Pergamon

*Pattern Recognition*, Vol. 31, No. 12, pp. 1849–1864, 1998
© 1998 Pattern Recognition Society. Published by Elsevier Science Ltd
All rights reserved. Printed in Great Britain
0031-3203/98 $19.00 + 0.00

PII: S0031-3203(98)00057-0

# A TRULY 2-D HIDDEN MARKOV MODEL FOR OFF-LINE HANDWRITTEN CHARACTER RECOGNITION†

HEE-SEON PARK‡ and SEONG-WHAN LEE*,§

‡Software Center, Corporate Technical Operations, Samsung Electronics Co., Ltd., Apkujong Bldg., 599-4 Shinsa-dong, Kangnam-ku, Seoul 135-120, South Korea
§Center for Artificial Vision Research, Department of Computer Science and Engineering, Korea University, Anam-dong, Seongbuk-ku, Seoul 136-701, South Korea

(*Received* 18 *December* 1997)

**Abstract**—In recent years, there have been several attempts to extend one-dimensional hidden Markov model (HMM) to two-dimension. Unfortunately, the previous efforts have not yet achieved a truly two-dimensional (2-D) HMM because of both the difficulty in establishing a suitable 2-D model and its computational complexity.

This paper presents a new framework for the recognition of handwritten characters using a truly 2-D model: hidden Markov mesh random field (HMMRF). The HMMRF model is an extension of a 1-D HMM to 2-D that can provide a better description of the 2-D nature of characters. The application of HMMRF model to character recognition necessitates two phases: the *training* phase and the *decoding* phase. Our optimization criterion for training and decoding is based on the maximum, marginal *a posteriori* probability. We also develop a new formulation of parameter estimation for character recognition. Computational concerns in 2-D, however, necessitate certain simplifying assumptions on the model and approximations on the implementation of the estimation algorithm. In particular, the image is represented by a third-order MMRF and the proposed estimation algorithm is applied over the look-ahead observations rather than over the entire image. Thus, the formulation is derived from the extension of the look-ahead technique devised for a real-time decoding.

Experimental results confirm that the proposed approach offers a great potential for solving difficult handwritten character recognition problems under reasonable modeling assumptions. © 1998 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Hidden Markov mesh random field (HMMRF) Off-line handwritten character recognition
Look-ahead technique Maximum marginal *a posteriori* probability

## 1. INTRODUCTION

The success of automatic speech recognition systems based on hidden Markov model (HMM)[1,2] has motivated recent attempts to apply similar methods to character recognition[3–6] and document image processing.[7] The central issue in adapting any type of speech technology to image analysis is that speech is a one-dimensional (1-D) signal whereas images are two-dimensional (2-D). In the case of HMMs, a majority of approaches to technology transfer have been focused on a subset of character recognition problems that may be viewed as actually one-dimensional. While encouraging results have been obtained on this problem, the characteristic that HMM is suitable for 1-D time series is unlikely to lead to a general framework for character recognition. Thus, a mathematical 2-D model of the images with finite number of parameters is desired.

More recently, there have been several attempts to extend the 1-D HMM to two-dimension. Unfortunately, the previous efforts[8–10] have not yet achieved truly 2-D HMM because of the difficulty in establishing a suitable 2-D model with reasonable computational complexity.

2-D Markov random field (MRF) model is a natural extension of 1-D autoregressive model to 2-D.[11] The MRFs have become more and more popular during the last few years in image processing.[12] A good reason for that is that such a model is the one which requires the less *a priori* information on the world model. In addition, the MRFs are well suited for representing the spatial continuity that is the characteristic of most images. In order to take the computational advantage of the recursive procedures in image processing, it is appropriate to model the image as a *causal MRF*[13] which is characterized by causal transition distributions. Since, however, there is no natural order in 2-D, the concept of causality needs refinement. There are two causal 2-D Markov chains in the literature; the Markov mesh random field

*Author to whom all correspondence should be addressed. E-mail: swlee@image.korea.ac.kr

†The preliminary version of this paper has been presented at the Third International Conference of Document Analysis and Recognition, Montral, Canada in August 1995. This research was supported by the Hallym Academy of Science, Hallym University and the 1997 Industry-University Co-Research Fund of Korea Ministry of Information and Communication.

(MMRF)[13] and the nonsymmetric half-plane (NSHP) Markov chain.[14] They differ in both their "past" and local state region as shown in Fig. 1.[15]

Of main interest within this paper is the MMRF, a sub-class of MRF models that was first proposed by Abend *et al.* in 1965.[13]

The MMRF model of images in image classification is very similar to the HMM of speech in speech recognition. An observed image is commonly modeled by two layers of stochastic processes: the observation process that describes the variation of the image and the MMRF that describes the statistical characteristics of the true image, just like the observation layer and the hidden layer of the HMM. We refer to this type of two-layered image model as the hidden Markov mesh random field (HMMRF) model. In an HMMRF model, the hidden layer is modeled by an MMRF that is characterized by initial and transition probabilities, and the observation layer is defined as a probabilistic function of the MMRF.

In this paper, we present a new framework for the recognition of handwritten characters using a truly 2-D model. Our work has been largely inspired by Devijver's work[16] for the modeling of digital images and image sequences using HMMRF model. In spite of the successful demonstration of the HMMRF model in the restoration, segmentation and modeling of static images, there has been no attempt to apply it to character recognition problem except for the attempt by the authors of this paper.[17] The main reason is that efficient parameter estimation algorithms such as the Baum–Welch algorithm[18] in 1-D HMM do not exist in 2-D HMMRF model. The application of HMMRF model to character recognition necessitates two phases: the *training* phase and the *decoding* phase. Our optimization criterion for training and decoding is based on the maximum, marginal *a posteriori* probability. The criterion can be viewed as an approximation to the maximum likelihood (MLE) criterion. We also develop a new formulation of parameter estimation for off-line character recognition. Computational concerns in 2-D, however, necessitate certain simplifying assumptions on the model and approximations on the implementation of the estimation algorithm. In this paper, the image is modeled as a third-order MMRF which is characterized by causal conditional distributions, and the proposed estimation algorithm is applied over the look-ahead observations rather than over the entire image. Thus, the formulation is derived from the extension of the look-ahead technique devised for a real-time decoding. We attempt to illustrate how the ideas of HMMRF model can be applied to the problems of off-line handwritten character recognition.

The paper is organized as follows. In Section 2 we give the motivations for this work and review the related works on off-line character recognition using HMM. In Section 3 HMMRF model is introduced and the two fundamental problems of the HMMRF model are discussed. Section 4 describes a decoding scheme which has been developed for an efficient estimation of the states. In Section 5, we raise the issue of parameter estimation and propose a new algorithm for estimation of the model parameters. In Section 6, we present experimental results demonstrating the effectiveness of the proposed estimation algorithm and the performance comparison of the proposed HMMRF-based approach with the HMM-based approaches. Finally, conclusions and further researches are discussed in Section 7.

## 2. RELATED WORKS

In this section, we review recent developments in character recognition in the framework of HMM (see Fig. 2).

HMMs have been widely used for automatic speech recognition,[1,2] and have proven successful in dealing

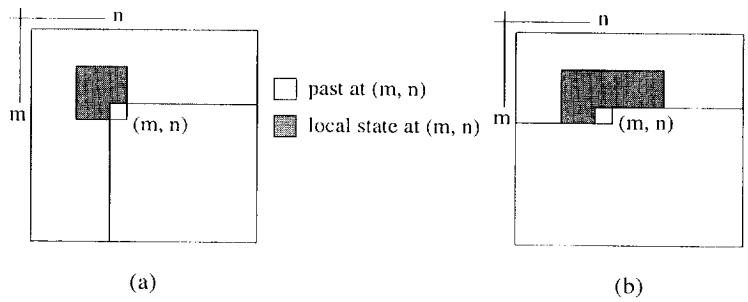



Fig. 1. (a) Regions of support of past and local state of MMRF. (b) Regions of support of past and local state of NSHP Markov chain.


Fig. 2. Evolution of HMM.

with the statistical and sequential aspects of speech signal. Based on its success in speech recognition, a question that arises naturally is how well these stochastic models would work on problems in character recognition.

Recently, there are many on-going researches to recognize text,[3,19] handwritten characters[4,20] or handwritten words[5,6,11] by using HMM-based approaches. While the recognition of handwritten words has a large amount of similarity with that of speech, there are also significant points of departure. The most notable of them is the 2-D nature of the character image. To model the images with HMM, we have to either recover the temporal information, or define a consistent sequence from static images. The former is known to be very difficult to solve using only heuristics on a raw image or a skeletonized image[21] because of numerous ambiguities just as in the problem of recovering 3-D information from 2-D image. The most conventional method of the latter approach is to split the word image into a sequence of overlapping vertical frames.[22] However, the frames may lose the spatial dependency of pixels in relative proximity of each other and contain the information capturing only the local features of the image. In this regard, 1-D HMM has limited capability of representing the 2-D nature of the static images.

The 2-D nature of the character recognition problem leads naturally to a 2-D structure for the HMM. It has been shown, however, that a fully connected 2-D HMM would lead to an NP-complete problem.[8] To avoid this problem, a model with reduced connectivity has been proposed. It is called "pseudo-2-D HMM" or "planar HMM".[9,10] Figure 3 shows the structure of a pseudo-2-D HMM, and its correspondence with a sample data. One of the main shortcomings of the pseudo-2-D HMM is that image lines are assumed to be conditionally independent of each other whereas in an MRF each pixel is dependent on its neighbors of adjacent lines and columns. In addition, the pseudo-2-D HMM has difficulty in estimating model parameters rather than in assigning states to image pixels.[23] Although they are not fully connected 2-D networks, they have been shown to be general enough in characterizing variations of the printed text recognition.

In a more recent work, Gilloux has proposed a handwritten character recognition method based on pseudo-2-D HMMs and Markov meshes.[23] He used the pseudo-2-D HMM to assign states to pixels and then the Markov mesh to estimate the probability of generating the image and the associated states. In the paper, the Markov mesh is defined through the property of dependence between states and super-states of neighboring pixels. While this method has the advantage of taking into account the 2-D nature of characters, the model does not likely to capture the 2-D nature of characters satisfactorily.

