BIOMETRIC AUTHENTICATION BY OFFLINE SIGNATURE IMAGES USING HIDDEN MARKOV MODEL

G.Precilla Jansi Rani[1], Dr.M.Pushpa Rani[2]

[1]Scholar, [2] Prof & Head Mother Teresa Women's University, Kodaikanal, (India)

ABSTRACT

During information technology era, where fast retrieving information is obtained using computer networking and applications, such as banking systems and border security. Offline Signature verification and validation is the process of identifying the given signatures and verifying those signatures to differentiate the original from the forged ones, by using some mathematical procedures, generated by some pattern recognition algorithm. Mostly passwords and PIN codes are very easily to be forgotten. But unlike those passwords and PIN codes, Signatures are very hard to be forgotten or even simulated by others. For that reason a system of authenticating the signature has been extensively used by the people as a secured way of identification. This paper presents an approach to find a method for signature authentication system by using Hidden Markov model. A hidden Markov model (HMM) is one in which you observe a sequence of emissions, but do not know the sequence of states the model went through to generate the emissions. Analyses of hidden Markov models seek to recover the sequence of states from the observed data. A publicly available database GPDS with Hidden Markov Model (HMM) as a Classifier is an idea of this paper to implement authentication of Handwritten Signature.

Keywords: SVS, HMM

I. INTRODUCTION

Signature verification methods can be performed in two ways; off-line signature verification mode and on-line signature mode. Off-line verification methods depend on the features that can be extracted from still images of the handwritten Signature which is already available. On the other hand, in the on-line methods, the signature is verified in real time while the person is signing [2]. With the help of the lowest error rate as a result of the verification system is the biggest challenge of accepting or rejecting an offline signature.

Basically, there are two approaches are available for Offline Signature Verification: 1. Pseudo Dynamic 2. Pseudo Static. Pseudo dynamic approach involves imperceptive or insensitive characteristics; therefore it is very inflexible to imitate. While static approach involves perceptive or sensitive characteristics, and therefore it is easy to imitate those signatures.

Many approaches are used for signature verification, including elastic image matching, Neural Networks and Euclidean Distance Classifier. This research proposes an off-line signature verification and validation system for handwritten signatures; by using Hidden Markov Model.

Static/Offline signatures derive as handwritten images on documents and by definition do not contain any dynamic information. This lack of information makes static signature verification systems significantly less reliable than their dynamic counterparts. This study involves extracting dynamic information from static images, while the signature was created. With the help of a hidden Markov model from the static image and match it to the dynamic version of the image the signature authentication would be done.

II. METHODOLOGY

Handwritten signature verification and recognition system is mainly divided into following steps.

- Data Acquisition
- Preprocessing
- Feature Extraction
- Classification

2.1 Data Acquisition

This is the phase we used to collect the scanned images of the signature collection by the scanner in order to have the digital form of the images for the processing.

2.2 Image Preprocessing

The purpose in this phase is to make signatures standard where the enhancement of the image is needed for the entire system and to be organized for feature extraction. Preprocessing steps can be invoked by using the following operations,

III. FILTERING

The removal of noises in the images will give the expected output.

IV. MORPHOLOGICAL OPERATIONS

Morphological operations are used to extract image components that are useful in the representation and description of region shape, such as

- Boundary Extraction
- Skeletonization
- Convex Hull
- Filtering
- Thinning
- Pruning
- Dilation
- Erosion

The preprocessing phases include the following steps: Background Elimination, Conversion of image to Binary, Image Resizing, Noise Reduction, Width Normalization and Skeletonization etc.

4.1 Background Elimination

The removal of the background of the test signature image will give much clear display.

4.2 Conversion of Image to Binary

Gray scale of the test signature image should be converted into binary image for making the feature extraction process easier.

4.3 Image Resizing

Standard size of the image is needed when signatures are in different sizes.

