

# Off-line signature recognition based on dynamic methods

Juan J. Igarza\*, Inmaculada Hernáez, Iñaki Goirizelaia, Koldo Espinosa, Jon Escolar  
Dept. of Electronics and Telecommunications, University of the Basque Country  
Alameda Urquijo s/n, Bilbao, Spain E48013

## ABSTRACT

In this paper we present the work developed on off-line signature verification as a continuation of a previous work<sup>1</sup> using Left-to-Right Hidden Markov Models (LR-HMM) in order to extend those models to the field of static or off-line signature processing using results provided by image connectivity analysis. The chain encoding of perimeter points for each blob obtained by this analysis is an ordered set of points in the space, clockwise around the perimeter of the blob. Two models are generated depending on the way the blobs obtained from the connectivity analysis are ordered. In the first one, blobs are ordered according to their perimeter length. In the second proposal, blobs are ordered in their natural reading order, i.e. from the top to the bottom and left to right. Finally, two LR-HMM models are trained using the (x,y) coordinates of the chain codes obtained by the two mentioned techniques and a set of geometrical local features obtained from them such as polar coordinates referred to the center of ink, local radii, segment lengths and local tangent angle. Verification results of the two techniques are compared over a biometrical database containing skilled forgeries.

**Keywords:** Off-line signature, LR-HMM, blob analysis.

## 1. INTRODUCTION

Left-to-Right Hidden Markov Models<sup>2</sup> is a well-known technique used in behavioral biometrics, especially in speaker recognition and dynamic or on-line signature verification, where acquired information is a set of values, ordered in time. In on-line signature identification systems LR-HMMs are trained using time ordered sequences of points acquired during the act of signing, including coordinates, pressure, pen tilt and more features extracted from them<sup>3,4</sup>. However, in off-line systems only the 2D image of the signature is available, and there is no time reference. The novel method here proposed consists on the generation of space ordered sequences of points from off-line signatures applying results provided by image connectivity analysis<sup>5</sup>, on extracting from them geometrical local features and on using both the points and derived geometrical features as input to the LR-HMM models.

Connectivity analysis determines which pixels in an image are interconnected and belong to the same object or region. The purpose of connectivity analysis is to separate multiple objects in a scene from each other and from areas of the image that represent noise or extraneous things. The connected components are called *blobs*, and they can be defined as a cluster of adjacent pixels of the same nature (gray level, or color in a binary image) which usually represents an object or region in the field of the view.

The connectivity algorithm gives as a result a description of an image as a linked list of blobs that are hierarchically structured, providing father-child or sibling-sibling relationships. For each blob a set of geometrical features are obtained such as its area, color, center of gravity or centroid, maximal and minimal coordinates or bounding box, moments of inertia or perimeter length. The processing of those results is known as blob analysis<sup>5</sup>, a very useful tool in machine vision for image analysis of parts. The most important geometrical feature for our purposes is the chain encoding of perimeter points for each blob, since it is an ordered set of points in the space, clockwise around the perimeter of the blob.

---

\* Contact author: Juan J. Igarza, email: jtpiguj@bi.ehu.es; phone: +34 946014124 fax: +34 946014259. Departamento de Electrónica y Telecomunicaciones, Escuela Técnica Superior de Ingeniería de Bilbao, Alameda Urquijo s/n, Bilbao, Spain E48013

It is clear that depending on how the order of blobs is chosen, the final sequence of perimeter points will be different and, as a consequence, final results can change dramatically. We discuss two different ways of generating the models depending on the way the blobs obtained from the connectivity analysis are ordered. In the first proposed method (here referred as LEN), blobs whose father is the background are ordered according to their perimeter length, from the longest one to the shortest one. Their children, if any, are inserted behind each father following the same criterion, i.e. from the longest one to the shortest one. The final sequence of space ordered points is the result of concatenating the perimeter of each blob as described. In the second proposal (referred as NAT), the main blobs or parts are ordered following their natural reading order, i.e. from top to bottom and from left to right and their children are inserted behind each father following the same criterion.