In this paper we describe a new framework using a truly 2-D model applicable to off-line handwritten character recognition problems. Since the 2-D model used here is well-suited for representing the spatial continuity of images, it offers a great potential for solving difficult handwritten character recognition problems.

## 3. HIDDEN MARKOV MESH RANDOM FIELD MODEL

In this section, we first give a brief definition of MMRF, and then extend the concept of the MMRF to HMMRF.

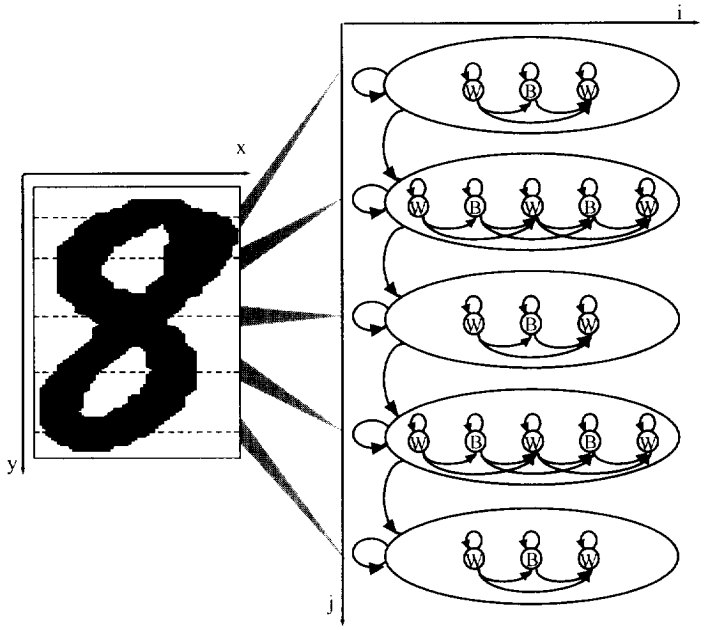



Fig. 3. Structure of a pseudo 2-D HMM.

Let us consider a random field $\mathbf{\Omega} = \{\omega_{m,n}\}$ defined on a finite $M \times N$ integer lattice $\mathbf{L} = \{(m, n): 1 \le m \le M, 1 \le n \le N\}$. Let

$$\Psi_{m,n} = \{(k, l): 1 < k < m \text{ or } 1 < l < n\} \quad \text{for} (m, n) \in \mathbf{L}.$$

Then, a third-order MMRF can be defined as follows.

*Definition* 1. $\mathbf{\Omega}$ is *a third-order MMRF* if and only if

$$P(\omega_{m,n} | \omega_{k,l}, (k, l) \in \Psi_{m,n}) = P(\omega_{m,n} | \omega_{k,l}, (k, l) \in \Lambda_{m,n}) \quad \text{for all} (m, n) \in \mathbf{L}, \tag{1}$$

where $\Lambda_{m,n} = \{(m, n-1), (m-1, n), (m-1, n-1)\}$.

The $\Lambda_{m,n}$'s at the boundary (i.e. for $m = 1$ or $n = 1$) have different configurations from those at the interior sites. At the boundary sites, $P(\omega_{m,n} | \omega_{k,l}, (k, l) \in \Lambda_{m,n})$ is specified as follows:

$$P(\omega_{m,n} | \omega_{k,l}, (k, l) \in \Lambda_{m,n}) = \begin{cases} P(\omega_{m,n}) & \text{if } m = n = 1, \\ P(\omega_{m,n} | \omega_{m,n-1}) & \text{if } n > m = 1, \\ P(\omega_{m,n} | \omega_{m-1,n}) & \text{if } m > n = 1. \end{cases}$$

Figure 4 illustrates the sets of random variables associated with a third-order MMRF.

### 3.1. Elements of a third-order HMMRF model

It is assumed that $\mathbf{\Omega} = \{\omega_{m,n}: (m, n) \in \mathbf{L}\}$ is a third-order MMRF with known transition probability $P(\omega_{m,n} | \omega_{k,l}, (k, l) \in \Lambda_{m,n})$ and initial probability $P(\omega_{1,1})$. In an HMMRF model there is an observation array of random variables, $\mathbf{X}$, which is a probabilistic function of the MMRF $\mathbf{\Omega}$.

Following the notational framework introduced by Rabiner[2] for 1-D HMM, the elements of a third-order HMMRF model can be formally defined as follows:

$S = \{q, r, \ldots, w, y, z\}$: the finite state (label) space, $|S| = \mathcal{S}$.

$V = \{\zeta_i\}$: the discrete set of possible observation symbols, $i = 1, 2, \ldots, I$.

$\mathbf{X} = \mathbf{X}_{1,1}^{M,N} = \{X_{m,n}: (m, n) \in \mathbf{L}\}$: $M \times N$ array of observation symbols $\zeta_i$.

$\mathbf{\Omega} = \mathbf{\Omega}_{1,1}^{M,N} = \{\omega_{m,n}: (m, n) \in \mathbf{L}\}$: $M \times N$ array of pixel states $\omega_{m,n} \in S$.

$A = \{P_{q|r}, P_{q|t}, P_{q|r,s,t}\}$: the 2-D transition probability distribution of states where

$P_{q|r}$: the transition probability distribution of states for the first column,

$$P_{q|r} = P(\omega_{m,1} = q | \omega_{m-1,1} = r) \quad \text{for all } q, r \in S,$$

$P_{q|t}$: the transition probability distribution of states for the first row,

$$P_{q|t} = P(\omega_{1,n} = q | \omega_{1,n-1} = t) \quad \text{for all } q, t \in S,$$

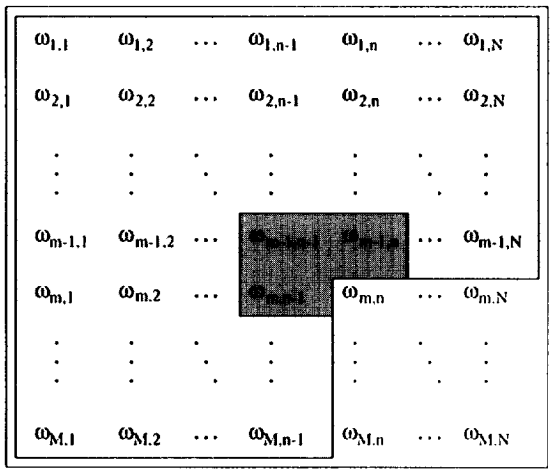



Fig. 4. Sets of random variables associated with a third-order MMRF.

$P_{q|r,s,t}$: the transition probability distribution of states for internal sites except for the first column and the first row,

$$P_{q,r,s,t} = P\left(\omega_{m,n} = q \,\middle|\, \begin{matrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t \end{matrix}\right)$$

for all $q, r, s, t \in S$.

$B = \{p_q(\zeta_i)\}$: the observation symbol probability distribution where

$$p_q(\zeta_i) = P(X_{m,n} = \zeta_i | \omega_{m,n} = q), \quad i = 1, 2, \ldots, I$$

for all $q \in S$.

$\Pi = \{P_q\}$: the initial state probability distribution where

$$P_q = P(\omega_{1,1} = q) \quad \text{for all } q \in S.$$

For convenience, a compact notation of parameter set $\Gamma = (A, B, \Pi)$ will be used.

### 3.2. Problems for HMMRF models

Given the form of HMMRF model of the previous section, two problems have to be addressed when using HMMRFs for character recognition.

- *Parameter estimation problem*—Given the observation array $\mathbf{X}$, how do we adjust the model parameters $\Gamma = (A, B, \Pi)$ to maximize $P(\mathbf{X} | \Gamma)$? This problem is to estimate the parameters of the model that best fits observed training data.
- *Decoding problem*—Given the observation array $\mathbf{X}$, what is the most likely state array $\mathbf{\Omega}$ according to some optimality criterion? This problem can be viewed as devising a real-time and efficient scheme for computing maximum *a posteriori* (MAP) estimate of states.

These are similar to the problems for HMMs. We will first deal with the decoding problem and return to the parameter estimation problem.

## 4. EFFICIENT DECODING ALGORITHM

Image decoding is a general framework to solve low-level vision tasks, such as image classification, edge detection, etc. The decoding problem will be to decide the pixel states or labels given the information comprised in the observations. Each pixel $(m, n)$ is associated with an unknown state $\omega_{m,n}$ and a known observation (feature/feature vector) $X_{m,n}$ containing imperfect information about the corresponding state.

### 4.1. Problem formulation

As defined in previous section, $\mathbf{X}$ denotes an observation array, and $\mathbf{\Omega}$ denotes a state array that could have caused $\mathbf{X}$. The general task of the decoding problem is to find a state array $\hat{\mathbf{\Omega}}$ that would maximize the posterior distribution $P(\mathbf{\Omega} \mid \mathbf{X}, \Gamma)$. But since

$$P(\mathbf{\Omega} \mid \mathbf{X}, \Gamma) = \frac{P(\mathbf{X}, \mathbf{\Omega} \mid \Gamma)}{P(\mathbf{X})} \tag{2}$$

and $P(\mathbf{X})$ does not depend on $\mathbf{\Omega}$, we might as well find $\hat{\mathbf{\Omega}}$ that maximizes $P(\mathbf{X}, \Omega \mid \Gamma)$ as

$$\begin{aligned} P(\mathbf{X}, \mathbf{\Omega} \mid \Gamma) &= P(\mathbf{X} \mid \mathbf{\Omega}, \Gamma)\, P(\mathbf{\Omega}, \Gamma) \\ &= \prod_{m=1}^{M} \prod_{n=1}^{N} P(\omega_{m,n} \mid \omega_{k,l}, (k,l) \in \Lambda_{m,n}) \\ &\quad \times p_{\omega_{m,n}}(X_{m,n}) \\ &= P(\omega_{1,1})\, p_{\omega_{1,1}}(X_{1,1}) \prod_{n=2}^{N} P_{\omega_{1,n} \mid \omega_{1,n-1}} \\ &\quad \times P_{\omega_{2,n}}(X_{1,n}) \prod_{m=2}^{M} P_{\omega_{m,1} \mid \omega_{m-1,1}} p_{\omega_{m,1}}(X_{m,1}) \\ &\quad \times \prod_{m=2}^{M} \prod_{n=2}^{N} P_{\omega_{m,n} \mid \omega_{m-1,n}, \omega_{m,n-1}} \\ &\quad \times p_{\omega_{m,n}}(X_{m,n}). \end{aligned} \tag{3}$$

Let $\mathbf{\Omega}'$ denote the set of all possible state arrays $\mathbf{\Omega}$ in $\Gamma$. Then the state array $\hat{\mathbf{\Omega}}$ that best explains the observation array $\mathbf{X}$ can be estimated as

$$\hat{\mathbf{\Omega}} \propto \arg\max_{\mathbf{\Omega} \in \mathbf{\Omega}'} P(\mathbf{X}, \mathbf{\Omega} \mid \Gamma), \tag{4}$$

and the observation likelihood $P(\mathbf{X} \mid \Gamma)$ is approximated as

$$\hat{P}(\mathbf{X} \mid \Gamma) \propto P(\mathbf{X}, \hat{\mathbf{\Omega}} \mid \Gamma) \tag{5}$$

which represents the probability of the best configuration or the maximum likelihood state array.