4.4 Noise Reduction

Reducing the noises if any presents in the test signature

4.5 Skeletonization

Making the given image into a distinct pixel width is called as Skeletonization. The creation of a skeleton of the image can be obtained by applying some iterative process in the object, layer by layer until only the object is spine, which form the skeleton remains, this iterative process is called thinning.

4.6 Thinning

A thinning algorithm contains a set of pixel deleting conditions, which enable it to erode the object iteratively. The skeletonization steps are as follows.

Step 1: mark all the points of the signature that are candidates for removing (black pixels that have at least one black with 8-white neighbor and at least two black with 8-white neighbors).

Step 2: Examine one by one pixel following the contour lines of the signature image, and remove them as their removal will not cause a break in the resulting pattern.

Step 3: If at least one point was deleted then go back to Step 1 and repeat the process once again. Skeletonization makes the extracted features invariant to image characteristics like the qualities of the pen, the paper, the signer, the digitizing method and quality. The Skeletonization process supports the following properties, so that the thinning result can be characterized as a skeleton of the 2D binary object.

This system starts with image preprocessing. The above diagrams illustrate various image processing steps involved in the Signature Verification System. In this step the noise is removed to eliminate unwanted information that negatively influences accuracy of verification and validation. Next, we perform a registration step where the signature is scaled into an appropriate form to gain better and accurate result, after that, the shifting operation invoked using center of gravity (COG) to determine the centric of the signature.

(a) (b)

Fig: (a) Image with Noises (b) Image after the removal of Noises.

(a) (b)

Fig: (a) Image before Cropping (b) Image after Cropping.

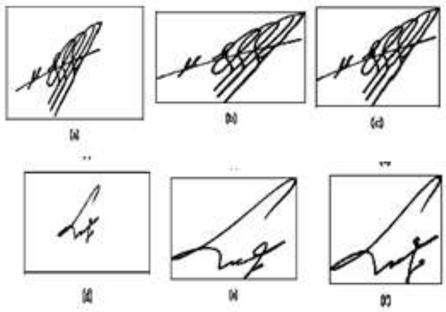

Figure 4: (a, d) original samples, (b, e) samples resized to maximum width and height of all persons, (c, f) samples resized to maximum height and width of the same person.

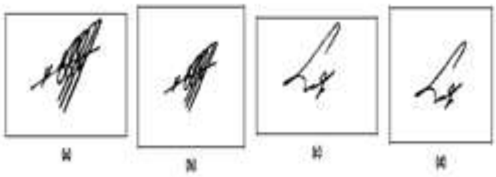

Figure 5: Shifting using COG, (a, c) before shifting, and (b, d) after shifting

Figure 6: Rotation step, (a) before rotation (b) after rotation

After applying shifting operation, the rotation is performed to align the signature to the correctly position. Experimental results show that the proposed system has high accuracy compared with other systems. Generally extracting the dynamic information from the static handwritten images can have useful applications.

Our approach constructs a Hidden Markov Model (HMM) from the static image. An HMM is a probabilistic model that models a time-dependent sequence of events with a sequence of states having transitions between them. In our case, the HMM describes the pen trajectory that created the image. Each state has an associated PDF, embedding geometric shape information of the static image. The HMM topology specifies the interconnection of states. Transitions between states are weighted with transition probabilities to dictate possible pen movements between static image coordinates.

Normally, both the state PDFs and the transition probabilities are obtained through a training process. Although training is possible for this application as well, data scarceness is a serious problem and we chose to specify the PDFs and transition probabilities in advance. It is not able to completely resolve ambiguities in regions with multiple intersections. The problem is due to a loss of context caused by the use of first-order HMMs: transition probabilities depend only on the current state. Higher-order HMMs, whose transition probabilities depend not only on the current state but also on the previous states, are much better equipped to take context into account. Usually, higher order HMMs tend to be computationally expensive. In this study, however, we use a second-order HMM with sparse transition probability matrices, reducing the computational cost to a manageable level. The suitable second-order HMM that is derived from a basic first-order HMM. Further, context is provided by comparing not only pen positions but also local line directions.