As an example, Fig. 1 a) shows the NAT criterion applied over the same signature as Fig. 1 b), where the LEN criterion was applied. Both pictures include a numbering of the blobs according to the used criterion. Blobs whose father is the background are numbered with a single digit and labeled in blue (dark), while child blobs depending on them are labeled in red (gray) and numbered with two digits, the first one referencing to its father and the second one to sibling order. Blobs show different numbers depending on the sorting criterion.

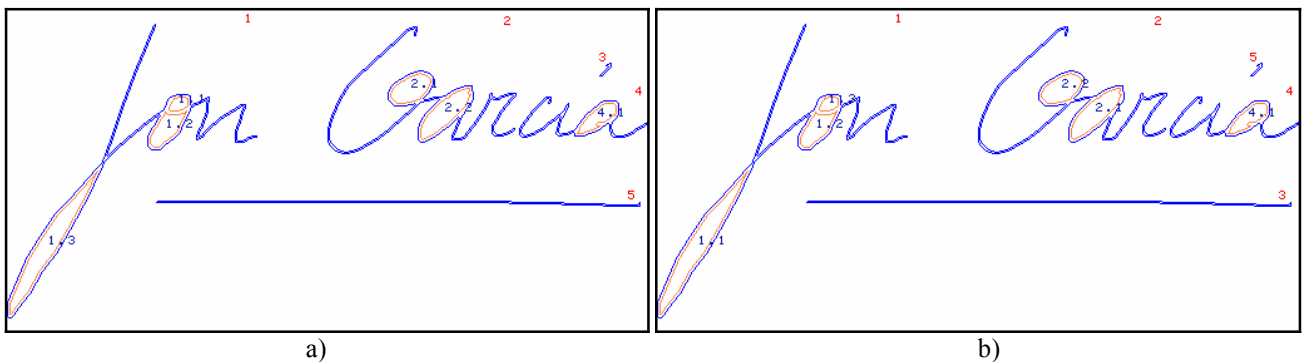


Fig. 1 Blobs of the same signature ordered according to a) NAT and b) LEN criteria. Main blobs are labeled in blue (dark) with a single digit in red. Child blobs depending on them are labeled in red (gray) with two digits in blue, the first one referencing to its father and the second one to sibling order. The final images have been enlarged to allow the reading of the child blobs identification numbers.

This paper is organized as follows: Section 2 describes the preliminary system developed in previous work and gives details about the signatures database, the preprocessing of the signatures and the training of the models. Section 3 explains the added new geometrical features extracted from the perimeter points. Finally, section 4 presents the values obtained, the resulting conclusions from them, and future work.

## 2. DESCRIPTION OF THE PRELIMINARY SYSTEM

In a preliminary system<sup>1</sup> we studied the influence of the number of states of the HMM models and well as the different possible normalizing values for the lengths of the perimeter's chain of points. The first step in the method was to normalize in 2D size all the signatures in the database. Then the connectivity analysis was performed over them to obtain the description of the perimeter of the signatures. After that, two different lists of (x,y) points were generated for each signature: one according to NAT criterion and the other according to LEN. Both chains of points were normalized in length and used as inputs to different HMM with different topologies (different number of states and possible transitions between states). These steps are described in detail in this section.

### 2.1. Signatures database

To perform the experiments, we used a database containing a total amount of 3750 signatures corresponding to 75 individuals. Half of the signatures (25 for each signer), were genuine. The rest were skilled forgeries, 25 for each signer done by 5 forgers. The forged signatures stored in the database were skilled forgeries, where every forger tried several times to forge the target signature. Only when forgers considered that his forgery was good enough the signature was acquired and stored, providing so a high level of quality to the forgeries.