There are several possible optimality criteria for solving equations (4) and (5). The obvious choices are (i) the state array that has overall maximum probability given the observation array, and (ii) that in which the state at each individual pixel has maximum probability given the observation array.[24] In the Bayesian framework, (i) corresponds to MAP estimation, whereas (ii) maximizes the marginal posterior probability at each pixel. We will find the optimal state array according to the criterion (ii).

Smoothing algorithm[25] is based on the computation of $P(\omega_{m,n} \mid \mathbf{X}_{1,1}^{M,N})$, the *a posteriori* distribution of individual signal values, given the entire observation $\mathbf{X}_{1,1}^{M,N}$. However, because of overwhelming computational difficulties in implementing the smoothing algorithm on an image matrix, we make certain approximations such as processing the image in narrow strips[25] or look-ahead data.[16]

The objective in this section is to compute $P(\omega_{m,n} \mid \mathbf{X}_{1,1}^{m-k,n+k})$, the *a posteriori* distribution of the state $\omega_{m,n}$ at pixel $(m, n)$ given the look-ahead data $\mathbf{X}_{1,1}^{m+k,n-k}$, for each pixel in the lattice. For the model under consideration the *a posteriori* distribution $P(\omega_{m,n} \mid \mathbf{X}_{1,1}^{m+k,n-k})$ can be calculated recursively using the so-called "*fixed-lag*" smoothing approach.

### 4.2. An efficient decoding algorithm using look-ahead technique

For the fixed-lag smoothing problem, we use the decoding algorithm based on the "*look-ahead*" technique[16] which enable an efficient estimation of MMRF states in real-time. The fixed-lag smoothing algorithm described in this paper takes the estimate of $\omega_{m,n}$ as the mode of the *a posteriori* distribution $P(\omega_{m,n} \mid \mathbf{X}_{1,1}^{m+k,n+k})$ for $k = 1$. In particular, it is called the one-row one-column look-ahead technique, as the state of pixel $(m, n)$ is not fixed until $(m+1, n+1)$ has been reached. The look-ahead decoding algorithm recursively computes the *a posteriori* distribution of the individual pixel values given the observed look-ahead image, under an additional assumption that columns of the scene constitute a vector Markov chain.

The classification of pixel $(m, n)$ will be

$$\hat{\omega}_{m,n} = \arg\max_{q \in S} P(\omega_{m,n} = q \mid \mathbf{X}_{1,1}^{m+1,n+1}) \tag{6}$$

for all $1 \le m \le M-1$ and $1 \le n \le N-1$.

For the purpose of this work, let us employ the following notations as used in Devijver's work:[16]

$$H_{m,n}(q, r, s, t) \doteq P\left(\begin{matrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q \end{matrix} \,\middle|\, \mathbf{X}_{1,1}^{m,n}\right),$$
$$2 \le m \le M \text{ and } 2 \le n \le N$$

$$G_{m,n}(r, s, t) \doteq P\left(\begin{matrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \end{matrix} \,\middle|\, \mathbf{X}_{1,1}^{m,n} \setminus \{X_{m,n}\}\right),$$
$$2 \le m \le M \text{ and } 2 \le n \le N$$

$$Y_{m,n}(q, t) \doteq P(\omega_{m,n-1} = t \;\; \omega_{m,n} = q \mid \mathbf{X}_{1,1}^{m,n}),$$
$$1 \le m \le M \text{ and } 2 \le n \le N$$

$$Z_{m,n}(q, r) \doteq P\begin{pmatrix}\omega_{m-1,n} = r \\ \omega_{m,n} = q\end{pmatrix} | \mathbf{X}_{1,1}^{m,n}\Bigg)$$

$$2 \le m \le M \text{ and } 1 \le n \le N$$

$$F_{m,n}(q) \doteq P(\omega_{m,n} = q \,|\, \mathbf{X}_{1,1}^{m,n}),$$

$$1 \le m \le M \text{ and } 1 \le n \le N$$

$$Q_{m-1,n-1}(s) \doteq P(\omega_{m-1,n-1} = s \,|\, \mathbf{X}_{1,1}^{m,n}),$$

$$2 \le m \le M \text{ and } 2 \le n \le N.$$

The following algorithm specifies the complete look-ahead decoding including the boundary conditions. This algorithm summarizes how the *a posteriori* probabilities shown in Fig. 5 are used in order to compute $Q_{m-1,n-1}(s) = P(\omega_{m-1,n-1} \,|\, \mathbf{X}_{1,1}^{m,n})$ for all $s$'s and for all $(m, n)$ of the **L**.

LOOK-AHEAD DECODING ALGORITHM

*Step* 1: *Initialization*

$$F_{1,1}(q) = \frac{P_q\, p_q(X_{1,1})}{\sum_{q'} P_{q'}\, p_{q'}(X_{1,1})}, \quad \forall q.$$

*Step* 2: *Recursion*

*Step* 2.1. *Recursion for the first row*:

For $n = 2, 3, \ldots, N$

$$Y_{1,n}(q, t) = \frac{F_{1,n-1}(t)\, P_{q|t}\, p_q(X_{1,n})}{\sum_{q',t'} F_{1,n-1}(t')\, P_{q'|t'}\, p_{q'}(X_{1,n})}, \quad \forall q, t$$

$$F_{1,n}(q) = \sum_t Y_{1,n}(q, t), \quad \forall q.$$

*Step* 2.2. *Recursion for the first column*:

For $m = 2, 3, \ldots, M$

$$Z_{m,1}(q, r) = \frac{F_{m-1,1}(r)\, P_{q|r}\, p_q(X_{m,1})}{\sum_{q',r'} F_{m-1,1}(r')\, P_{q'|r'}\, p_{q'}(X_{m,1})}, \quad \forall q, r$$

$$F_{m,1}(q) = \sum_r Z_{m,1}(q, r), \quad \forall q.$$

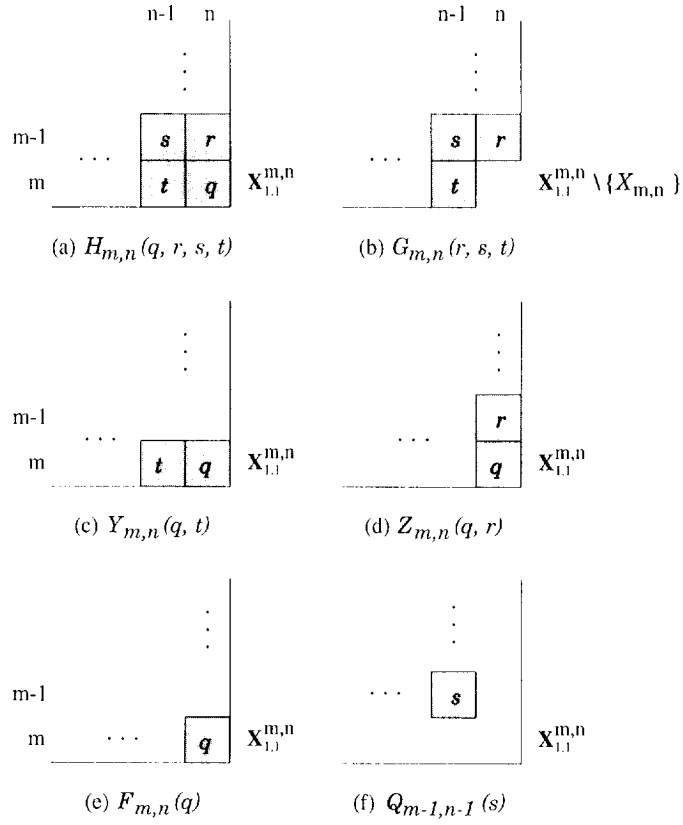
*Step* 2.3. Recursion for interior sites except for the first row and the first column:

For $m = 2, 3, \ldots, M$ and $n = 2, 3, \ldots, N$

$$G_{m,n}(r, s, t) = \frac{Y_{m-1,n}(r, s)\, Z_{m,n-1}(t, s)}{F_{m-1,n-1}(s)}, \quad \forall r, s, t$$

$$H_{m,n}(q, r, s, t) = G_{m,n}(r, s, t)\, P_{q|r,s,t}\, p_q(X_{m,n}), \quad \forall q, r, s, t$$

$$Y_{m,n}(q, t) = \sum_{r,s} H_{m,n}(q, r, s, t), \quad \forall q, t$$




(a) $H_{m,n}(q, r, s, t)$ (b) $G_{m,n}(r, s, t)$

(c) $Y_{m,n}(q, t)$ (d) $Z_{m,n}(q, r)$

(e) $F_{m,n}(q)$ (f) $Q_{m-1,n-1}(s)$

Fig. 5. Illustration for $H$, $G$, $Y$, $Z$, $F$, and $Q$.

$$Z_{m,n}(q, r) = \sum_{s,t} H_{m,n}(q, r, s, t), \quad \forall q, r$$

$$F_{m,n}(q) = \sum_{t} Y_{m,n}(q, t), \quad \forall q$$

$$Q_{m-1,n-1}(s) = \sum_{q,r,t} H_{m,n}(q, r, s, t), \quad \forall s$$

[Classification]

$$\hat{\omega}_{m-1,n-1} = \arg\max_{s} Q_{m-1,n-1}(s).$$

*Step* 3: *Extra Recursion*

*Step* 3.1. Recursion for the first site of the last row:

$$Q_{M,1}(s) = \sum_{r} Y_{M,2}(s, r), \quad \forall s$$

[Classification]

$$\hat{\omega}_{M,1} = \arg\max_{s} Q_{M,1}(s).$$

*Step* 3.2. Recursion for the rest of the last row:

For $n = 2, 3, \ldots, N$

$$Q_{M,n}(s) = \sum_{r} Y_{M,n}(r, s), \quad \forall s$$

[Classification]

$$\hat{\omega}_{M,n} = \arg\max_{s} Q_{M,n}(s).$$

*Remark* 1. Since values of the first column, the first row, and the last row are not available on a third-order MMRF, sites at the boundary are processed by a first-order Markov chain.

