. of the International Joint Conference on Neural Networks, IJCNN'92*, volume 1, pages 794-799, 1992.
7. G.A. Carpenter, S. Grossberg, N. Markuzon, J. Reynolds, and D. Rosen, "Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps", *IEEE Transactions on Neural Networks*, 3(4):698-713, September 1992.
8. G.A. Carpenter, S. Grossberg, and J. Reynolds, "ARTMAP: Supervised real-learning and classification of nonstationary data by self-organizing neural network", *Neural Networks*, 4:565-588, 1991.
9. G.A. Carpenter, S.Grossberg, and D.B. Rosen, "Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance systems", *Neural Networks*, 4(1):759-771, 1991.
10. G.A. Carpenter and H.A. Tan, "Rule extraction: From neural architecture to symbolic representation", *Connection Science*, 7(1):3-27, 1995.
11. Y.A. Dimitriadis and J. López Coronado, "Towards an ART-based mathematical editor, that uses on-line handwritten symbol recognition", *Pattern Recognition*, 28(6):807-822, June 1995.
12. E. Gómez Sánchez, J.A. Gago González, Y.A. Dimitriadis, J.M. Cano Izquierdo, and J. López Coronado, "Experimental study of a novel neuro-fuzzy system for on-line handwritten UNIPEN digit recognition", *Pattern Recognition Letters*, 19:357-364, 1998.
13. S. Grossberg, *Studies of mind and brain: Neural principles of learning, perception, development cognition, and motor control*, Reidel Press, Boston, MA, USA, 1982.
14. I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, and S. Janet, "UNIPEN: Project of on-line data exchange and recognizer benchmarks", *Proceedings of the 12th International Conference on Pattern Recognition*, Jerusalem, Israel, pages 9-13, October 1994.
15. C.T. Lin and C.S.G. Lee, *Neural Fuzzy Systems*, Prentice Hall, 1996.
16. F. Merino Pastor, Y.A. Dimitriadis, R. García García, and J. López Coronado, "A dictionary-based neural network for on-line handwriting recognition", In M.L. Simner, C.G. Leedham, and A.J.W.M. Thomassen, editors, *Handwriting and drawing research: Basic and applied issues*, pages 343-357, IOS Press, Amsterdam, Netherlands, 1996.
17. P. Morasso, L. Barberi, S. Pagliano, and D. Vergano, "Recognition experiments of cursive dynamic handwriting with self-organizing networks", *Pattern Recognition*, 26(3):451-460, 1993.
18. S. Narayanaswamy et al, "Application and network support for Infopad", *IEEE Personal Communications*, 3(2):4-17, March 1996.
19. M. Parizeau and R. Plamondon, "A fuzzy-syntactic aproach to an allograph modeling for cursive script recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(7):702--712, July 1995.
20. J. Pérez Maroto, Y.A. Dimitriadis, J.M. Cano Izquierdo, and J. LópezCoronado, "Analysis and applications of the STORE neural model in recognizing handwritten symbols", *Proc. of the 3rd International Workshop on Artificial Neural Networks*, volume 943, Málaga, Spain, June 1995.

21. R. Plamondon, "A kinematic theory of rapid human movements. Part 1: Movement representation and generation", *Biological Cybernetics*, 72:295-307, 1995.
22. R.Plamondon, "A kinematic theory of rapid human movements. Part 2: Movement time and control", *Biological Cybernetics*, 72:309-320, 1995.
23. G.I. Sainz Palmero, Y.A. Dimitriadis, J.M. Cano Izquierdo, E. Gómez Sánchez and E. Parrado Hernández, "ART based model set for pattern recognition: FasArt family", in H. Bunke, editor *Neuro-fuzzy pattern recognition*, to appear, World Scientific
24. L.R.B. Schomaker, "Using stroke -or character- based self-organizing maps in the recognition of on-line connected cursive script", *Pattern Recognition*, 26(3):443-450, 1993.
25. C. Shannon, "A mathematical theory of communication", *Bell System Journal*, 27:379-423, July 1948.
26. M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling", *IEEE Transactions on Fuzzy Systems*, 1(1):7-31, February 1983.
27. C.C. Tappert, C.Y. Suen, and T. Wakahara, "The state of the art in on-line handwriting recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(8):787-808, August 1990.
28. H.L. Teulings and F.L. Maarse, "Digital recording and processing on handwritting movements", *Human Movement Science*, 3:193-217, 1984.
29. M.J. Verdú, O.M. González, et al. "Pen-computing and Groupware: A chance to improve collaborative writting", *Proc. of the 7th International Conference on Human-Computer Interaction*, volume 2, pages 463-466, San Francisco, CA, USA, August, 1997.
30. L.A. Zadeh, "Fuzzy sets", *Information and Control*, 8:338-353, June 1965.