V. THE HIDDEN MARKOV MODEL

An HMM has N emitting states {q1,q2, . . . , qN} that have observation PDFs associated with them. The two states q0 and qN+1, without associated PDFs, are called nonemitting states. These two additional nonemitting states serve as initial and terminating states, respectively, thus eliminating the need for separate initial and terminating probabilities (see [34] for more detail).

All state observation PDFs in the context of this paper are spherical Gaussians, described by

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{D}{2}}\sigma} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{x}-\boldsymbol{\mu})^T(\mathbf{x}-\boldsymbol{\mu})\right), \qquad (1)$$

where x is a D-dimensional vector that must be matched to the PDF and μ is the D-dimensional mean of the Gaussian.

The standard deviation σ is preset. For brevity, the PDF associated with state i having mean μi and standard deviation σ will be referred to as N(μi, σ). Geometric shape information of the static image is embedded in the PDF parameters μi and σ, as described in Section 3.2.

States are connected by transition links that dictate the possible pen movements. All transitions between states are weighted with transition probabilities. The order of the HMM specifies the number of previous states the HMM considers when transiting to a next state. Sections 3.2-3.6 describe how the order of our HMM is increased to take context into account.

In order to match a static image and a dynamic exemplar, the dynamic exemplar is presented as a sequence of quantifiable characteristics called feature vectors. The sequence is given by X =[x1, x2, . . . , xT], where xt denotes a D-dimensional feature vector at discrete-time instant t and T is the number of feature vectors (number of samples in the dynamic exemplar). Using the Viterbi algorithm, X is matched to our HMM to produce a hidden state sequence s =[s1, s2, . . . , sT], which estimates the desired sequence of skeleton samples, as described in Section 3.7.

5.1 First-Order HMM

The shorthand notation for an HMM
λ is

λ={A,{N(μi,σ), i=1, . . .;N}}, (2)

where A is a matrix representing the transition links and N(μi,σ), as described by (1), is the observation PDF of state I for I Є {1,…..N}.

We begin by constructing a first-order HMM from the skeleton of the static image. The skeleton consists of M unordered samples {p1, p2, . . . , pM}, where px is the 2D coordinate of sample x. Each emitting state i is associated with a skeleton sample via a mapping r(i) and the sample coordinates are embedded in the observation PDF by setting $\mu i = p_{r(i)}$. For a first-order HMM, we have N = M and r(i)= i. Our first-order HMM matches only 2D feature vectors, in this case, the pen positions of the dynamic exemplar. We choose σ= 0.7 pixels in (2) for all states, in order to relate the match between the position coordinates of the dynamic exemplar and static image to Euclidean distance.

The HMM topology is crucial to our algorithm, as it constrains the range of possible pen movements that could generate the static image. For our first-order HMM, the probability of reaching the next state depends only on the current state, so that the transition probability matrix A =[aij], where aij = P(st+1 = qj|st = qi) is the probability of a transition from qi to qj at instance t + 1, with I, j Є{0, 1, . . .,N + 1} and t Є {1, 2, . . . , T – 1}.

HMM states are called neighbors if their associated skeleton samples are adjacent. All emitting states are linked to their neighbors, to allow the pen to move to an adjacent skeleton point on a transition. However, this only takes local information into account, and not context. Context is incorporated by using second-order HMMs, which allow us to include a directional feature, as described in Section 3.3.

Since we have no prior knowledge of where the pen trajectory of the static image may start or end, the non-emitting initial state can enter any emitting state. Also, each emitting state is directly connected to the non-emitting terminating state. One also needs elasticity in the model, to allow the static image and dynamic exemplar to have different numbers of samples. This is accomplished by including skip-links and self-loops in the HMM. A skip-link is a transition between two states separated by a neighbor common to both.