*Remark* 2. A normalization operation is needed in order to avoid underflow when simulated on a computer.

Figure 6 illustrates the computational process for $F$ and $Q$ over the lattice **L**. The reader is referred to ref. (16) for further details.

### 4.3. Recognition

Each class of patterns has a single HMMRF model. Once all the HMMRF models are trained, the recognition is straightforward. Assume that we have $V$ models denoted by $\Gamma_v$, $v = 1, 2, \ldots, V$. Given an

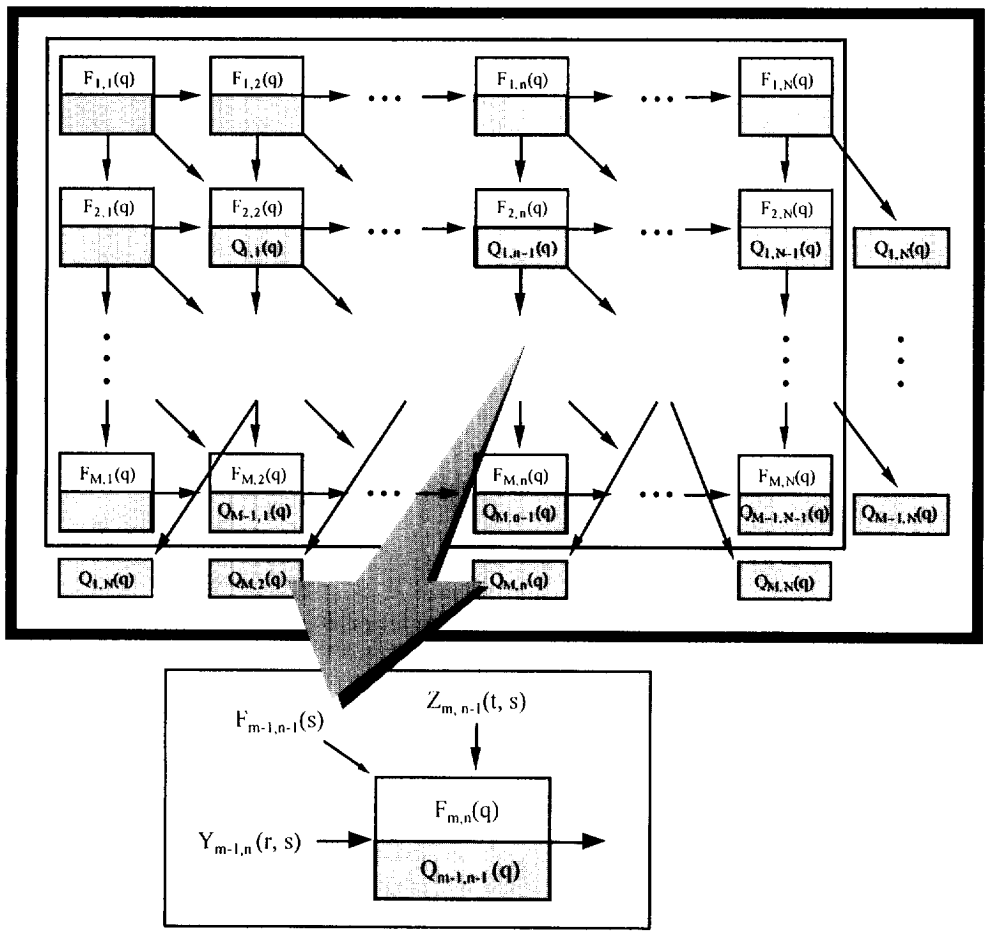



Fig. 6. Implementation diagram of decoding process.

observation array $\mathbf{X}$, we first calculate $P(\mathbf{X}, \Omega | \Gamma_v)$, $v = 1, 2, \ldots, V$. Here, $\Omega$ stands for the optimal state array given by the look-ahead decoding algorithm. We then select the character whose state-optimized likelihood is highest, i.e.

$$v^* = \arg \max_{1 \le v \le V} P(\mathbf{X}, \hat{\Omega} | \Gamma_v). \tag{7}$$

## 5. ESTIMATION OF MODEL PARAMETERS

To use the HMMRF model of Section 3, it is necessary to determine the values of its statistical parameters. Model parameter adjustment to fit a set of observed training data is often referred to as *training*. This section discusses training problems that arise in using our image model, and proposes a novel solution to this problem.

As mentioned earlier, the use of 2-D MMRF in image segmentation has become common. However, it has not been applied to character recognition problem at all except for Gilloux's suggestion[23] and the attempt by the authors of this paper.[17] The main reason is that efficient training algorithms do not exist. A fundamental difference between image segmentation and character recognition lies in parameter estimation. In the former, the parameter set of a model is estimated using a realization from the image model, while in the latter it is estimated using a set of known training data. One may attempt to extend the efficient training algorithm of HMM to 2-D, but causality in 2-D model does not guarantee the existence of exact recurrence relationships.

In this section, we will first review briefly the training algorithm of HMMs[2] so that we can make our presentation clearer and the comparisons with the proposed estimation algorithm easier. We will introduce two approximate solutions: the decision-directed (DD) method[26] and the proposed method.

### 5.1. The classical hidden Markov model

The training problem of HMMs is to determine a method to adjust the model parameters $(A, B, \pi)$ to maximize the probability of the observation sequence given the model. There is no known way to analytically solve for the model which maximizes the probability of the observation sequence.[2] However, the Baum Welch algorithm[18] iteratively provides parameter estimates that maximize the actual MLE criterion, i.e. $P(\mathbf{O}|\lambda)$; those estimates are based on partial path probabilities computed by forward and backward recurrences. Convergence at least to a local maximum is guaranteed.

### 5.2. A decision-directed estimation algorithm

Although the problem of unsupervised learning can be stated as merely the problem of estimating parameters of a mixture density, neither the maximum likelihood nor the Bayesian approach yields analytically simple results.[26] Exact solutions for even the simplest nontrivial examples lead to computational requirements that grow exponentially with the number of samples. However, the problem of unsupervised learning is too important to abandon just because exact solutions are hard to find, and numerous procedures for obtaining approximate solutions have been suggested. An obvious approach to unsupervised learning is the *decision-directed* method. It permits a considerable reduction in computation and storage.

Let the look-ahead decoding algorithm be carried out on the observation array $\mathbf{X}_{1,1}^{m+1,n+1}$ and result in the estimated most probable state array $\hat{\mathbf{\Omega}} = \{\hat{\omega}_{1,1}, \hat{\omega}_{1,2}, \ldots, \hat{\omega}_{M,N}\}$. Let $N(S, x)$ denote the number of times the state $S$ is visited in the state array $\hat{\mathbf{\Omega}}$ and corresponds to the output $x$, and let $N(\{S\}, S')$ denote the number of times the transitions $\{S\} \to S'$ take place in the decoded state array $\hat{\mathbf{\Omega}}$ where $S$ and $S'$ are pixel state labels. For example, $N(\{r, s, t\}, q)$ denotes the number of times the transition $\{r, s, t\} \to q$ takes place in $\hat{\mathbf{\Omega}}$.

When using the DD re-estimation method, the respective estimators are:

$$\hat{p}_q(x) = \frac{N(q, x)}{\sum_x N(q, x)},$$

$$\hat{P}_{q|r} = \frac{N(\{r\}, q)}{\sum_q N(\{r\}, q)}, \quad \hat{P}_{q|t} = \frac{N(\{t\}, q)}{\sum_q N(\{t\}, q)}$$

$$\hat{P}_{q\,r,s,t} = \frac{N(\{r, s, t\}, q)}{\sum_q N(\{r, s, t\}, q)}, \tag{8}$$

If the look-ahead estimated state array $\hat{\mathbf{\Omega}} = \{\hat{\omega}_{1,1}, \hat{\omega}_{1,2}, \ldots, \hat{\omega}_{M,N}\}$ was the one actully realized by the source when it generated the data $\{X_{1,1}, X_{1,2}, \ldots, X_{M,N}\}$, then formula (8) would give the actual maximum likelihood estimates of the probabilities $p$ and $P$ underlying the source. The process is iterated until no substantial change in the estimated values takes place. The success of the DD re-estimation method thus depends on both the closeness of the initial guesses $\hat{p}^0$ and $\hat{P}^0$ to the actual $p$ and $P$ values.

### 5.3. New estimation algorithm

The formulation of the proposed estimation can be derived from the extension of the look-ahead technique which is based on a maximum, marginal *a posteriori* probability criterion for a third-order HMMRF model. In order to overcome the computational problems that have precluded the use of a truly 2-D model, we shall need simplifying assumptions.