The database, called *MCYT baseline Corpus*<sup>6</sup>, is a multimodal database that also includes fingerprints acquired by using both optical and capacitive sensors and the voice of some signers. The signatures were acquired on-line using a graphic tablet with a spatial resolution of 100 points per millimeter and a sampling frequency of 100Hz to capture the (x, y) coordinates, the inking pen pressure on 1024 levels, inclination and azimuth. Simultaneously the signatures were written on a paper with normal ink. After the acquisition phase the signatures were scanned to obtain 2D static images suitable for off-line processing. In this way, both on-line and off-line signature recognition can be accomplished for all of the signers in the database.

The signatures of this database have been obtained with black ink on white sheets of paper, making the segmentation process easier. Fig. 2 a) shows a scanned signature using 600 optical points per inch (without interpolation) and 256 grey levels. Part b) of Fig. 2 shows the same signature at 512x256 pixels obtained from the on-line signature, and eliminating any dynamic feature such as pressure level, inclination, azimuth and spatial or time sequence information.

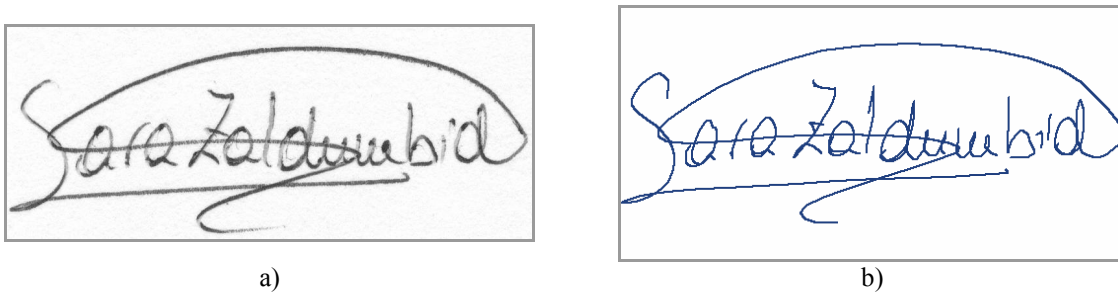


Fig. 2 a) Original signature of an individual of the database scanned at 600 ppi and b) his corresponding static image of 512 x 256 obtained by eliminating the dynamic information. Both images have been enlarged to twice their real size.

The images extracted from the background are normalized to a maximum size of 512 pixels wide x 256 pixels high and they are centre justified to those dimensions. The signatures with horizontal trend will show empty rows (see Fig. 2 b) and those with vertical trend will show empty columns (see Fig. 3). Each image so generated keeps its conversion scale factor from pixels to millimeters in the bmp file header and the aspect ratio is always 2:1 for all the images (see Fig. 2 b and Fig. 3).

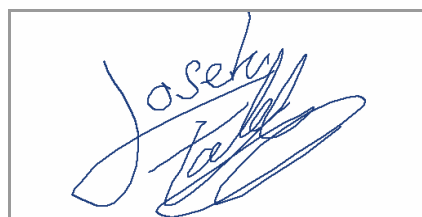


Fig. 3 The signature of this individual has been justified to the 2:1 format, horizontally centered adding empty columns to the left and to the right. The image has been enlarged by 2 relative to its natural size.

## 2.2. Preprocessing of the signatures

The preprocessing stage of the signatures consists on its segmentation from the background and on performing a specific normalization to avoid size fluctuations (every time we sign we get different dimensions for the signatures). Once the normalized binary image is obtained, the analysis is carried out in two steps. First a connectivity analysis is performed.

This can be done by using the static 512 x 256 pixels sized signatures. Secondly the perimeter points are extracted from the signature blobs, by analyzing the set of blob structures that constitute the binary image of the signature.

After the connectivity analysis, we get one linked list of (x, y) points or chain-code per blob describing its perimeter. For each blob, this chain of (x, y) points is covered clockwise starting with the most up-righted point in the blob and finishing at the same starting point. The chains of (x, y) points corresponding to the perimeters of the blobs, both for part blobs and their children, were the biometric information relevant to the preliminary system.

