A Saliency-Based Multiscale Method for On-Line Cursive Handwriting Shape Description

Claudio DE STEFANO°, Marco GARRUTO* and Angelo MARCELLI*
° DAEIIMI - Università di Cassino (FR), ITALY
** DIIIE - Università di Salerno (SA), ITALY*
destefano@unicas.it, marcounisa@libero.it, amarcelli@unisa.it

Abstract

We propose a method derived from an analogy with the primate visual system for selecting the best scale at which the electronic ink of the handwriting should be described. According to this analogy, the method computes a multiscale features maps by evaluating the curvature along the ink at different levels of resolution and arranges them into a pyramidal structure. Then, feature values extracted at different scales are combined in such a way that values that locally stand out from their surrounds are enhanced, while those comparable with their neighbours are suppressed. A saliency map is eventually obtained by combining those features value across all possible scales. Such a map is then used to select a representation that is largely invariant with respect to the shape variations encountered in handwriting. Experiments on two data sets have shown that simple algorithms adopting the proposed representation lead to very stable stroke segmentation and feature matching.

1. Introduction

Many studies on handwriting generation have shown that complex movements like handwriting can be seen as a composition of elementary movements, or strokes, each corresponding to an elementary shape [1-3]. According to this approach, handwriting generation can be seen as the result of a complex motor program that generates the appropriate sequence of strokes needed to draw the sequence of elementary shapes forming the handwriting. Thus, handwriting recognition may be seen as a bottom-up process that extracts the strokes from the ink, encodes their features and eventually performs the classification by comparing the description of the specimen with those of a set of allographs.

Studies on visual perception have shown that curvature plays a key role in our perception of shape and its organization into parts [4, 5]. Therefore, since the sixties many efforts have been made to develop algorithms for computing the curvature along a line and then use this information for both locating curvature maxima, in order to extract the elementary parts forming the shape, and describing the shape of each part by some encoding of its curvature [6].

The main problem while pursuing this approach is that of finding an operative definition of curvature able to cope with the large variability exhibited by handwriting. Such variability emerges from four main factors: posture, neuro-biomechanical noise, style and sequencing. Posture refers to changes in size, position, orientation and slant of the handwriting that mainly depends on the postural condition of the writer. Neuro-biomechanical factors greatly affect the quality of handwriting by modifying both the motor control program and the production of individual strokes. As a matter of fact, fluency in handwriting emerges from the time superimposition of strokes due to anticipatory effects, so that the actual trajectory for drawing a stroke, and therefore its shape, depends on both the previous and the successive ones [3]. Style refers to the various models that are associated to a single character by different writers. Finally, sequencing refers to the variation in the order of individual strokes during handwriting. The overall results is that handwriting meant to encode the same word, produced by the same writer at different times or by different writers, may correspond to rather different set of digital lines. As a consequence, applying the mathematical definition of curvature to those lines may result in detecting curvature maxima not corresponding to perceptually relevant points, eventually providing different descriptions for similar shapes. Overall, the classifier has to deal with one more source of variability in addition to those naturally embodied by handwriting.

To solve this problem, the large majority of the algorithms adopt some kind of technique to filter out “non

0-7695-2187-8/04 $20.00 © 2004 **IEEE**

relevant" changes of curvature. This is generally achieved by some thresholding technique, according to which curvature changes whose value is larger then the threshold are retained as perceptually relevant, while the remaining one are discarded. Thus, setting the value for the threshold represents the major challenge for those methods, and seems to be the main reason of the erratic behaviour they sometimes exhibit. To overcome this drawback, it has been suggested that effective shape representations can be achieved by a multiresolution approach, computed along a fine-to-coarse scale. Those parametric representations consider features over a continuum of scales simultaneously rather than at an individual, predefined scale. Independently of the actual method to compute the intermediate representations, scale-space exhibits many interesting properties, mainly hierarchical decomposition in a perceptually satisfying manner, and therefore seems very promising. Nonetheless, even when this approach is adopted, the problem of selecting the appropriate scale, i.e. the setting of the value for the scale parameter, still remain an open issue.