Each training data is associated with a specific HMMRF model $\Gamma$ of parameters $(A, B, \Pi)$. We assume that the probability of pixel $(m, n)$ being in state $q$ given all the observations $\mathbf{X}_{1,1}^{M,N}$ is fairly well estimated given only the look-ahead observations $\mathbf{X}_{1,1}^{m+1,n+1}$. Estimates of the model parameters are

given by

$$\hat{P}_q \approx \frac{P(\omega_{1,1} = q \mid \mathbf{X}_{1,1}^{2,2})}{\sum_{q'} P(\omega_{1,1} = q' \mid \mathbf{X}_{1,1}^{2,2})}, \tag{9}$$

$$\hat{P}_{q|r} \approx \frac{\sum_{m=2}^{M-1} P\left(\begin{matrix}\omega_{m-1,1} = r \\ \omega_{m,1} = q\end{matrix}\middle| \mathbf{X}_{1,1}^{m+1,2}\right)}{\sum_{m=2}^{M-1}\sum_{q'} P\left(\begin{matrix}\omega_{m-1,1} = r \\ \omega_{m,1} = q'\end{matrix}\middle| \mathbf{X}_{1,1}^{m+1,2}\right)}, \quad \forall r \tag{10}$$

$$\hat{P}_{q\,t} \approx \frac{\sum_{n=2}^{N-1} P(\omega_{1,n-1} = t, \omega_{1,n} = q \mid \mathbf{X}_{1,1}^{2,n+1})}{\sum_{n=2}^{N-1}\sum_{q'} P(\omega_{1,n-1} = t, \omega_{1,n} = q' \mid \mathbf{X}_{1,1}^{2,n+1})}, \quad \forall t \tag{11}$$

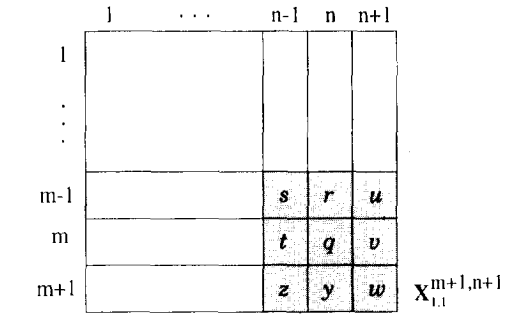



Fig. 7. Labeling configuration of $\mathbf{X}_{1,1}^{m+1,n+1}$.

$$\hat{P}_{q|r,s,t} \approx \frac{\sum_{m=2}^{M-1}\sum_{n=2}^{N-1} P\left(\begin{matrix}\omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q\end{matrix}\middle| \mathbf{X}_{1,1}^{m+1,n+1}\right)}{\sum_{m=2}^{M-1}\sum_{n=2}^{N-1}\sum_{q'} P\left(\begin{matrix}\omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q'\end{matrix}\middle| \mathbf{X}_{1,1}^{m+1,n+1}\right)}, \quad \forall r, s, t \tag{12}$$

$$\hat{p}_q(\zeta_i) \approx \frac{\sum_{m,n \mid X_{m,n} = \zeta_i} P(\omega_{m,n} = q \mid \mathbf{X}_{1,1}^{m+1,n+1})}{\sum_{m,n} P(\omega_{m,n} = q \mid \mathbf{X}_{1,1}^{m+1,n+1})}, \quad i = 1, 2, \ldots, I \tag{13}$$

for all $q \in S$.

Now an iterative re-estimation procedure to compute equations (9)–(13) can be established. Let us first consider the re-estimation of 2-D state transition parameter $P_{q|r,s,t}$. Referring to Fig. 7, the posterior probability

$$P\left(\begin{matrix}\omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q\end{matrix}\middle| \mathbf{X}_{1,1}^{m+1,n+1}\right)$$

requires summing the configurations over the state variables $u, v, z, y$ and $w$ as

$$P\left(\begin{matrix}\omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q\end{matrix}\middle| \mathbf{X}_{1,1}^{m+1,n+1}\right)$$

$$= \sum_{u,v,z,y,w \in S} P\left(\begin{matrix}\omega_{m-1,n-1} = s & \omega_{m-1,n} = r & \omega_{m-1,n+1} = u & \omega_{m,n-1} = t \\ & \omega_{m,n} = q & \omega_{m,n+1} = v & \omega_{m+1,n-1} = z \\ & \omega_{m+1,n} = y & \omega_{m+1,n+1} = w & \end{matrix}\middle| \mathbf{X}_{1,1}^{m+1,n+1}\right)$$

$$\propto \sum_{u,r,z,y \in S} P\left(\begin{matrix}\omega_{m-1,n-1} = s & \omega_{m-1,n} = r & \omega_{m-1,n+1} = u \\ \omega_{m,n-1} = t & \omega_{m,n} = q & \omega_{m,n+1} = v \\ \omega_{m+1,n-1} = z & \omega_{m+1,n} = y & \end{matrix}\middle| \mathbf{X}_{1,1}^{m+1,n+1} \setminus \{X_{m+1,n+1}\}\right)$$

$$\times \sum_{w \in S} P_{w|u,r,s,t,z,v,q,y}\, p_w(X_{m+1,n+1}). \tag{14}$$

By definition of the a third-order MMRF, we can verify the following corollary easily:

*Corollary.*

$$P_{w|u,r,s,t,z,v,q,y} = P_{w|v,q,y} \quad \text{where } u, r, s, t, z, v, q, y, w \in S. \tag{15}$$

We can now address the problem of computing the first factor in the right-hand side of equation (14). Yet a straightforward calculation is not practical. To obtain an implementable estimation algorithm, the following two assumptions are imposed. Prior to making the problem easier with assumptions, consider the following notations. For $m > 1$ and $n > 1$, let us define $\mathbf{\Omega}_{m,n}^{C} = \{\omega_{k,n}: \ 1 \le k \le m\}$ and $\mathbf{\Omega}_{m,n}^{R} = \{\omega_{m,l}: \ 1 \le l \le n\}$ and similarly for $\mathbf{X}$ and $\mathbf{L}$.

*Assumption* 1. Given $\mathbf{\Omega}_{1,1}^{m,n}$, the column $\mathbf{\Omega}_{m,n+1}^{C}$ and row $\mathbf{\Omega}_{m+1,n}^{R}$ are conditionally independent.

*Assumption* 2. $\mathbf{\Omega}_{m,n+1}^{C}$ and $\mathbf{\Omega}_{m+1,n}^{R}$ are conditionally independent provided that only the left column $\mathbf{\Omega}_{m,n}^{C}$ and upper row $\mathbf{\Omega}_{m,n}^{R}$ are given.

We can derive the following theorem, which is the extension of derivation of $G_{m,n}$ in Section 4.

*Theorem* 1. Let

$$U_{m+1,n+1}(u, r, s, t, z, v, q, y)$$

$$\doteq P\left(\begin{matrix}\omega_{m-1,n-1} = s & \omega_{m-1,n} = r & \omega_{m-1,n+1} = u \\ \omega_{m,n-1} = t & \omega_{m,n} = q & \omega_{m,n+1} = v \\ \omega_{m+1,n-1} = z & \omega_{m+1,n} = y & \end{matrix}\middle| \mathbf{X}_{1,1}^{m+1,n+1} \setminus \{X_{m+1,n+1}\}\right).$$

Then under Assumptions 1 and 2, $U_{m+1,n+1}(u, r, s, t, z, v, q, y)$ *is given by*

$$U_{m+1,n+1}(u, r, s, t, z, v, q, y) \propto \begin{cases} 0 & \text{if } Z_{m,n}(q, r) = 0 \text{ or } Y_{m,n}(q, t) = 0, \\ \dfrac{H_{m,n}(q, r, s, t)\, H_{m,n+1}(v, u, r, q)\, H_{m+1,n}(y, q, t, z)}{Z_{m,n}(q, r)\, Y_{m,n}(q, t)} & \text{otherwise.} \end{cases} \tag{16}$$

*Proof.* We decompose $\mathbf{L}_{1,1}^{m+1,n+1} \backslash (m+1, n+1)$ into three parts:

$$\mathbf{L}_{1,1}^{m+1,n+1} \backslash (m+1, n+1) = \mathbf{L}_{1,1}^{m,n} \cup \mathbf{L}_{m,n+1}^{C} \cup \mathbf{L}_{m+1,n}^{R}$$

where $\mathbf{L}_{m,n+1}^{C}$ and $\mathbf{L}_{m+1,n}^{R}$ are shown in Fig. 8.

We proceed as follows:

$$U_{m+1,n+1}(u, r, s, t, z, v, q, y)$$

$$\propto P\begin{pmatrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r & \omega_{m-1,n+1} = u \\ \omega_{m,n-1} = t & \omega_{m,n} = q & \omega_{m,n-1} = v \cup \mathbf{X}_{1,1}^{m,n} \cup \mathbf{X}_{m,n+1}^{C} \cup \mathbf{X}_{m+1,n}^{R} \\ \omega_{m+1,n-1} = z & \omega_{m+1,n} = y & \end{pmatrix}$$

$$= P\begin{pmatrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q \end{pmatrix}, \mathbf{X}_{1,1}^{m,n}\Bigg)$$

$$\times P\left(\begin{pmatrix} \omega_{m-1,n+1} = u \\ \omega_{m,n+1} = v \end{pmatrix}, \mathbf{X}_{m,n+1}^{C} \,\middle|\, \begin{pmatrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q \end{pmatrix}, \mathbf{X}_{1,1}^{m,n}\right).$$

$$\times P\left(\omega_{m+1,n-1} = z,\ \omega_{m+1,n} = y, \mathbf{X}_{m+1,n}^{R} \,\middle|\, \begin{pmatrix} \begin{pmatrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q \end{pmatrix}, \mathbf{X}_{1,1}^{m,n} \end{pmatrix} \right). \tag{17}$$

$$\begin{pmatrix} \omega_{m-1,n+1} = u \\ \omega_{m,n+1} = v \end{pmatrix}, \mathbf{X}_{m,n+1}^{C}$$

From Assumptions 1 and 2, we have

(i) Given $\begin{pmatrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q \end{pmatrix}, \mathbf{X}_{1,1}^{m,n}\Bigg)$,

$$\begin{pmatrix} \omega_{m-1,n+1} = u \\ \omega_{m,n+1} = v \end{pmatrix}, \mathbf{X}_{m,n+1}^{C}\Bigg) \text{ and}$$

$(\omega_{m+1,n-1} = z, \omega_{m+1,n} = y, \mathbf{X}_{m+1,n}^{R})$ are conditionally independent.