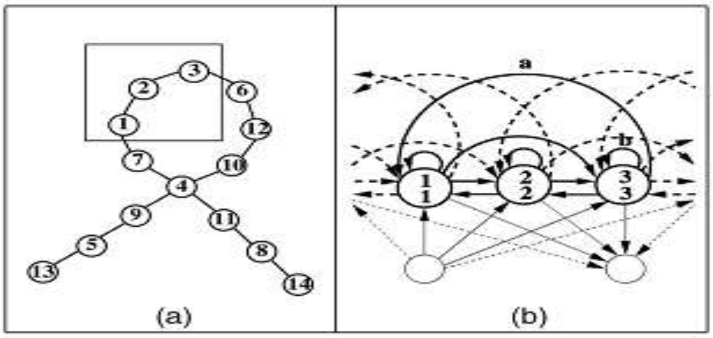

Fig. 4

(a) Isolated unordered skeleton samples in a simplified signature

(b) their corresponding first-order HMM.

A self-loop connects a state back to itself. Self-loops are added to the emitting states. In this paper, we use skip-links to skip states with only two neighbors. Equal transition probabilities are assigned to all transition links leaving a state, normalized to sum to one.

We emphasize an important feature of Fig. 4b. Any two neighboring emitting states and any two emitting states connected by a skip-link are connected both ways: If one enters the HMM at an emitting state with more than one neighbor it is not possible to determine locally in which direction one should move next and, therefore, both directions are allowed. Since all transition links are assigned the same probability, all skeleton samples are potential turning points. It is therefore entirely possible and it indeed happens, in practice, that the extracted pen trajectory may incorrectly reverse direction. One way to address this problem is to include more context. Bengio and Frasconi [35], supported by the experiments of Abou-Moustafa et al. [36], investigated the effect of topology on the ability of an HMM to learn context.

They showed that the addition of hidden states with a sparse connectivity can increase the ability of a Markov model to learn long-term dependencies and reduce the diffusion of context. The topology of our first-order HMM is ergodic with a sparse connectivity. When using second-order HMMs, we include extra states and the connectivity becomes even sparser in a natural way, as discussed in the next section.

Thus, in accordance with Bengio and Frasconi, we improve the ability of the HMM to model context.

VI. SECOND-ORDER HMMS AND THEIR FIRST-ORDER EQUIVALENTS

In order to take past context into account, we use second order HMMs. We have shown that the transition probabilities of first-order HMMs only depend on the current state, so that $a_{ij} = P(s_{t+1} = q_j|s_t = q_i)$. The transition probabilities of second-order HMMs depend on the current and previous states. The probability of a transition from state j to state k, given that state j is preceded by state i, becomes $a_{ijk} = P(s_{t+1} = q_k|s_{t-1} = q_{i,St=}q_j)$. Second-order HMMs can then be reduced to first-order equivalents to simplify their implementation, by using the Order Reducing (ORED) algorithm [34], [37]. These ideas are illustrated in Fig. 5. The HMM fragment in the figure forms part of a larger HMM. We only consider the transitions between the visible states. The second-order HMM in Fig. 5b is formed by extending all transitions of the first-order HMM in Fig. 5a to second-order connections (the order of the transitions is encoded in the subscripts of the transition probabilities). We do not show second-order connections depending on states outside of the HMM fragment shown.

The basic idea behind the ORED algorithm is to reduce an Rth-orderHMMto its (R - 1)th-order equivalent, by creating states for all pairs of connected states in the Rth-order HMM. Applying this procedure recursively, we reduce an HMM of arbitrary order to its first-order equivalent [34], [37]. The first order equivalent of the second-order HMM of Fig. 5b is shown in Fig. 5c. It should be noted that new first-order transition probabilities are created from the second-order probabilities. Bottom numbers, in general, now label the state PDFs which in our application translate to the skeleton indexes inherited from the first-order model. States k and m (top numbers), for example, are created in Fig. 5c from the connected pairs j_k and k_m in Fig. 5b, with the same PDFs (bottom numbers) as states k and m. They are connected by a $d_{km} = a_{jkm}$, so that one can interpret a_{jkm} as a first-order transition probability. In general, M different pairs of connected states in the Rth-order model result in approximately M states in the (R - 1)th-order model. The pairs kk, jk, ik, ij, km, and xi in Fig. 5b, for example, are connected and become states n, k, o, j, m, and i in Fig. 5c, respectively, where x can be any state connected to i via a dashed line.