We propose to address the problem of selecting the best scale at which the electronic ink of the handwriting should be described by exploiting the concept of saliency introduced for modelling visual attention shift in primate visual system [7]. According to this model, multiscale features maps are computed at different levels of resolution and arranged into a pyramidal structure. Then, feature values extracted at different scales are combined in such a way that values that locally stand out from their surrounds are enhanced, while those comparable with their neighbours are suppressed. The saliency map is eventually obtained by combining those features value across all possible scales. Such a map enjoys the property of exhibiting higher values in correspondence of region of the scene whose features stands out from their surroundings on a larger number of scales. In other words, it encodes for local conspicuity over the entire visual scene. Following this approach, the problem of selecting the most suitable representation can be reformulated as an early, preattentive scene analysis problem. The scene the system is looking at is the electronic ink, and the features we extract from the ink, of which we want to estimate the saliency at different levels of resolution, is its curvature. The best representation, thus, is that corresponding to the scale at which the observed curvature changes are the closest, according to a given metric, to the saliency map. By reformulating the scale selection problem as a preattentive scene analysis we expect to provide a biologically plausible background to decide when a curvature change is "non relevant" and therefore should be discarded. Moreover, by analogy with the experimental results on preattentive visual tasks, the obtained representation should be much more invariant with respect to locally prominent but globally non-significant changes of curvature.

The remaining of the paper is organized as follows: Section 2 describes the adopted curvature scale-space and the saliency map obtained from it. Section 3 illustrates the selection of the desired representation and the description of the underlying handwriting shape in terms of its curvature. Preliminary experimental results are reported in Section 4 and some concluding remarks are eventually left to Section 5.

2. The curvature scale-space

As mentioned in the introduction, the information we extract from the ink, of which we want to estimate the saliency at different level of resolution, is the curvature. To implement this idea, we need to construct representations of the ink at different levels of resolution, estimate the curvature for each of them, and eventually compute the saliency of the curvature.

In differential calculus, the curvature c at a point p on a continuous plane curve Γ is defined as

$c=lim_{\Delta\lambda->0}\ \Delta\alpha/\Delta\lambda$

where λ is the curvilinear abscissa along Γ and $\Delta\alpha$ is the change in the angles of the tangents to Γ at distance λ and $\lambda+\Delta\lambda$, respectively. When the analytical representation of the curve is not available, the above limit is difficult to calculate [8]. However, by using a small unity interval $\Delta\lambda=1$ along the curve, c can be approximated as $c=\Delta\alpha$. This idea can be implemented by interpolating the available data points in such a way that the distance between successive points is equal to 1.

In our case, the input provided by the tablet are the sequences x(n) and y(n) (n=1..N), representing the coordinates of the points in the (x,y) tablet plane corresponding to the uniform time sampling of the ink produced by the writer. Thus, changes in the writing speed, either due to noise or exhibited in correspondence of the terminal parts of the strokes or where two successive strokes interact, produce changes in the density of the points along the line. For this reason, before applying our method, we need to preprocess the original set of points by adding new points in such a way to obtain a 8-connected line in the (x,y) plane. Along such a line, the distance between a pixel and each of its 8-neighbour is assumed to be 1 [9]. In the sequel, we will denote by N the number of points provided by the tablet, by M the number of points obtained at the end of the preprocessing step, by Λ the 8-connected line, and by x'(m) and y'(m) (m=1..M) the coordinates on the (x,y) plane of the points belonging to Λ. Note that $M >> N$, where the magnitude of the inequality depends on the resolution of the tablet and on the writing speed. The interpolation is achieved by connecting any pair of successive data points P=(x(n),y(n)) and Q=(x(n+1),y(n+1)) along the shortest digital path between them. The algorithm for finding the shortest digital path between two points finds the digital points by incrementing x or y - depending on which variable exhibits the largest variation from P to Q - and then computing the value for the other variable according to the equation of the analog interpolating line. Thus, the coordinates of each interpolating point between P and Q depend on both the coordinate of P and Q. Moreover, since in an 8-connected digital plane the maximum distance between two successive points after the interpolation is 1, the two signals x'(m) and y'(m) shall be much more correlated than the original x(n) and y(n).