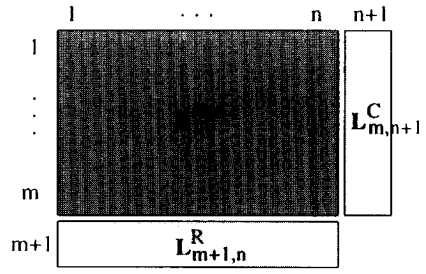



Fig. 8. Decomposition of $\mathbf{L}_{1,1}^{m+1,n+1} \backslash (m+1, n+1)$.

(ii) Given $\begin{pmatrix} \omega_{m-1,n} = r \\ \omega_{m,n} = q \end{pmatrix}, \mathbf{X}_{1,1}^{m,n}\Bigg)$,

$$\begin{pmatrix} \omega_{m-1,n+1} = u \\ \omega_{m,n+1} = v \end{pmatrix}, \mathbf{X}_{m,n+1}^{C}\Bigg)$$

is independent from

$$\begin{pmatrix} \omega_{m-1,n-1} = s \\ \omega_{m,n-1} = t \end{pmatrix}\Bigg).$$

Similarly, given $(\omega_{m,n-1} = t,\ \omega_{m,n} = q,\ \mathbf{X}_{1,1}^{m,n})$, $(\omega_{m+1,n-1} = z,\ \omega_{m+1,n} = y,\ \mathbf{X}_{m+1,n}^{R})$ is independent from $(\omega_{m-1,n-1} = s,\ \omega_{m-1,n} = r)$.

Then, it follows that

$$U_{m+1,n+1}(u, r, s, t, z, v, q, y)$$

$$\propto P\begin{pmatrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q \end{pmatrix}, \mathbf{X}_{1,1}^{m,n}\Bigg)$$

$$\times P\left(\begin{pmatrix} \omega_{m-1,n+1} = u \\ \omega_{m,n+1} = v \end{pmatrix}, \mathbf{X}_{m,n+1}^{C} \,\middle|\, \begin{pmatrix} \omega_{m-1,n} = r \\ \omega_{m,n} = q \end{pmatrix}, \mathbf{X}_{1,1}^{m,n}\right).$$

$$\times P(\omega_{m+1,n-1} = z, \omega_{m+1,n} = y,$$

$$\mathbf{X}_{m+1,n}^{R} \,|\, \omega_{m,n-1} = t, \omega_{m,n} = q, \mathbf{X}_{1,1}^{m,n}) \tag{18}$$

We would like to rewrite equation (18) in the form of the conditional probability given the observations. Hence, we get

$$U_{m+1,n+1}(u, r, s, t, z, v, q, y)$$

$$\propto \frac{1}{P\left(\begin{pmatrix} \omega_{m-1,n} = r \\ \omega_{m,n} = q \end{pmatrix} \middle| \mathbf{X}_{1,1}^{m,n}\right) P(\omega_{m,n-1} = t, \omega_{m,n} = q \,|\mathbf{X}_{1,1}^{m,n})}$$

$$\times P\left(\begin{pmatrix} \omega_{m-1,n-1} = s & \omega_{m-1,n} = r \\ \omega_{m,n-1} = t & \omega_{m,n} = q \end{pmatrix} \middle| \mathbf{X}_{1,1}^{m,n}\right)$$

$$\times P\begin{pmatrix} \omega_{m-1,n}=r & \omega_{m-1,n+1}=u \\ \omega_{m,n}=q & \omega_{m,n+1}=v \end{pmatrix} \mathbf{X}_{1,1}^{m,n+1}$$

$$\times P\begin{pmatrix} \omega_{m,n-1}=t & \omega_{m,n}=q \\ \omega_{m+1,n-1}=z & \omega_{m+1,n}=y \end{pmatrix} \mathbf{X}_{1,1}^{m+1,n}. \quad (19)$$

This probability is zero if any one of the factors in the denominator is equal to zero. By replacing $H$, $Z$ and $Y$, we can get Theorem 1. □

Applying Corollary and Theorem 1 to equation (14) yields

$$P\begin{pmatrix} \omega_{m-1,n-1}=s & \omega_{m-1,n}=r \\ \omega_{m,n-1}=t & \omega_{m,n}=q \end{pmatrix} \mathbf{X}_{1,1}^{m+1,n+1} \propto \sum_{u,r,z,y\in S} \frac{H_{m,n}(q,r,s,t)\, H_{m,n+1}(v,u,r,q)\, H_{m+1,n}(y,q,t,z)}{Z_{m,n}(q,r)\, Y_{m,n}(q,t)}$$

$$\times \sum_{w\in S} P_{w|v,q,y}\, p_w(X_{m+1,n+1}). \quad (20)$$

We proceed in a similar way for the estimates (10) and (11). Consider first the transition probabilities for the Markov chains governing the distributions of states along the first column. The result of the derivation for equation (10) is as follows:

$$P\begin{pmatrix} \omega_{m-1,1}=r \\ \omega_{m,1}=q \end{pmatrix} \mathbf{X}_{1,1}^{m+1,2} = \sum_{u,r,y,w\in S} \times P\begin{pmatrix} \omega_{m-1,1}=r & \omega_{m-1,2}=u \\ \omega_{m,1}=q & \omega_{m,2}=v \\ \omega_{m+1,1}=y & \omega_{m+1,2}=w \end{pmatrix} \mathbf{X}_{1,1}^{m+1,2}$$

$$\propto \sum_{u,r,y\in S} P\left(\begin{pmatrix} \omega_{m-1,1}=r & \omega_{m-1,2}=u \\ \omega_{m,1}=q & \omega_{m,2}=v \\ \omega_{m+1,1}=y \end{pmatrix} \mathbf{X}_{1,1}^{m+1,2}\setminus \{X_{m+1,2}\}\right)$$

$$\times \sum_{w\in S} P_{w|u,r,v,q,y}\, p_w(X_{m+1,2}), \quad (21)$$

Using the properties of MMRF and the conditional properties we have the following theorem.

*Theorem* 2: Let

$$U_{m+1,2}^{C}(u,r,v,q,y) \doteq P\left(\begin{pmatrix} \omega_{m-1,1}=r & \omega_{m-1,2}=u \\ \omega_{m,1}=q & \omega_{m,2}=v \\ \omega_{m+1,1}=y \end{pmatrix} \mathbf{X}_{1,1}^{m+1,2}\setminus\{X_{m+1,2}\}\right).$$

Then under Assumptions 1 and 2, $U_{m+1,2}^{C}(u,r,v,q,y)$ is given by

$$U_{m+1,2}^{C}(u,r,v,q,y) \propto \begin{cases} 0, & \text{if } F_{m,1}(q)=0, \\ \dfrac{H_{m,2}(v,u,r,q)\, Z_{m+1,1}(y,q)}{F_{m,1}(q)}, & \text{otherwise.} \end{cases} \quad (22)$$

*Proof.* See the appendix.

Hence, we get

$$P\begin{pmatrix} \omega_{m-1,1}=r \\ \omega_{m,1}=q \end{pmatrix} \mathbf{X}_{1,1}^{m+1,2} \propto \sum_{u,r,y\in S} \frac{H_{m,2}(v,u,r,q)\, Z_{m+1,1}(y,q)}{F_{m,1}(q)} \times \sum_{w\in S} P_{w|r,q,y} p_w(X_{m+1,2}). \quad (23)$$

Finally, to re-estimate the transition probability distribution of states along the first row, we should compute the posterior probability $P(\omega_{1,n-1}=t, \omega_{1,n}=q \mid \mathbf{X}_{1,1}^{2,n+1})$. This probability can be developed as follows:

$$P(\omega_{1,n-1}=t, \omega_{1,n}=q \mid \mathbf{X}_{1,1}^{2,n+1}) = \sum_{r,z,y,w\in S} P\left(\begin{pmatrix} \omega_{1,n-1}=t & \omega_{1,n}=q & \omega_{1,n+1}=v \\ \omega_{2,n-1}=z & \omega_{2,n}=y & \omega_{2,n+1}=w \end{pmatrix} \mathbf{X}_{1,1}^{2,n+1}\right)$$

$$\propto \sum_{r,z,y\in S} P\left(\begin{pmatrix} \omega_{1,n-1}=t & \omega_{1,n}=q & \omega_{1,n+1}=v \\ \omega_{2,n-1}=z & \omega_{2,n}=y \end{pmatrix} \mathbf{X}_{1,1}^{2,n+1}\setminus n\{X_{2,n+1}\}\right) \times \sum_{w\in S} P_{w|t,z,v,q,y}\, p_w(X_{2,n+1}) \quad (24)$$

And the following is the associated theorem for the first factor on the right-hand side of equation (24).

*Theorem* 3. Let

$$U_{2,n+1}^{R}(t,z,v,q,y) \doteq P\left(\begin{pmatrix} \omega_{1,n-1}=t & \omega_{1,n}=q & \omega_{1,n+1}=v \\ \omega_{2,n-1}=z & \omega_{2,n}=y \end{pmatrix} \mathbf{X}_{1,1}^{2,n+1}\setminus\{X_{2,n+1}\}\right).$$

Then under Assumptions 1 and 2, $U_{2,n+1}^{R}(t,z,v,q,y)$ is given by

$$U_{2,n+1}^{R}(t,z,v,q,y) \propto \begin{cases} 0 & \text{if } F_{1,n}(q)=0, \\ \dfrac{Y_{1,n+1}(v,q)\, H_{2,n}(y,q,t,z)}{F_{1,n}(q)} & \text{otherwise.} \end{cases} \quad (25)$$

The proof is straightforward and omitted.

Thus, we have

$$P(\omega_{1,n-1} = t, \omega_{1,n} = q \mid \mathbf{X}_{1,1}^{2,n+1}) \propto \sum_{r,z,y \in S} \frac{Y_{1,n+1}(r, q) H_{2,n}(y, q, t, z)}{F_{1,n}(q)} \times \sum_{w \in S} P_{w|r,q,y} p_w(X_{2,n+1}). \quad (26)$$

*Remark* 3. Note that we can easily obtain the estimates of $P_q$ and $p_q(\zeta_i)$ because $P(\omega_{m,n} = q \mid \mathbf{X}_{1,1}^{m+1,n+1}) = Q_{m,n}(q)$.