The order reduction significantly increases the number of states. An Rth-order model with N states reduces to an equivalent first-order model with $O(N^R)$ states. However, it should be noted that this expansion does not increase the number of free parameters. Tied PDFs, which are PDFs shared by more than one state, are evaluated only once and only the original number of transition probabilities need to be considered. Therefore, the ORED algorithm does not affect processing requirements. It is shown by [38] that memory requirements are not affected either. Computational cost depends on transition probabilities, as discussed in the final section. The computational cost of our proposed algorithm is manageable, as our transition probability matrix remains sparse. This avoids redundant calculations. In the following sections, we show how to use the flexibility of higher-order HMMs to model handwritten scripts, starting with line segments. All second-order HMMs will hereafter be represented by their first-order equivalents, without loss of generality.

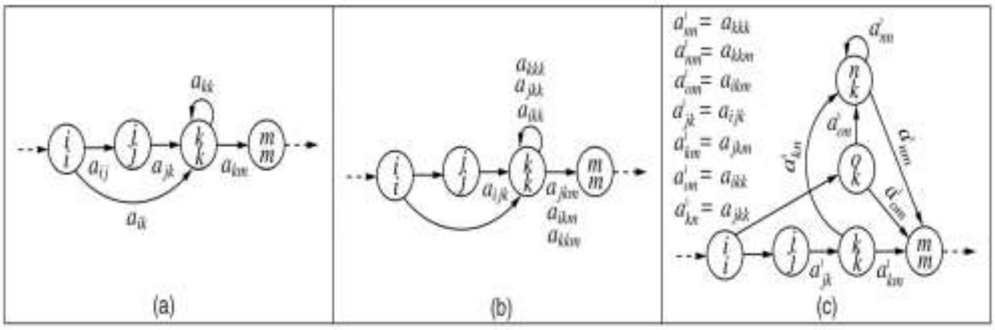

Fig. 5. (a) A first-order HMM Expanded to (b) a Second-Order HMM, and (c) the First-Order Equivalent of (b)

Static signatures originate as handwritten images on documents and by definition do not contain any dynamic information. This lack of information makes static signature verification systems significantly less reliable than their dynamic counterparts. This study involves extracting dynamic information from static images, specifically the pen trajectory while the signature was created.