After this preprocessing, the aim of the first step of our method is that of performing a spatial frequency analysis of Λ in order to obtain a multi-scale representation of the original curve. The above mentioned properties of the sequences x'(m) and y'(m) allow to perform such an analysis separately on the two one-dimensional components rather than on the whole two-dimensional ink. To this purpose the Discrete Fourier Transforms (DFT) X(K) and Y(K) (K=1..*M*) of the sequences x'(m) and y'(m) are computed. At each scale, the desired representation of the original curve is obtained by applying the Inverse Discrete Fourier Transform (IDFT) to the first *T* elements of the sequences X(K) and Y(K): the smaller the value of *T*, the coarser the approximations of Λ. *T* ranges in [3,*N*], because 3 is the minimum number of points to define a curve and *N* is the original number of points provided by the tablet. At the end of this step we obtain different representations Λ_i (i=1..*N*-2) of the original curve containing a number of points ranging from 3 to *N*. Figure 1 shows the original ink and its multiscale representation obtained by applying the above mentioned procedure. The second step of our method is devoted to estimate the curvature of each representation. To this purpose, the arclength representation of each representation Λ_i is computed [10]. This representation is a function $\alpha(\lambda)$ where λ is the curvilinear abscissa of a point, and α is the angle of the tangent to the curve at that point with respect to the horizontal axis. The length of the curve is normalized, so that at each scale λ ranges between 0 and 1, and partitioned in *N* intervals of the same size. In this way, independently of the actual length of each curve Λ_i, the arclength representations have the same number of points. The T values $\alpha(\lambda)$ measured on the curve are mapped into the corresponding intervals, while the remaining *N-T* ones are obtained by linearly interpolating the values measured on Λ_i they lie in between. Thus, the multiscale representation assumes the form of a two-dimensional array *A*, with *N-2* rows (one for each scale) and *N* column (one for each point of the finest resolution).

The third step is devoted to build the feature map. In our case, the feature we want to estimate is the curvature along the ink. Center-surround operation is implemented by finding the local curvature at each scale. Figure 2a)-b) depict a word and a graphic rendering of its feature map.

The last step is aimed at building a saliency map, that in our case assumes the form of a one-dimensional array reporting the *N* average values $<\alpha(\lambda)>$ of the curvature detected in the previous step. Figure 2c illustrate the saliency map associated to a word.

3. Saliency-based scale selection

As mentioned before, we want to describe the shape of each stroke by means of its curvature. Intuitively, we would like to select the representation corresponding to the lowest resolution at which the less salient - but still "relevant" - changes of curvature are still detectable.

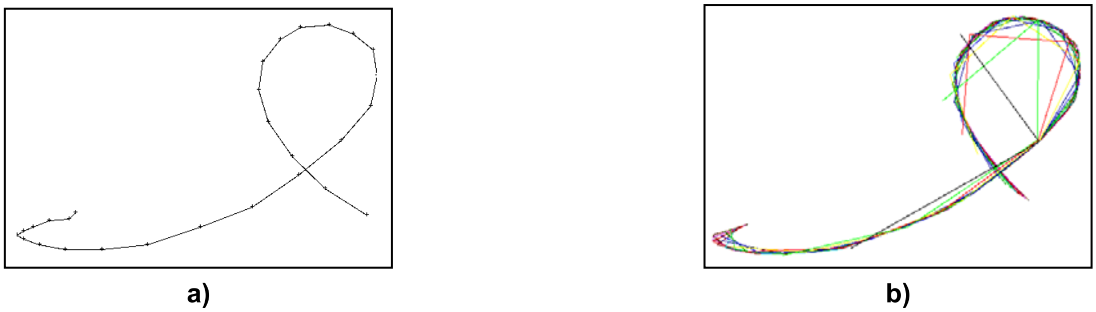
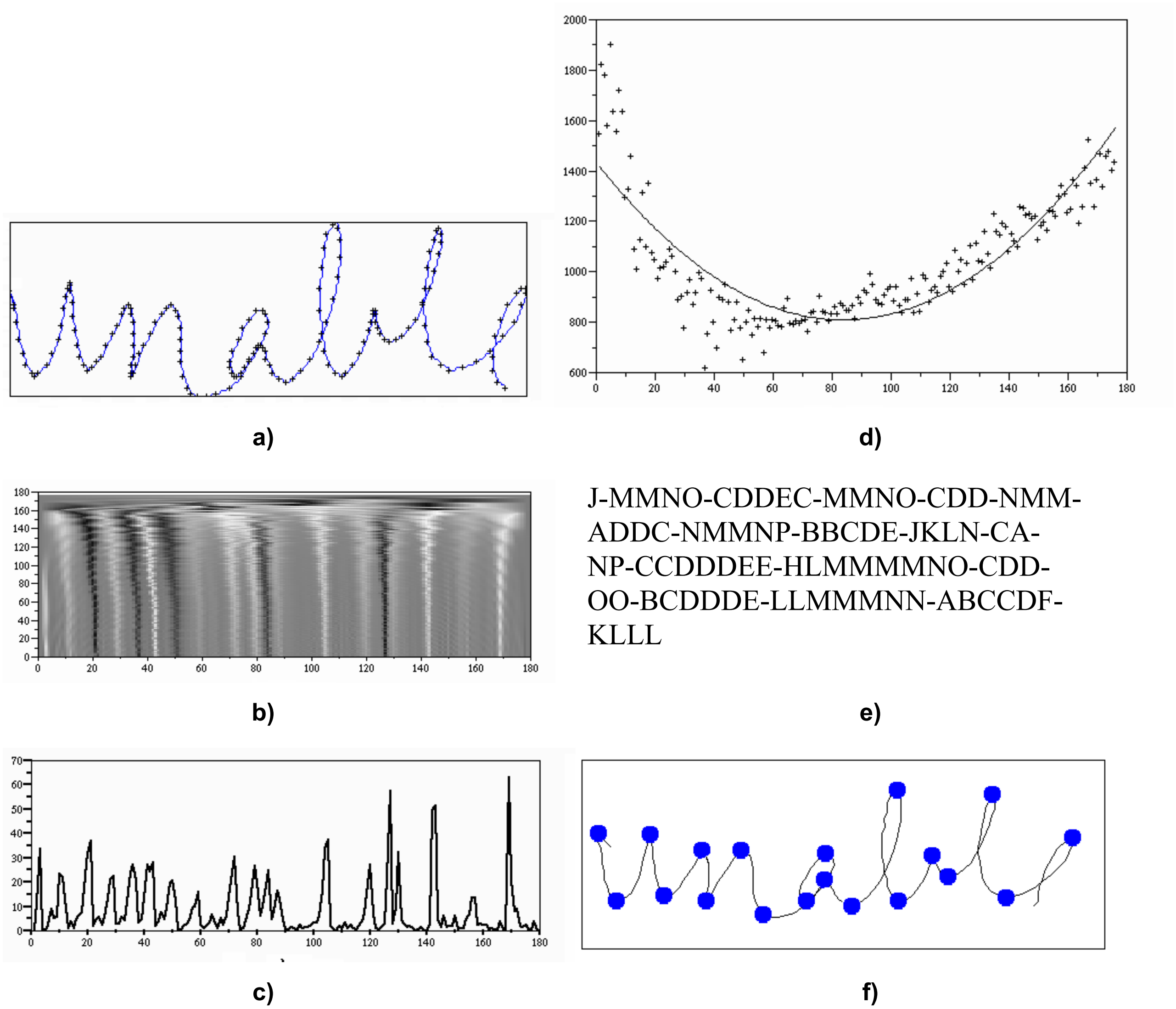