With these results, we can estimate all model parameters $A$, $B$, and $\Pi$.

## 6. EXPERIMENTAL RESULTS

Unlike the case of synthetic images, there is considerable uncertainty in handwritten character images. The purpose of this section is to show how the HMMRF-based approach can be applied to the problems of off-line handwritten character recognition.

### 6.1. Database

Off-line character recognition involves the use of a fixed number of models, each of which corresponds to a class of target objects. In order to verify the performance of the proposed approach using a truly 2-D HMM, the unconstrained handwritten numeral database of Concordia University of Canada has been used. It consists of 6000 unconstrained numerals originally collected from dead letter envelopes by the U.S. Postal Services at different locations in the U.S. Among them, 4000 numerals were used for training and 2000 numerals were used for testing. The numerals of this database were digitized in bilevel on a $64 \times 224$ grid of 0.153 mm square elements, giving a resolution of approximately 166 PPI.

### 6.2. The effect of the different number of states and observation symbols

The recognition rate is affected by the number of states and the number of observation symbols of HMMRF model. The determination of optimal number of states and observation symbols is almost always based on empirical finding. Table 1 shows recognition rates using the different number of states and observation symbols. Each observation symbol is represented by the intensity of input grey-level image. For the test set, the results range from 77.6% for 4 state model to 91.7% for 8 state model when the number of observation symbols per state is 16. From this, we see that there is a trend of global improvement as the number of states increases; however, as the number of states goes from 6 to 8, there is a slight performance improvement in spite of increasing computational complexity of the estimation algorithm. Therefore, 6 state model with 16 observation symbols is found to be adequate in the context of the current test.

### 6.3. System comparison

*6.3.1. Comparison of the re-estimation algorithms.* In order to examine the validity of the proposed re-estimation algorithm developed in Section 5, we have compared it with DD re-estimation algorithm. In this test, HMMRF models were designed with six states. Figure 9 shows the average log likelihood probability for typical training runs using the DD re-estimation algorithm and the proposed re-estimation algorithm, respectively. Training was continued until the increase of the average log likelihood probability between iterations becomes less than $2 \times 10^{-3}$ or convergence is achieved.

Learning curves for two re-estimation algorithms. The proposed re-estimation algorithm required only an average of 10 iterations for the convergence while the DD algorithm converged after about 40 iterations. Note that, however, we cannot guarantee convergence for both algorithms in principle although we have achieved convergence for 400 training data per digit in practice.

The recognition results using the two re-estimation algorithms are given in Table 2. As expected, the recognition result of the proposed re-estimation algorithm is better than that of the DD re-estimation algorithm.

*6.3.2. Comparison of filtering technique with fixed-lag smoothing technique.* To evaluate the validity of the fixed-lag smoothing based on the look-ahead

Table 1. Performance comparison using different number of states and observation symbols

| Number of states | Number of observation symbols | Recognition rates | |
|---|---|---|---|
| | | Training set (%) | Testing set (%) |
| 4 | 8 | 77.4 | 74.3 |
| | 16 | 81.1 | 77.6 |
| 6 | 8 | 90.5 | 88.0 |
| | 16 | 93.5 | 90.8 |
| 8 | 8 | 92.2 | 88.4 |
| | 16 | 94.7 | 91.7 |

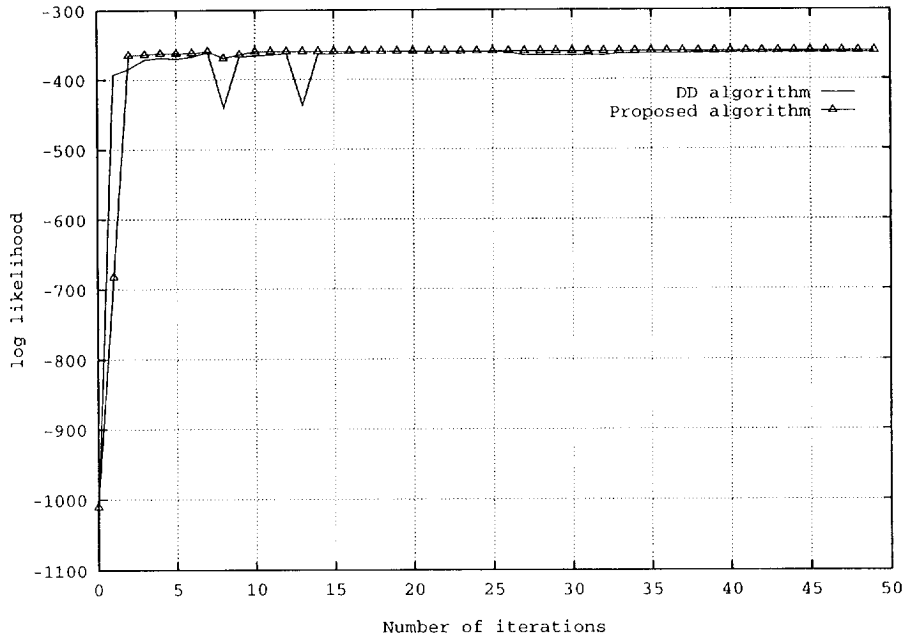



Fig. 9. Learning curves for two re-estimation algorithms.

Table 2. Comparison of the recognition results using different re-estimation algorithms

| Re-estimation algorithm | Recognition rates | |
|---|---|---|
| | Training set (%) | Testing set (%) |
| DD algorithm | 78.2 | 74.9 |
| Proposed algorithm | 93.5 | 90.8 |

Table 3. Recognition results of the filtering versus the fixed-lag smoothin techniques

| Decoding technique | Recognition rates | |
|---|---|---|
| | Training set (%) | Testing set (%) |
| Filtering | 82.2 | 78.0 |
| Fixed-lag smoothing | 93.5 | 90.8 |

technique, we investigate the performance comparison with the filtering. In our experiments, the filtering was performed by labeling pixels using $F_{m,n}(q) = P(\omega_{m,n}|\mathbf{X}_{1,1}^{m,n})$. Table 3 shows the recognition results obtained using the filtering and the fixed-lag smoothing techniques. It can be observed from Table 3 that in the presence of correlation between successive states, *look-ahead* observations convey significant information about $\omega_{m,n}$. From these results, it is apparent that the fixed-lag smoothing technique significantly outperforms the filtering.

*6.3.3. Comparison with HMM-based approaches.* It is natural that the 2-D technique is compared with its 1-D counterpart on the basis of its performance, measured by recognition rates and simplicity. Two HMM-based approaches have been considered:

- 1-D HMM[20]
- Pseudo 2-D HMM.[8]

Table 4 shows the performance of the HMMRF-based approach with those of the HMM-based approaches. The results reveal that the modeling of the 2-D nature is generally an important factor in recognition rates. We see that 1-D HMM without modeling 2-D nature gives the worst performance. After compensating for 2-D nature, the performance improved considerably as shown in the results of pseudo 2-D HMM and HMMRF model. HMMRF model gave

Table 4. Recognition results of three different models

| Model type | Number of states | Recognition rates | |
|---|---|---|---|
| | | Training set (%) | Testing set (%) |
| 1-D HMM | 8 | 87.2 | 82.0 |
| Pseudo 2-D HMM | 10 | 88.4 | 85.2 |
| HMMRF | 6 | 93.5 | 90.8 |

the best recognition result because the spatial correlation has been fully exploited.

Clearly, from the viewpoint of recognizer performance, the proposed HMMRF-based approach is more powerful than HMM-based approaches. At the computational complexity, however, HMMRF-based approach is not so efficient as the HMM-based approaches because the decoding procedure in HMMRF model has much higher complexity than the corresponding parts in HMM. In a third-order HMMRF model the decoding algorithm requires the total complexity of $O(\mathcal{G}^4)$ operations per pixel where $\mathcal{G}$ is the number of states.

In summary, a remark is possible that the proposed HMMRF-based approach is promising for the recognition of handwritten characters with large variations and distortions. It is due to its ability to cope with the variation in handwritten characters by means of statistical modeling.

## 7. CONCLUSIONS

In this paper, we have presented a new framework for the recognition of handwritten characters using a truly 2-D model: hidden Markov mesh random field (HMMRF). The HMMRF model which is a statistical model for 2-D image modeling and recognition is an extension of a 1-D HMM to 2-D and it can provide a better description of the 2-D nature of characters. The character recognition based on HMMRF model consists of the training phase and the decoding phase. Our optimization criterion for training and decoding is based on the maximum, marginal *a posteriori* probability. We have also developed a new formulation of parameter estimation for off-line character recognition. Computational concerns in 2-D, however, necessitate certain simplifying assumptions on the model and approximations on the implementation of the estimation algorithm. In this paper, the image is modeled as a third-order MMRF which is characterized by causal conditional distributions, and the proposed estimation algorithm is applied over the look-ahead observations rather than over the entire image. Thus, the formulation has been derived from the extension of the look-ahead technique devised for a real-time decoding.

The experimental results revealed that the proposed approach is capable of providing desirable performance on the task of recognizing handwritten characters under reasonable modeling assumptions. Although the proposed estimation algorithm has yielded good recognition results for handwritten characters, it is desirable to reduce the computational complexity of the algorithm for practical application. Also we expect further improvements of the proposed approach by generating discriminant symbols using more robust and efficient feature preserving the spatial continuity of image.

