Handprinted Character Recognition Based on Spatial Topology Distance Measurement

Cheng-Yuan Liou, *Member, IEEE*, and Hsin-Chang Yang

Abstract—In this work we present a self-organization matching approach to accomplish the recognition of handprinted characters drawn with thick strokes. This approach is used to flex the unknown handprinted character toward matching its object characters gradually. The extracted character features used in the self-organization matching are center loci, orientation, and major axes of ellipses which fit the inked area of the patterns. Simulations provide encouraging results using the proposed method.

Index Terms—Handprinted character recognition, spatial topology distance, self-organizing map, neural networks, elastic matching.

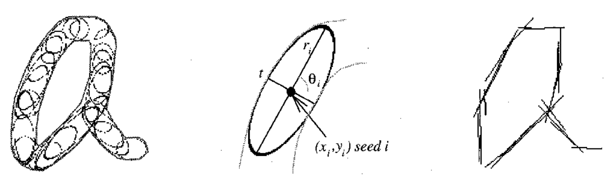
1 Introduction

We briefly review the handprinted character recognition techniques for thick strokes and discuss their difficulties. The difficulties are mainly arisen from the various flexible distortions produced during handwriting. Robust techniques on the thinning method, correlation matching, elastic matching, and distance measurement are the main focuses for solving such difficulties.

Most recognition systems extract features from the skeletons which are obtained by applying well thinning algorithms to the thick handprinted patterns. The skeletons are always obtained in advance to simplify the representation and to reduce the computation cost. However, there is no evidence that human eyes perform the same thinning process to the input pattern. The thinning process may not be the only choices. Besides, for a character with complex structure, it is hard to obtain correct features from its skeleton. This is because the thinning process often distorts the structure, especially in the intersections, joints, and the ends. There are serious spurious pixels always occurred in the intersections, turnings and forks. These pixels may mislead the features for further processing.

So far many kinds of classification methods have been developed based on various feature representations. One of the methods is the correlation matching method. This method compares the unknown input pattern with all the standard template patterns in database and measures the distances between them according to certain distance measurement. Two major concerns of the correlation matching are 'where' and 'how' to measure the distance. For the first concern, the correspondence across feature points of those two character patterns must be found. One way in finding the correspondence is the elastic matching method[1], [2], [3]. The elastic matching is used for matching nonlinearly aligned point pairs. It provides a flexible correspondence between feature points across two distorted patterns. Since the structure of a thinned skeleton could be much distorted, all elastic matching methods do not satisfactorily solve the topological correlation between two sets of feature points. The main problem in applying elastic matching is that they use simple attributes for local feature points, such as position and slope informations. To solve the second concern we must define distance measurement based on this correspondence. All common distance measurements cannot be directly applied to a distorted structure effectively.

To overcome the drawbacks of using the skeleton with improper information on local structure, we propose a robust feature representation for the character pattern. The proposed representation is designed to capture the whole local information of a stroke to support the global structure of a character. The unknown input handprinted pattern is normalized, preprocessed and translated into N ellipses as its features, $\{e_i, 1 \le i \le N\}$. See Fig. 1a for these ellipses. Each ellipse is fully extended within the local stroke region. The center locus of the ellipse is right on a preselected skeleton pixel. These N ellipses are used to represent the unknown pattern. The same processes are applied to the M standard (template) character patterns, each of which possesses c_j ellipses, $\{g_{jk}, 1 \le k \le c_j\}$, $1 \le j \le M$. We express each ellipse e_i as a four-dimensional (4D) feature vector $\mathbf{e}_i = [x_i, y_i, r_i, \theta_i]$. These four parameters are shown in Fig. 1b. The first two real numbers, x_i and y_i, denote the coordinate of the center of the ellipse, and is located on a regularly preselected skeleton pixel. The r_i denotes the length of the major axis, and θ_i denotes the orientation of the major axis. Each vector denotes a feature point in the 4D space. The two parameters r_i and θ_i can provide constructive information. The improper information can be effectively removed by using these ellipses. Since the width of the stroke depends on the pen and provides little information on the structure, we neglect the minor axis in the vector. The same representation can also be obtained for the ellipse g_{j_k} to get the vector $\mathbf{g}_{j_k}$. We will use $\mathbf{x}$ and $\mathbf{c}_j$ in the following context to denote the unknown handprinted feature collection and the jth standard template feature collection respectively, then $\mathbf{x} = \{\mathbf{e}_i, 1 \le i \le N\}$ and $\mathbf{c}_j = \{\mathbf{g}_{jk}, 1 \le k \le c_j\}$.