### 2.3. Training of the models

As mentioned in the introduction, we discuss two different ways of generating the models depending on the way the blobs obtained from the connectivity analysis were ordered. For both criteria, the final sequence of (x, y) points is the result of concatenating the perimeter of each blob, joining one linked list of perimeter points to the next linked list following NAT or LEN criterion. It can be thought a priori that the NAT order is more adequate to the Latin writing in which the signatures have been captured.

For every criterion we have generated one HMM representing the signatures of one user. One image will always give the same blob structure, so the difference in the modeling will be given by the different criteria used in the sorting of the blob perimeter points. Fig. 1 a) shows the NAT criterion applied over the same image as Fig. 1 b), where the LEN criterion was applied. Both pictures include a numbering of the blobs according to the used criterion. In both figures there are 5 parts or main blobs ordered as follows: “Jon”, “Gar”, acute accent over the vowel ‘i’, “cia” and the underlining flourish using NAT criterion, and “Jon”, “Gar”, the flourish, “cia” and acute accent for LEN criterion. The three children of “Jon” have a different order if NAT or LEN criterion is used. The same applies to the two children of “Gar”. The blob “cia” has a sole child, so it is not affected by the ordering criteria. Even though in this particular example the same concatenation numbering is obtained for both models (1+<sub>1.1+1.2+1.3</sub>+2+<sub>2.1+2.2</sub>+3+4+<sub>4.1</sub>+5) the blobs associated to the numbers differ for both criteria and consequently, the resulting chains are different. However, usually the numberings obtained by each criterion do not show any coincidence.

As was mentioned in the previous section, a total of 25 signatures by user is available. The signatures are numbered from 00 to 24 according to the time of the acquisition. The first five signatures (00 to 04) are used for the initial training of the HMM. Four additional signatures (05, 10, 15 and 20) are chosen to re-estimate the initial model. As a result 16 genuine signatures and 25 forgeries are finally available for training and testing.

An important point to consider when training an HMM is the topology selection and the length of the sequences used to generate them. Different topologies were tested using different number of states and different kind of state transitions. In order to analyze the possibility of getting an improvement by using normalization, experiments were performed by training the models with free input length, but also using normalized length values 750, 1000, 1500, 2000 and 5000.

Table 1 EER percentage rates obtained for the five normalizing values and free length in the training of the LR-HMMs with different number of states and later verification tests for both NAT and LEN criteria.

Normalizing length	5 states		6 states		7 states		8 states	
	NAT	LEN	NAT	LEN	NAT	LEN	NAT	LEN
Free	-	-	30.41	32.92	33.13	34.34	-	-
750	-	-	30.24	32.47	28.05	30.77	30.36	32.73
1000	31.46	32.83	29.51	33.05	28.34	31.67	30.03	32.33
1500	32.32	34.59	29.63	31.59	<b>27.58</b>	<b>30.37</b>	31.52	32.94
2000	-	-	29.73	31.97	29.78	32.27	31.65	32.67
5000	-	-	30.90	33.29	33.13	34.34	-	-

Table 1 shows the EER percentage values obtained for the experiments in the preliminary system, and the results can be compared to the values obtained without using any normalization for LR-HMMs with different number of states. It can be seen that NAT criterion produces better results than LEN criterion in all the experiments, independently of the number

of states of the models and the value of the normalizing length. The models obtained using the LEN criterion never give EER values better than 30%, while the models obtained using the NAT criterion give better results with 6 and 7 states, reaching the best rates for 7 states and using 1500 normalized samples.

### 3. ADDED GEOMETRICAL FEATURES

New features have been added for every point  $(x_i, y_i)$  belonging to the perimeter of each blob: polar coordinates  $(\rho_i, \theta_i)$ , local radius of curvature  $R_i$ , and local arc length  $L_i$ , and local tangent line angle  $tg_i$ . Following is a brief description of the algorithms used to obtain these features.