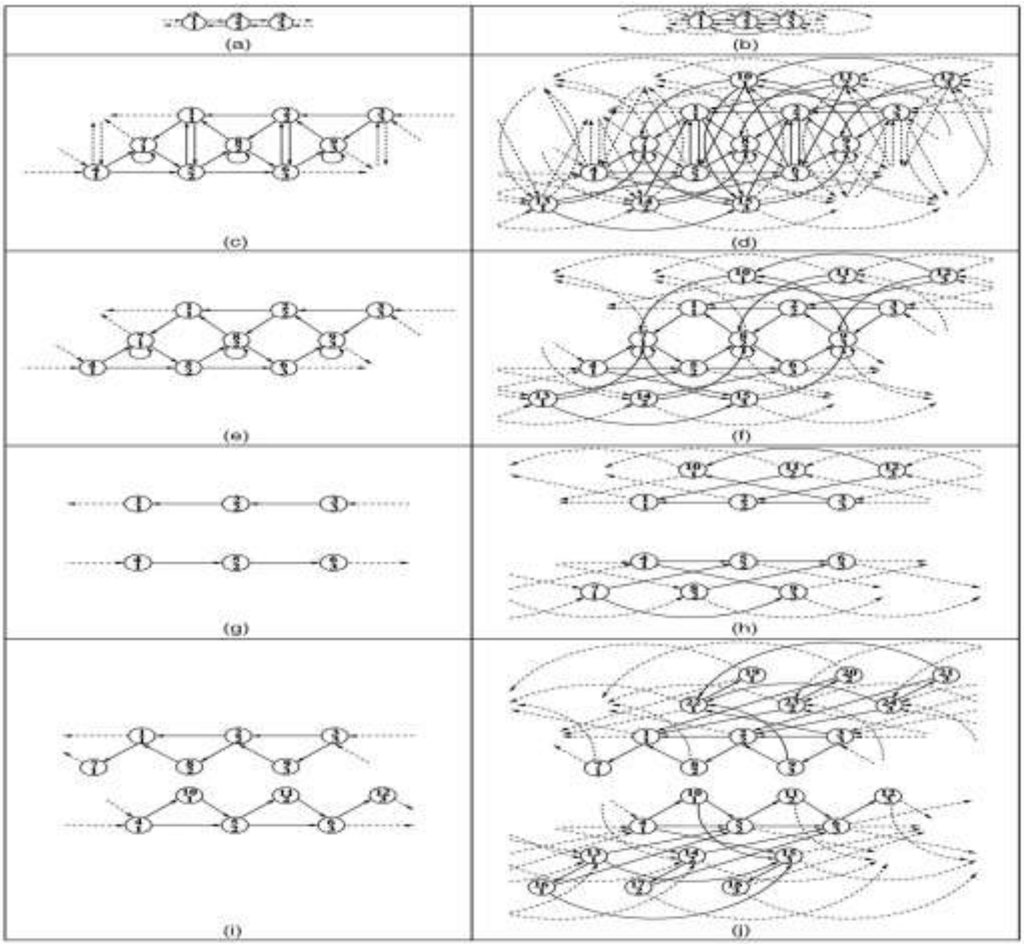

Fig. 6. The Simplified HMM Topology for a Line Segment in the Left-Hand Column with a Detailed Version in the Right-Hand Column when Skip-Links are Added to our first-order HMM.

(a) and (b) First-order HMMs.

(c) and (d) Second-order HMMs.

(e) and (f) Assigning the cost function.

(g) and (h) Removal of self-loop states.

(i) and (j) Inclusion of duration states

*Reference: Estimating the Pen Trajectories of Static Signatures Using Hidden Markov Models
Emli-Mari Nel, Johan A. du Preez, IEEE, Member, and B.M. Herbst
Ieee Transactions On Pattern Analysis And Machine Intelligence, Vol. 27, No. 11, November 2005 1733

VII. CONCLUSION

This system is robust and can detect random, simple and semi-skilled forgeries but the performance deteriorates in case of skilled forgeries. Using a higher dimensional feature space and also incorporating dynamic information gathered during the time of signature can also improve the performance. The concepts of Hidden Markov Model hold a lot of promise in building systems with high accuracy. We assume that a dynamic version of the static image is available (typically obtained during an earlier registration process). We then derive a hidden Markov model from the static image and match it to the dynamic version of the image.

REFERENCES

[1] R. Plamondon and S.N. Srihari, "Online and Offline Handwriting Recognition: A Comprehensive Survey," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 1, pp. 63-84, Jan. 2000.

[2] T. Huang and M. Yasuhara, "A Total Stroke SLALOM Method for Searching for the Optimal Drawing Order of Off-Line Handwriting," Proc. IEEE Systems, Man, and Cybernetics, vol. 3, pp. 2789-2794, Oct. 1995.

[3] Y. Kato and M. Yasuhara, "Recovery of Drawing Order from Single-Stroke Handwriting Images," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 9, pp. 938-949, Sept. 2000.