Figure 1. The multiscale representation of the ink. a) The original ink, containing 28 points. b) The inks at the 26 different resolutions. The coarsest reconstruction of the ink, Λ_1, appears as made of two segments forming an almost right angle, while the finest one, Λ_{26}, is almost indistinguishable from the original ink. Note that the shortening of the ink due to truncation becomes apparent only at very coarse scales.



a) d)

J-MMNO-CDDEC-MMNO-CDD-NMM-ADDC-NMMNP-BBCDE-JKLN-CA-NP-CCDDDEE-HLMMMMNO-CDD-OO-BCDDDE-LLMMMNN-ABCCDF-KLLL

b) e)

c) f)

Figure 2: The proposed method. a) The original input. b) A graphical rendering of the multiscale representation: at each scale, the darkest/brightest points correspond to most salient curvature changes. c) The saliency map: peaks correspond to most salient changes of curvature as measured across the scales. d) The histogram of the differences between the curvature at each scale and the saliency map. It is also shown the best fitting parabola: the selected resolution correspond to the vertex of such a parabola. e) The string encoding the change of curvature along the curve at the selected resolution: hyphens correspond to segmentation points. f) The obtained segmentation of the ink.

Selecting a representation corresponding to a lower resolution would be too coarse, and therefore would possibly hide some relevant features, while one corresponding to a higher resolution would be too fine, and therefore too sensitive to non-salient changes in the shape. Then, once this representation has been selected, the final description of the handwriting shape would be given in terms of a set of features extracted from the curvature of the ink at the selected resolution.