#### 3.1. Polar coordinates $(\rho_i, \theta_i)$ from the center of ink of the signature

Polar coordinates are calculated from the reference system centered in the center of gravity  $(x_{cg}, y_{cg})$  or center of ink of the signature. The center of gravity of the signature is calculated according to (1), being  $(x_j, y_j)$  any inked point of the image and  $N$  the total number of inked points in the image. Then (2) is applied to obtain  $(\rho_i, \theta_i)$ .

$$x_{cg} = \frac{1}{N} \sum_{j=1}^N x_j \quad y_{cg} = \frac{1}{N} \sum_{j=1}^N y_j \quad (1)$$

$$\rho_i = \sqrt{(x_i - x_{cg})^2 + (y_i - y_{cg})^2} \quad \theta_i = \arctan((y_i - y_{cg}) / (x_i - x_{cg})) \quad (2)$$

#### 3.2. Local radius $R_i$

Local radius  $R_i$  of each perimeter point  $(x_i, y_i)$  is defined as the radius of the longest arc of length  $L_i$ , that can be seen from that point, and that includes all the points of the perimeter from the point  $(x_i, y_i)$  to the point  $(x_{i+L_i}, y_{i+L_i})$ . The algorithm to calculate this parameter is shown in Fig.4.

For each perimeter point we define three points whose initial values are in a first iteration, the point itself or initial point  $(x_i, y_i)$ , the next perimeter point or intermediate point  $(x_{im}, y_{im})$  and the next one to this or last point  $(x_f, y_f)$ . Each iteration is used to calculate whether all the intermediate points between the initial and last points,  $(x_i, y_i)$  and  $(x_f, y_f)$ , belong to the same segment, either line or arc segment. If that is the case, we move the last point to the next one, the new intermediate point is calculated and a new iteration is performed. This process is repeated until an intermediate point not belonging to the line or arc segment is found. Then the valid radius is the one calculated in the previous iteration.

Local radii of curvature  $R_i$  can be obtained solving the geometrical problem of the center  $(x_{cc}, y_{cc})$  of the circle across three points. This problem has no solution when denominators of the formulae (3) and (4) are 0, i.e. when the three points are aligned; meaning that  $R_i$  will be infinite. When denominators are not 0 a one and only solution will be obtained for the center of curvature  $(x_{cc}, y_{cc})$  and as a consequence,  $R_i$  will be the Euclidean distance between  $(x_{cc}, y_{cc})$  and any of the three points.

$$x_{cc} = \frac{+(x_{im}^2 + y_{im}^2 - x_i^2 - y_i^2)(y_f - y_{im}) - (x_f^2 + y_f^2 - x_{im}^2 - y_{im}^2)(y_{im} - y_i)}{2((x_{im} - x_i)(y_f - y_{im}) - (y_{im} - y_i)(x_f - x_{im}))} \quad (3)$$

$$y_{cc} = \frac{-(x_{im}^2 + y_{im}^2 - x_i^2 - y_i^2)(x_f - x_{im}) + (x_f^2 + y_f^2 - x_{im}^2 - y_{im}^2)(x_{im} - x_i)}{2((x_{im} - x_i)(y_f - y_{im}) - (y_{im} - y_i)(x_f - x_{im}))} \quad (4)$$