(a) (b) (c)

Fig. 1. The ellipses fitted in the stroke. (a) all ellipses in the character 'a' (b) a fitted ellipse and its parameters (c) the feature representations for 'a'.

Our purpose is to develop a quantitative approach for identifying the unknown pattern by using these ellipses. Later we will define a distance quantity, $\mathcal{D}(\mathbf{x}, \mathbf{c}_j)$, to measure the spatial topology distortion (or dissimilarity) between the unknown character $\mathbf{x}$ and each $\mathbf{c}_j$. The recognition is done by calculating all $\mathcal{D}(\mathbf{x}, \mathbf{c}_j)$, $1 \le j \le M$ and selecting the standard character $\mathbf{c}_{j^*}$ which is closest to the unknown character, where $D(\mathrm{x}, \mathrm{c}_{j^*}) = \min_j D(\mathrm{x}, \mathrm{c}_j)$, $1 \le j \le M$. In order to obtain $\mathcal{D}(\mathbf{x}, \mathbf{c}_j)$ we need two steps to do this. The first step involves an elastic matching to find the correspondence across the feature points of these two feature collections. A devised self-organizing map (SOM) is presented

• *The authors are with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan 10764, Republic of China. E-mail: cyliou@csie.ntu.edu.tw.*

Manuscript received Oct. 14, 1994; revised Nov. 22, 1995. Recommended for acceptance by R. Kasturi.
For information on obtaining reprints of this article, please send e-mail to: transpami@computer.org, and reference IEEECS Log Number P95181.

0162-8828/96$05.00 ©1996 IEEE

to accomplish this elastic matching. This SOM preserves the spatial topology of a pattern during matching. With this SOM, we can achieve better mapping correspondence. The other step involves defining a distance measurement based on such correspondence.

The rest of this correspondence is divided into four sections: Section 2 describes the method for obtaining the 4D representations of a character pattern. Section 3 contains the details of the devised self-organizing map network. We may add constraints to the SOM network to improve performance. Simulations are included in Section 4. We draw brief conclusions in the last section.

2 The 4-D Representation of Character Patterns

All characters must be normalized properly in advance. The features of an unknown handprinted pattern are represented by a set of 4D vectors $\{\mathbf{e}_i = [x_i, y_i, r_i, \theta_i], 1 \le i \le N\}$. These vectors are selected so that they could accurately capture all local stroke informations. We now show the way to obtain this 4D representation for a pattern.

We regularly sample seed pixels from the skeleton as the centers of the ellipses. We use the Voronoi method to obtain the skeleton. These seeds constitute the support of the character pattern. The seeds' coordinates are the x_i and y_i components of the 4D vector. The seeds may be regularly selected with various methods. We will use either concentric sampling or grid sampling to select seeds in this work.

The number of seeds selected is determined experimentally. Large number of seeds will result in heavy computational cost as well as redundant features but accurate result. On the other hand, too few seeds will not provide enough information about the character pattern. Intuitively, the number of skeleton pixels of a character should be large when the structure of the character is complex. Typically we select 100-200 seeds for each Chinese character according to the number of strokes of that character. In real applications we find it still adequate when the number of seeds is less than 100.

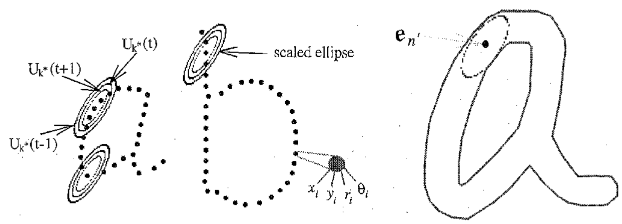
We then grow concentric circles with the center located on each seed. When the circle grown from seed i intersects with the outer boundary of a stroke at point t, we stop growing. The radius of this circle, a, is fixed as half the length of the minor axis and we start growing ellipse from this circle by increasing the length of major axis. The orientation of the major axis is perpendicular to $\overline{it}$. We grow the ellipse gradually according to the ellipse function $\frac{(x-x_i)^2}{a^2}+\frac{(y-y_i)^2}{b^2}=1$, where b is half the length of major axis. a is fixed during the growing process while b is increased gradually. The orientation of this ellipse may be slightly adjusted to obtain a better fitted ellipse within the local stroke. Fig. 1b shows a grown ellipse. The growing method is similar to the Voronoi process. This ellipse stops growing when it is totally confined by the outer boundary of the local stroke. The length and orientation of the major axis can be obtained to give the $r_i = 2b$ and θ_i components of the 4D vector associated with this seed. Note that the orientation of the major axis θ_i should be considered the same as its opposite direction, i.e. $\theta_i + \pi$. Thus we limit θ_i in the range $(0, \pi)$.