## APPENDIX A

### A.1. *Proof of Theorem 2*

As stated earlier in re-estimation procedure of $P_{q|r,s,t}$, it is possible to decompose $\mathbf{L}_{1,1}^{m+1,2}\backslash(m+1,2)$ as

$$\mathbf{L}_{m+1,2}\backslash(m+1,2) = \mathbf{L}_{1,1}^{m,1} \cup \mathbf{L}_{m,2}^{C} \cup \mathbf{L}_{m+1,1}^{R}.$$

Then, $\mathbf{L}_{m+1,2}^{C}(u, r, v, q, y)$ is developed as follows:

$$\mathbf{L}_{m+1,2}^{C}(u, r, v, q, y)$$

$$\propto P\left(\begin{matrix}\omega_{m-1,1}=r\\ \omega_{m,1}=q\end{matrix}, \mathbf{X}_{1,1}^{m,1}\right) P\left(\begin{matrix}\omega_{m-1,2}=u\\ \omega_{m,2}=v\end{matrix}, \mathbf{X}_{m,2}^{C} \middle| \begin{matrix}\omega_{m-1,1}=r\\ \omega_{m,1}=q\end{matrix}, \mathbf{X}_{1,1}^{m,1}\right)$$

$$\times P\left(\omega_{m+1,1}=y, \mathbf{X}_{m+1,1}^{R} \middle| \begin{pmatrix}\omega_{m-1,1}=r\\ \omega_{m,1}=q\end{pmatrix}, \mathbf{X}_{1,1}^{m,1}, \begin{pmatrix}\omega_{m-1,2}=u\\ \omega_{m,2}=v\end{pmatrix}, \mathbf{X}_{m,2}^{C}\right)$$

$$= P\left(\begin{matrix}\omega_{m-1,1}=r \\ \omega_{m,1}=q\end{matrix},\ \mathbf{X}_{1,1}^{m,1}\right) P\left(\begin{matrix}\omega_{m-1,2}=u \\ \omega_{m,2}=v\end{matrix},\ \mathbf{X}_{m,2}^{C} \middle| \begin{matrix}\omega_{m-1,1}=r \\ \omega_{m,1}=q\end{matrix},\ \mathbf{X}_{1,1}^{m,1}\right)$$

$$\times P(\omega_{m+1,1}=y, \mathbf{X}_{m-1,1}^{R} | \omega_{m,1}=q,\ \mathbf{X}_{1,1}^{m,1})$$

$$\propto \frac{P\left(\begin{matrix}\omega_{m-1,1}=r & \omega_{m-1,2}=u \\ \omega_{m,1}=q & \omega_{m,2}=v\end{matrix} \middle| \mathbf{X}_{1,1}^{m,2}\right) P\left(\begin{matrix}\omega_{m,1}=q \\ \omega_{m+1,1}=y\end{matrix} \middle| \mathbf{X}_{1,1}^{m,2}\right)}{P(\omega_{m,1}=q | \mathbf{X}_{1,1}^{m,1})}. \tag{A1}$$

By using definition of $H, Z$ and $F$, we can get Theorem 2.

## REFERENCES


1. F. Jelinek, Continuous speech recognition by statistical method, *Proc. IEEE* **64**(4), 532–536 (1976).
2. L. R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, *Proc. IEEE* **77**(2), 257–286 (1989).
3. C. Bose and S.-S. Kuo, Connected and Degraded text recognition using hidden Markov model, *Pattern Recognition* **27**(10), 1345–1363 (1994).
4. A. Kundu and P. Bahl, Recognition of handwritten script: a hidden Markov model based approach, *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, New York City, USA, pp. 928–931 (April 1988).
5. A. Kundu, Y. He and P. Bahl, Recognition of handwritten word: first and second order hidden Markov model based approach, *Pattern Recognition*, **22**(3), 283–297 (1989).
6. M.-Y. Chen, A. Kundu, and J. Zhou, Off-line Handwritten word recognition using a hidden Markov model type stochastic network, *IEEE Trans. Pattern Anal. Mach. Intell.* **16**(5), 481–496 (1994).
7. G. E. Kopec and P. A. Chou, Document image decoding using Markov source models, *IEEE Trans. Pattern Anal. Mach. Intell.* **16**(6), 602–617 (1994).
8. E. Levin and R. Pieraccini, Dynamic planar warping for optical character recognition, *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, San Francisco, California, pp. 23–26 (March 1992).
9. O. E. Agazzi and S.-S. Kuo, Hidden Markov Model Based Optical Character Recognition in the Presence of Deterministic Transformations, *Pattern Recognition*. **26**(12), 1813–1826 (1993).
10. S.-S. Kuo and O. E. Agazzi, Keyword spotting in poorly printed documents using pseudo 2-D hidden Markov models, *IEEE Trans. Pattern Anal. Mach. Intell.* **16**(8), 842–848 (1994).
11. M.-Y. Chen, A. Kundu and S. N. Srihari, Variable duration hidden Markov model and morphological segmentation for handwritten word recognition, *IEEE Trans. Image Process.* **4**(12), 1675–1688 (1995).
12. H. Derin and P. A. Kelly, Discrete-index markov-type random processes, *Proc. IEEE* **77**(2), 257–286 (1989).
13. K. Abend, T. J. Harley and L. N. Kanal, Classification of binary random patterns, *IEEE Trans. Inform. Theory*, **11**, 538–544 (1965).
14. M. P. Ekstrom and J. W. Woods, Two-dimensional spectral factorization with applications in recursive digital filtering, *IEEE Trans. Acoust. Speech Signal Process.* **24**, 115–128 (1976).
15. F.-C. Jeng and J. W. Woods, On the relationship of the Markov mesh to the NSHP Markov chain, *Pattern Recognition Lett.* **5**, 273–279 (1987).
16. P. A. Devijver, Hidden Markov mesh random field models in image analysis, in *Advances in Applied Statistics (Statistics and Images 1)*, K. V. Mardia and G. K. Kanji, eds. Carfax, Abingdon. pp. 187–227 (1993).
17. H.-S. Park and S.-W. Lee, Hidden Markov mesh random field: theory and its application, *Proc. 3rd Int. Conf. on Document Analysis and Recognition*, Montreal, Canada, pp. 409–412 (August 1995).
18. L. E. Baum, An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes, *Inequalities* **3**, 1–8 (1972).
19. J. C. Anigbogu and A. Belaid, Performance evaluation of an HMM based OCR system, *Proc. 11th Int. Conf. on Pattern Recognition*, pp. 565–568. The Hague, The Netherlands. (August 1992).
20. H.-S. Park and S.-W. Lee, Off-line recognition of large-set handwritten characters with multiple hidden Markov models, *Pattern Recognition* **29**(2), 231–244 (1996).
21. G. Boccignone, A. Chianese, L. P. Cordella and A. Marcelli, Recovering dynamic information from static handwriting, *Pattern Recognition* **26**(3), 409–418 (1993).
22. W. G. Cho, S.-W. Lee and J. H. Kim, Modeling and recognition of Cursive words with hidden Markov models, *Pattern Recognition* **28**(12), 1941–1953 (1995).
23. M. Gilloux, Handwritten digit recognition using Markov Meshes, *Proc. 4th Int. Workshop on Frontiers in Handwriting Recognition*, pp. 107–114. Taipei, Taiwan, (December 1994).
24. J. Besag, On the statistical analysis of dirty pictures, *J. Roy. Statist. Soc. Ser. B* **48**(3) 259–302 (1986).
25. H. Derin, H. Elliot, R. Christi and D. Geman, Bayes smoothing algorithms for segmentation of binary images modeled by Markov random fields, *IEEE Trans. Pattern Anal. Mach. Intell.* **6**(6), 707–720 (1984).
26. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. Wiley-InterScience, New York: (1973).
27. A. Krzyzak, W. Dai and C. Y. Suen, Unconstrained handwritten character classification using modified backpropagation model, *Proc. 1st Int. Workshop on Frontiers in Handwriting Recognition*, pp. 155–166. Montreal, Canada. (April 1990).


**About the Author**—HEE-SEON PARK received the B.S. degree in computer science and statistics from Chungbuk National University, Cheongju, Korea, in 1991, and the M.S. and Ph.D. degrees in computer science from Chungbuk National University, Cheongju, Korea, in 1993 and 1996, respectively. Since 1996, she has been working as a senior research engineer at Samsung Electronics Co., Ltd., Suwon, Korea. Her research interests include stochastic modeling, image processing, and speech recognition.

**About the Author**—SEONG-WHAN LEE received the B.S. degree in computer science and statistics from Seoul National University, Seoul, Korea, in 1984 and the M.S. and Ph.D. degrees in computer science from the Korea Advanced Institute of Science and Technology in 1986 and 1989, respectively.

In 1987, he worked as a visiting researcher at the Pattern Recognition Division, Delft University of Technology, Delft, the Netherlands. He was a visiting scientist at the Centre for Pattern Recognition and Machine Intelligence, Concordia University, Montreal, Canada, during the winter of 1989 and the summer of 1990. From 1989 to 1994, he was an Assistant Professor in the Department of Computer Science, Chungbuk National University, Cheongju, Korea. In March 1995, he joined the faculty of the Department of Computer Science and Engineering, Korea University, Seoul, Korea, as an Associate Professor. His research interests include pattern recognition, document image analysis, content-based image retrievel, computer vision, and neural networks.

He was the winner of the Annual Best Paper Award of the Korea Information Science Society in 1986. He obtained the First Outstanding Young Researcher Award at the 2nd International Conference on Document Analysis and Recognition in 1993, and the First Distinguished Research Professor Award from Chungbuk National University in 1994. In 1996, he also obtained the Outstanding Research Award from the Korea Information Science Society. He has more than 100 publications on pattern recognition and neural networks in Internaational Journals and Conference Proceedings, and authored two Korean books: Theory and Practice of Character Recognition, Hongneung Press (1993) and Principles of Pattern Recognition, Hongneung Press (1994).

He is the Co-Editor-in-Chief of International Journal of Document Analysis and Recognition, and the Associate Editor of the Pattern Recognition Journal, International Journal of Pattern Recognition and Artificial Intelligence, and International Journal of Computer Processing of Oriental Languages. He was the Program Chairman of the 17th International Conference on Computer Processing of Oriental Languages and the 6th International Workshop on Frontiers in Handwriting Recognition. He is the General Co-chairman of 3rd International Workshop on Document Analysis Systems, and the Program Co-chairman of the 5th International Conference on Document Analysis and Recognition and 2nd International Conference on Multimodal Interface. He served on the program committees of several well-known international conferences.

He is a fellow of International Association for Pattern Recognition, a senior member of the IEEE Computer Society and a life member of the Korea Information Science Society, the Pattern Recognition Society, the International Neural Network Society, and the Oriental Language Computer Society.