$$R_i = \sqrt{(x_i - x_{cc})^2 + (y_i - y_{cc})^2} \quad (5)$$

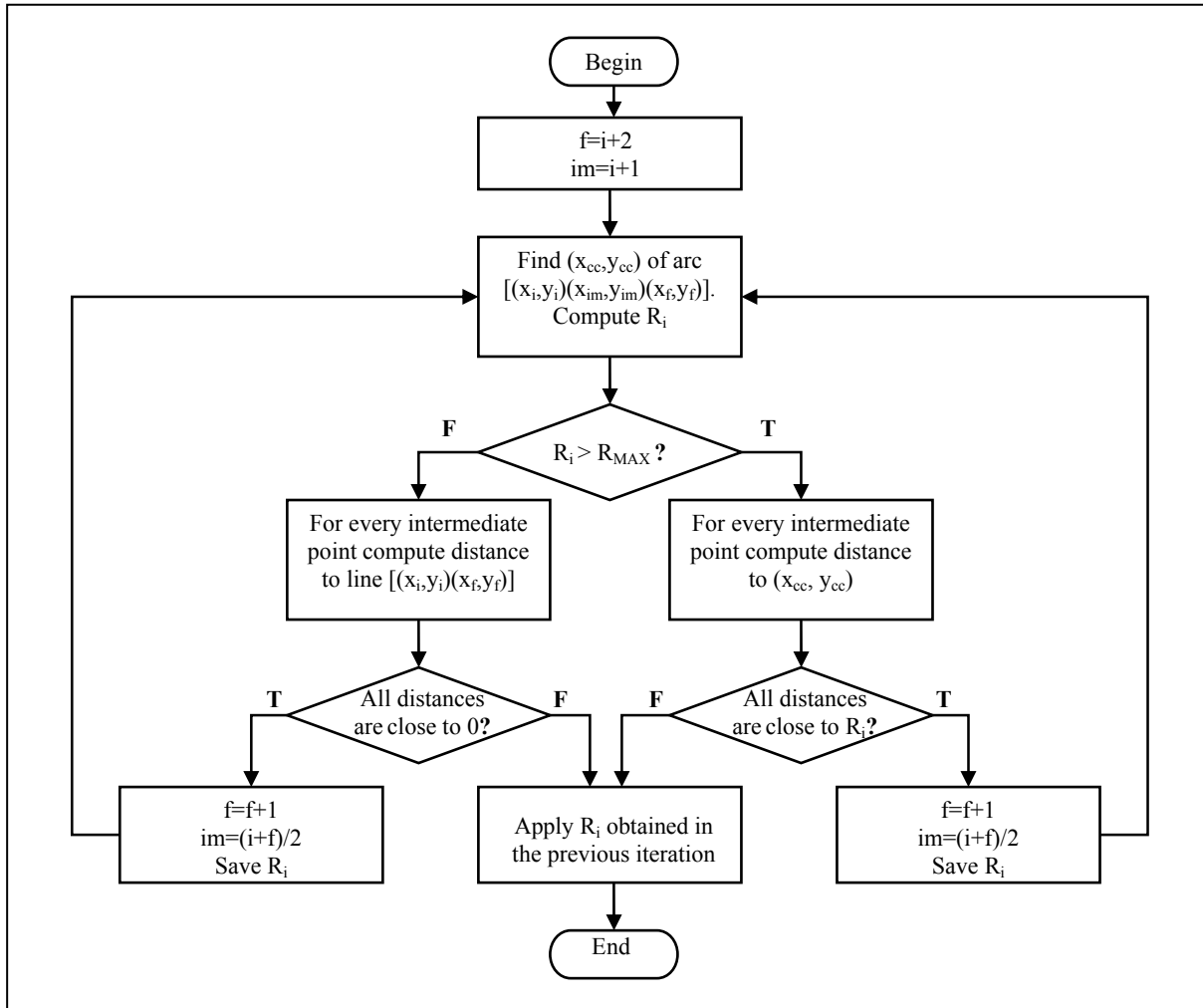


Fig.4 Algorithm to obtain the local radius  $R_i$  of each perimeter point  $(x_i, y_i)$ . At the end of the algorithm the segment length  $L_i$  for the same perimeter point can be computed as a simple subtract of the final point and the initial one,  $f - i$ . Tangent line angle  $t_{g_i}$  can be obtained as the angel of line  $[(x_i, y_i)(x_f, y_f)]$  when  $R_i > R_{MAX}$  or as a the angel perpendicular to the line defined by the perimeter point and the center of curvature  $(x_{cc}, y_{cc})$  if  $R_i < R_{MAX}$ .