The algorithm for selecting the most suitable representation exploits the saliency map. In particular, for each representation Λ_i, it computes the distance between the vector $\alpha(\lambda)$ and $<\alpha(\lambda)>$, i.e. the difference between the curvature observed at that scale and the saliency map. According to the model, such difference should be very high in correspondence of the lowest resolutions, get smaller as far as the resolution approaches the "right" one and then increase again as the resolution becomes too high. Therefore, to select the most suitable representation we find the best fit of the distances with a parabola and select the scale ν corresponding to the vertex of the parabola. Once the scale ν has been selected, the shape of the handwriting is described by the direction of corresponding curve α(λ), which has been already computed in the second step of the algorithm described in the previous Section. The actual values of α(λ) are quantized into 16 intervals and each interval encoded by one of the letter of the subset [A-P] in such a way that the letter A corresponds to the first interval (from 0 to 2π/16), the letter B to the second one (from 2π/16 to 2∗2π/16) and so on, counter clockwise. By this encoding, then, the shape of the word is described by a string of characters that represents the desired set of features. Figures 2d shows the histogram of the distances along the scale-space and the best fitting parabola, while figure 2e reports the string encoding the shape of the word.

4. Experimental results

The main purpose of the algorithm described in the previous Section, as already noted, is that of finding the best compromise between the two conflicting aims of any representation: hiding “non relevant” changes, i.e. changes that appear in samples belonging to the same class and, at the same time, preserving “relevant” changes, i.e. those exhibited by samples belonging to different classes. Therefore, we have designed a set of experiments to show that the features extracted from the proposed representation are very stable with respect to “non relevant” changes, as to allow simple but very performing implementations of many tasks of interest for handwriting recognition. In this study, we have considered stroke segmentation and feature matching. The experiments reported below have been carried out by using a set of 1,000 words produced by the same writer, provided by the Handwriting Recognition group at IBM T.J. Watson Research Center. Each word was manually segmented into strokes by three different subjects and only those points on which there was agreement among at least two experts were retained as actual segmentation point. Let us explicitly note that 99.86% of the actual segmentation points were agreed upon by all the experts.

The first experiment was aimed at performing the segmentation of the handwriting into strokes. This is achieved by parsing the strings encoding the shape of the word and locating a segmentation point whenever the lexical distance between the labels of two successive strokes is larger than 1. Note that the lexical distance d(P,A)=d(A,P)=1. The results obtained while processing the string in fig. 2e are shown in figure 2f. The figure has been obtained by locating on the original curve the arc corresponding to a point at the scale ν, and locating the segmentation point in correspondence of the extreme of the arc that exhibits the sharpest variation with the following/preceding arc of the curve. This last variation is easily computed by looking at the feature string. In order to provide a quantitative evaluation, we have assumed that a segmentation point provided by the algorithm was correctly located if it was located within the arc of the original curve delimited by the location of the experts' points. Under this assumptions, the algorithm correctly located 99.23% of the actual segmentation points.

In the second experiment, the strings encoding the shape of the words were compared by means of a string matching algorithm that exploits the observation that long strokes, typically ascenders and descenders, play an important role in driving the recognition process [11]. The string matching algorithm uses as input both the unsegmented strings S1 and S2, and the segmented ones, SS1 and SS2, respectively. It starts by searching for the longest common substring (LCS) with a gap of two between S1 and S2. The algorithm then assume that there is a match among all the strokes of SS1 and SS2 that are included, even partially, in LCS. In other words, the matching is "extended" at the stroke level. Then, the matching strokes are logically removed from both the unsegmented and the segmented strings and the algorithm search for the next LCS, and so on. The algorithm stops when either there are no more matching strokes, or the LCS includes only fragments of single strokes. The results of the algorithm for feature matching show that the algorithm succeed in providing similar description for similar shapes, as hypothesized.

Eventually, to provide quantitative estimation of the results, and because the database does not contain many instances of the same word, we have collected another database made of 240 words produced by 6 different writers, collected by using a Wacom PL 100V tablet with a cordless stylus and a sampling rate of 100 Hz. Each writer was required to drawn 10 times a set of 4 words without any specific instruction or model to adhere. Table I reports the results obtained on our database. Each entry in the table reports, for each word and for each writer, the number of words correctly decomposed (according to the human expert), and the number of different descriptions for that word, computed by assuming that two strings were considered as the same if there were at most two different symbols for each stroke. This choice for the string matching follows from the observation that the anticipatory effects mainly influence the beginning and the end of the strokes and therefore the first and the last symbols of each string. Let us note that the worst performance is obtained on the word “nothing” produced by the writer #5. The large number of different descriptions, however, is due to two different allographs for the letter “g” used by that writer, as well as to two different sequences used to draw the bar of the letter “t”. The former drawback should be dealt somehow during the classification, while an algorithm for overcoming the latter has been proposed in [12]. When such a reordering algorithm is applied, the number of different descriptions depends only on the number of different allographs currently used by the writer for the character composing the word.