[4] K.K. Lau, P.C. Yuen, and Y.Y. Tang, "Stroke Extraction and Stroke Sequence Estimation on Signatures," Proc. Int'l Conf. Pattern Recognition, pp. 119-122, 2002.

[5] S. Lee and J.C. Pan, "Offline Tracing and Representation of Signatures," IEEE Trans. Systems, Man, and Cybernetics, vol. 22, no. 4, pp. 755-771, July 1992.

[6] G. Boccignone, A. Chianese, L.P. Cordella, and A. Marcelli, "Recovering Dynamic Information from Static Handwriting," Pattern Recognition, vol. 26, no. 3, pp. 409-418, 1993.

[7] P.M. Lallican and C. Viard-Gaudin, "A Kalman Approach for Stroke Order Recovering from Off-Line Handwriting," Proc. Int'l Conf. Document Analysis and Recognition, pp. 519-523, 1997.

[8] H. Chang and H. Yan, "Analysis of Stroke Structures of Handwritten Chinese Characters," IEEE Trans. Systems, Man, and Cybernetics B, vol. 29, no. 1, pp. 47-61, Feb. 1999.

[9] W. Huang, G. Rong, and Z. Bian, "Strokes Recovering from Static Handwriting," Proc. Int'l Conf. Document Analysis and Recognition, pp. 861-864, 1995.

[10] V. Govindaraju and R.K. Krishnamurthy, "Holistic Handwritten Word Recognition Using Temporal Features Derived from Off- Line Images," Pattern Recognition Letters, vol. 17, no. 5, pp. 537-540, 1996.

[11] I.S.I. Abuhaiba, M.J.J. Holt, and S. Datta, "Processing of Binary Images of Handwritten Text Documents," Pattern Recognition, vol. 29, no. 7, pp. 1161-1177, 1996.

[12] S. Ja¨ger, "Recovering Dynamic Information from Static, Handwritten Word Images," PhD dissertation, Univ. of Freiburg, 1998.

[13] S. Ja¨ger, "A Psychomotor Method for Tracking Handwriting," Proc. Int'l Conf. Document Analysis and Recognition, pp. 528-531, 1997.

[14] P.M. Lallican, C. Viard-Gaudin, and S. Knerr, "From Off-Line to On-Line Handwriting Recognition," Proc. Seventh Int'l Workshop Frontiers in Handwriting Recognition, pp. 303-312, 2000.

[15] Y. Kato and M. Yasuhara, "Recovery of Drawing Order from Scanned Images of Multi-Stroke Handwriting," Proc. Int'l Conf. Document Analysis and Recognition, pp. 261-264, 1999.

[16] Y. Al-Ohali, M. Cheriet, and C.Y. Suen, "Efficient Estimation of Pen Trajectory from Off-Line Handwritten Words," Proc. Int'l Conf. Pattern Recognition, pp. 323-326, 2002.

[17] Y. Al-Ohali, M. Cheriet, and C.Y. Suen, "Dynamic Observations and Dynamic State Termination for Off-Line Handwritten Word Recognition Using HMM," Proc. Eighth Int'l Workshop Frontiers in Handwriting Recognition, 2002.

[18] J.K. Guo, D. Doerman, and A. Rosenfeld, "Forgery Detection by Local Correspondence," Int'l J. Pattern Recognition and Artificial Intelligence, vol. 15, no. 4, pp. 579-641, 2001.

[19] K.K. Lau, P.C. Yuen, and Y.Y. Tang, "Recovery of Writing Sequence of Static Images of Handwriting Using UWM," Proc. Int'l Conf. Document Analysis and Recognition, pp. 1123-1128, 2003.

[20] L.R. Rabiner and B.H. Juang, "An Introduction to Hidden Markov Models," IEEE ASSP Magazine, pp. 4-16, Jan. 1986.