If the value of the radius  $R_i$  corresponding to the arc defined by three points is infinite or higher than a predefined threshold  $R_{MAX}$ , one can think that it is a line segment and therefore, in order to know whether all the points between the initial and last point are aligned with the line segment, their distance to the line defined by the two points  $[(x_i, y_i)(x_f, y_f)]$  is calculated. That distance should be almost zero and in any case smaller than a predefined tolerance value.

On the other hand, if the value of the radius  $R_i$  is not infinite, the coordinates of the center of curvature  $(x_{cc}, y_{cc})$  are calculated. Then the distance of each point between the initial point and the last one to the center of curvature  $(x_{cc}, y_{cc})$  is calculated. If all the distances are similar to the value of the radius  $R_i$ , within a tolerance value, we may conclude that all points are aligned in the arc defined by the three points  $[(x_i, y_i)(x_{im}, y_{im})(x_f, y_f)]$ .

The solution during the first iteration of the algorithm is trivial since all the intermediate points are aligned either in a line segment or an arc segment, as there is just one intermediate point. As new iterations of the algorithm are executed, it may happen that a line segment becomes an arc segment and vice versa, because in each new iteration new points are

taken into account, and the value of  $R_i$  might be modified. Finally when the last iteration fails there is always a previously calculated value that will be the applicable.

### 3.3. Segment length $L_i$

Segment length  $L_i$  is calculated simultaneously with the local radius  $R_i$ . Its value is the total number of points in pixels between the initial point  $(x_i, y_i)$  and last point  $(x_f, y_f)$  of the arc or line segment.

$$L_i = f - i \quad (6)$$

### 3.4. Local tangent line angle $tg_i$

Local tangent line angle  $tg_i$  is also calculated simultaneously as the previous two features. In the case of a line segment, its value is given by the tangent of the line defined by the initial and last points as in (7):

$$tg_i = \arctan((y_f - y_i)/(x_f - x_i)) \quad (7)$$

On the other hand, if it is an arc segment, the angle is perpendicular to the line defined by the center of curvature  $(x_{cc}, y_{cc})$  and the initial point, as shown in equation (8). The sign in the equation depends on the sense of the intermediate point  $(x_{im}, y_{im})$  view from de initial one  $(x_i, y_i)$ .

$$tg_i = \arctan((y_i - y_{cc})/(x_i - x_{cc})) \pm \frac{\pi}{2} \quad (8)$$

## 4. OBTAINED VALUES AND FUTURE WORK

In order to evaluate the efficiency of the new added features we have tested the system by running experiments both with skilled forgeries, and random or casual forgeries. Casual forgeries are those where the impostors or forgers don't know any details about the original signature, and therefore they are not skilled forgeries. In order to evaluate verification efficiency while testing casual forgeries all the genuine signatures of the data base belonging to the rest of signers are considered as forgeries, i.e. any genuine signature which does not belong to a person is considered as a random forgery for that person.

The new features described in the previous section have been used progressively to feed the LR-HMMs. Models with 7 states and using 1500 points as normalized length value were selected, following the results described in Table 1 and both methods of ordering the blobs, NAT and LEN, have been tested. Table 2 shows the EERs obtained using different sets of geometrical features, always using the same HMM topology.

Table 2 EER percentage rates obtained for NAT and LEN models using skilled and casual forgeries after training and verification with different geometrical features.

Trained Features	NAT		LEN	
	Skilled	Casual	Skilled	Casual
$x_i, y_i$	27.58	20.08	30.37	22.63
$\rho_i, \theta_i$	36.40	25.79	35.17	23.07
$x_i, y_i, \theta_i$	29.45	18.19	30.62	20.76
$x_i, y_i, \rho_i$	29.43	18.44	30.45	20.95
$x_i, y_i, \rho_i, \theta_i$	27.94	16.32	29.58	18.45
$x_i, y_i, \rho_i, \theta_i, R_i$	27.81	15.89	31.00	18.56
$x_i, y_i, \rho_i, \theta_i, L_i$	27.60	17.47	29.28	17.95
$x_i, y_i, \rho_i, \theta_i, R_i, L_i$	28.33	17.41	30.15	18.20

The first line of Table 2 shows the results of previous experiments using only  $(x_i, y_i)$  coordinates, and corresponds to the best values shown in Table 1. Values obtained using only polar coordinates  $(\rho_i, \theta_i)$  centered in the center of ink are shown in the second line, and mixing of  $(x_i, y_i)$  coordinates with other geometrical features are shown below them.