5. Concluding remarks

We have propose a method for selecting the best scale at which the electronic ink of the handwriting should be considered in the following steps of the recognition process. The method has been derived from an analogy with the primate visual system. According to this analogy, multiscale curvature maps are computed at different levels of resolution and arranged into a pyramidal structure. The saliency map is eventually obtained by averaging those features values across all possible scales. Such a map enjoys the property of exhibiting higher values in correspondence of region of the ink whose curvature stands out from their surroundings on a larger number of scales. Following this approach, thus, the best representation is that corresponding to the scale at which the observed curvature changes are the closest, according to a given metric, to the saliency map.

The experiments conducted till now, despite the simplicity of the feature extraction and the string matching algorithm, confirm that the proposed method provides a representation that leads to very stable and consistent results, and therefore seems a promising way to represent handwriting in order to extract basic, writer-specific writing units.

Further developments will address the problem of using such basic writing units to perform the recognition of any words that can be produced by means of such writing units, thus encompassing the disadvantages of both holistic and analytical methods for cursive handwriting recognition. As with respect to holistic methods, it should be possible to recognize any word composed by means of such writing units, not only the ones belonging to a given, size limited dictionary. Approaches based on this idea have been recently proposed [13], and they will certainly benefits from a better feature matching, as the one provided by our method. Similarly, analytical methods may benefit as well from the segmentation and description of the individual strokes provided by our method, because those strokes, rather than the individual characters of the alphabet, constitute the graphic alphabet of a writer, by means of which his handwriting is produced.

Table I. Results on our data base

	#1	#2	#3	#4	#5	#6
but	10(2)	9(2)	10(1)	10(2)	10(1)	10(1)
they	10(2)	10(3)	9(3)	9(1)	9(2)	10(2)
have	10(3)	10(2)	10(2)	10(2)	9(1)	10(2)
nothing	9(3)	9(2)	9(3)	10(1)	9(4)	9(1)

References

[1] R. Plamondon and F.J. Maarse , “An Evaluation of Motor Models for Handwriting”, *IEEE Trans. on Systems, Man and Cybernetics*, *19,* 1989, 1060-1072.

[2] A. Alimi and R. Plamondon, “Performance Analysis of Handwritten Strokes Generation Models”, *Preproc. of 3th Int. Workshop on Frontiers in Handwriting Recognition - IWFHR III*, Buffalo, NY (USA), 1993, 272-283.

[3] R. Plamondon, “A kinematic theory of rapid human movements. Part I: Movement representation and generation”, *Biological Cybernetics, 72*, 1995, 297-307.

[4] F. Atteneave, “Some informational aspects of visual perceptions”, *Physic. Review*, 61, 1954, 183-193.

[5] M.A. Fischler and R.C. Bolles, “Perceptual organization and curve partitioning”, *IEEE Trans. on PAMI, 8(1),*1986 , 100-105.

[6] S. Marshall, “Review of shape coding techniques”, *Image and Vision Comp.,* 7(4), 1989, 281-294.

[7] L. Itti, C. Koch and E. Niebur, "A Model of Saliency-based Visual Attention for Rapid Scene Analysis", *IEEE Trans. on PAMI, 10(12),* 1998, 1254-1259.

[8] X. Li, M. Parizeau, R. Plamondon, “Segmentation and reconstruction of on-line handwritten scipts”, *Pattern Recognition*, vol. 31, no. 4, 1998, pp. 675-684.

[9] A. Rosenfeld, A. C. Kak, *Digital Picture Processing*, Academic Press, 1982

[10] T. Pavlidis, *Structural Pattern Recognition*, Springer-Verlag, 1977.

[11] R.Plamondon and S.N. Shrihari, “On-line and Off-line Handwriting Recognition: A Comprehensive Survey”, *IEEE Trans. on PAMI, 12(8),* 2000 , 787-808.

[12] G. D’Andria, C. De Stefano, R. Foglia and A. Marcelli, “An algorithm for handwriting strokes reordering”, *Proc.11th Conf. of the Int. Graphonomics Soc. – IGS2003*, Scottsdale, AZ (USA), 2003, 237-240.

[13] A. El-Nasan and G. Nagy, “Ink-Link”, *Proc. Int. Conf. on Patt. Rec. – ICPR’02*, Quebec-city, PQ (CANADA), 2000, 573-576.