[21] L. Lam, S. Lee, and C.Y. Suen, "Thinning Methodologies—A Comprehensive Survey," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 14, no. 9, pp. 869-885, Sept. 1992.

[22] R.C. Gonzalez and R.E. Woods, Digital Image Processing. Addison- Wesley, 1992.

[23] M. Seul, L. O'Gorman, and M.S. Sammon, Practical Algorithms for Image Analysis: Description, Examples, and Code. Cambridge Univ. Press, 2000.

[24] R.C. Gonzalez, R.E. Woods, and S.L. Eddins, Digital Image Processing Using Matlab. Pearson Prentice Hall, 2004.

[25] J.J. Zou and H. Yan, "Skeletonization of Ribbon-Like Shapes Based on Regularity and Singularity Analysis," IEEE Trans. Systems, Man, and Cybernetics B, vol. 31, no. 3, pp. 401-407, June 2001.

[26] J. Rocha, "Perceptually Stable Regions for Arbitrary Polygons," IEEE Trans. Systems, Man, and Cybernetics B, vol. 33, no. 1, pp. 165- 171, Feb. 2003.

[27] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, Computational Geometry Algorithms and Applications, second ed. Springer, 1997.

[28] M.E. Munich and P. Perona, "Visual Identification by Signature Tracking," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 25, no. 2, pp. 200-217, Feb. 2003.

[29] P. Fra¨nti, A. Mednonogov, and H. Ka¨lvia¨inen, "Hough Transform for Rotation Invariant Matching of Line-Drawing Images," Proc. Int'l Conf. Pattern Recognition, pp. 389-392, 2000.

[30] O.R. Terrades and E. Valveny, "Radon Transform for Lineal Symbol Representation," Proc. Int'l Conf. Document Analysis and Recognition, pp. 195-199, 2003.

[31] A.K. Jain, Fundamentals of Digital Image Processing. Prentice-Hall, 1989.

[32] A.S. Rosenthal, J. Hu, and M.K. Brown, "Size and Orientation Normalization of On-Line Handwriting Using Hough Transform," Proc. Int'l Conf. Acoustics, Speech, and Signal Processing, p. 3077, 1997.

[33] R. Kapoor, D. Bagai, and T.S. Kamal, "A New Algorithm for Skew Detection and Correction," Pattern Recognition Letters, vol. 25, pp. 1215-1229, Aug. 2004.

[34] J.A. du Preez, "Efficient Training of High-Order Hidden Markov Models Using First-Order Representations," Computer Speech and Language, vol. 12, no. 1, pp. 23-39, Jan. 1998.

[35] Y. Bengio and P. Frasconi, "Diffusion of Context and Credit Information in Markovian Models," J. Artificial Intelligence Research, vol. 3, pp. 249-270, 1995.

[36] K.T. Abou-Moustafa, M. Cheriet, and C.Y. Suen, "On the Structure of Hidden Markov Models," Pattern Recognition Letters, vol. 25, no. 8, pp. 923-931, June 2004.

[37] J.A. du Preez and D.M. Weber, "The Integration and Training of Arbitrary-Order HMMs," Australian J. Intelligent Information Processing Systems, vol. 5, no. 4, pp. 261-268, 1998.

[38] J.A. du Preez, "Efficient High-Order Hidden Markov Modelling," PhD dissertation, Stellenbosch Univ. Press, 1997.

[39] P.Z. Peebles, Probability, Random Variables, and Random Signal Principles, third ed., McGraw-Hill, 1993.

[40] J.R. Deller, J.H.L. Hansen, and J.G. Proakis, Discrete-Time Processing of Speech Signals. IEEE Press, 2000.

[41] J.G.A. Dolfing, "Handwriting Recognition and Verification: A Hidden Markov Approach," PhD dissertation, Eindhoven, Netherlands, 1995.

[42] Y. Bengio, "Markovian Models for Sequential Data," Neural Computing Surveys, vol. 2, pp. 129-162, 1999.