When comparing values of Table 2 with results obtained in previous work in Table 1 one can observe that Equal Error Rates EERs provided by the new models, are still better for NAT than for LEN models, i.e., NAT models based on perimeter where the reading natural order is used for insertion provide better results than models based on length insertion.

EERs improve as new features are added to the models only for casual forgeries while improvement is small or none for skilled forgeries. Intramodal<sup>7</sup> fusion of the two off-line experts based on LEN an NAT models and intermodal<sup>7</sup> fusion of off-line experts and our on-line expert based on  $(x, y)$  alignment<sup>8</sup>. is being performed. More work has to be done on the way the geometrical information is given to the system i.e., a more elaborated interpretation of the blob perimeters should be done, identifying perimeter chains related to strokes, in order to improve the results.

### ACKNOWLEDGEMENTS

This work is supported by the “Ministerio de Ciencia y Tecnología” of the Spanish Government under projects MCYT TIC2000-1669-C04-03 and MCYT TIC2003-08382-C05-03.

### REFERENCES

1. J.J. Igarza, I. Hernáez, I. Goirizelaia, K. Espinosa “Applying dynamic methods in off-line signature recognition” *Biometric Technology for Human Identification SPIE Bellingham WA*, Proceedings of SPIE **Vol. 5404** , pp. 418-424, 2004.
2. L.R. Rabiner, B.H. Juang “An Introduction to Hidden Markov Models”, *IEEE Acoustics, Speech & Signal Processing Magazine*, **3**, pp. 4-16, 1986.
3. L.Yang, B.K.Widjaja, R. Prasad. “Application of Hidden Markov Models for Signature Verification”. *Pattern Recognition*, **28(2)**, pp. 161-170, 1995.
4. J.J. Igarza, I. Goirizelaia, K. Espinosa, I. Hernáez, R. Méndez, J. Sánchez “On-line Handwritten Signature Verification Using Hidden Markov Models” *Progress in Pattern Recognition, Speech and Image Analysis Springer-Verlag Berlin Heidelberg*, LNCS **2905** , pp. 391-399, 2003.
5. G.J. Agin, in *Handbook of Industrial Robotics*, 1st edition, edited by Shimon Y. Nof (John Wiley & Sons, New York, 1985), Chap. 14 Vision Systems, pp. 231-261.
6. J. Ortega-Garcia, J. Fierrez-Aguilar, D. Simon, J. Gonzalez, M. Faundez-Zanuy, V. Espinosa, A. Satue, I. Hernaez, J.J. Igarza, C. Vivaracho, D. Escudero, Q.I. Moro. “MCYT Baseline Corpus: A Bimodal Biometric Database”. *IEE Proceedings-Vision, Image and Signal Processing. Special Issue on Biometrics on the Internet*, **150(6)**, pp. 395-401, 2003.
7. J. Kittler, K. Messer and J. Czyz, “Fusion of Intramodal and Multimodal Experts in Person Identity Authentication Systems”, Proc. COST 275 Workshop on “The Advent of Biometrics on the Internet”, Rome, Italy, pp. 17-24, Nov 2002.
8. J.J. Igarza, L. Gómez, I. Hernáez, I. Goirizelaia “Searching for an Optimal Reference System for On-line Signature Verification based on  $(x, y)$  Alignment” *ICBA 2004 Biometric Authentication First International Conference Springer-Verlag Berlin Heidelberg*, LNCS **3072** , pp. 519-525, 2004.