These 4D representations for a character pattern generated by the above algorithm provide a lot of essential feature information for our purpose. Fig. 1c depicts these representations. The θ_i component, which is the orientation of the major axis, provides very accurate local stroke orientation information. The r_i component indicates the extension and straightness of a local stroke. Small r_i may indicate the turnings, or ends, or joints of strokes. Spurious and noise pixels have much smaller r_i. This kind of representations can be used to identify different types of strokes in different regions of this 4D space.

3 The Devised SOM Network

We start the elastic matching by devising a modified SOM network. The formal SOM network consists of neurons which are located on the regular grid points in a two dimensional square map. The locations of the neurons in the map constitute the neuron support. When using the SOM network to perform the elastic matching, each neuron will try to match an input feature point. When the network converge, a correspondence between the input feature patterns and neuron support is obtained. In our network, the geometry of the neuron support is not a fixed square. Each neuron is located in a seed position on the standard character plane. The geometry of the neuron support is roughly similar to that of the skeleton of the standard character with less pixels.

The x_k and y_k components of $\mathbf{g}_{jk}$ constitute a plane coordinate for locating the kth neuron in the character plane. This means the kth neuron of the SOM network is located right at the coordinate (x_k, y_k). The SOM network contains c_j neurons. Each standard template character has its own neuron support. An example is shown in Fig. 2a.



(a) (b)

Fig. 2. The SOM (a) neuron supports and neighborhood for 'a' and 'b' (b) an input pattern $\mathbf{e}_n$.

This assignment of the neurons' positions has the property that it incorporates the topology correlation among the strokes of the standard template pattern into the neuron support. The neighbor neurons provide local and global topological information. This kind of structure information is hard to retain by other methods.

The learning process of the standard SOM is also modified to cope with the topological receptive field requirement. Each neuron in the neuron support contains four synapses (or weights) which are initialized with the values of the corresponding 4D feature vector. We use $\mathbf{w}_{jk}(t)$ to denote the weights of the kth neuron in the neuron support of the jth standard character pattern at learning time t. We set $\mathbf{w}_{jk}(0) = \mathbf{g}_{jk}$. For a set of unknown handprinted pattern features, $\{\mathbf{e}_n \mid 1 \le n \le N\}$, the SOM network performs the elastic matching by iteratively applying the following algorithm:

1) Randomly select an $\mathbf{e}_{n'}$ from the set $\mathbf{x} = \{\mathbf{e}_n \mid 1 \le n \le N\}$
2) Find the neuron k^* where $\left\|\mathbf{e}_{n'} - \mathbf{w}_{jk^*}(t)\right\| = \min_{1\le k\le c_j}\left\|\mathbf{e}_{n'} - \mathbf{w}_{jk}(t)\right\|$
3) Update the weights $\mathbf{w}_{jk}(t)$ by adding a factor

$$\Delta \mathbf{w}_{j_k}(t) = \alpha\left[\mathbf{e}_{n'} - \mathbf{w}_{j_k}(t)\right], \quad \text{for } k \in U_{j^*}(t) \tag{1}$$

4) if $t < T$, $t = t + \Delta t$ and return to 1(experimentally $\frac{T}{\Delta t}$ is within 50 ~ 100).

where T is a predetermined total learning iteration. In the algorithm, $U_{k^*}(t)$ is the neighborhood of neuron k^* at time t and α is the learning rate. The network converges by iteratively applying this algorithm with decreasing learning rate α and neighborhood U_{k^*}.

After the evolution at time T, each input pattern $\mathbf{e}_n$ is matched to a neuron k. This is not a precise matching, i.e., $\mathbf{e}_n \approx \mathbf{w}_{jk}(T)$. From the topology preserving property of the SOM [4], this approximation can be considered as an elastic matching since the neighboring points of the vector $\mathbf{e}_n$ are also within the proximity of the neuron support.

Generally U_{k^*} is defined as the set of neurons within a square or circle with center at neuron k^*. This simple selection is due to the lack of knowledge about the correlation among input feature points. The neighborhood is defined as all the neurons within an scaled ellipse which has center at neuron k^*. This elliptic neighborhood is geometrically similar to the input ellipse $\mathbf{e}_{n'}$ in terms of $r_{n'}$ and $\theta_{n'}$. We use this scaled ellipse as the boundary of neighborhood U_{k^*}. This is shown in Fig. 2. The ellipse at the correct position of a similar character pattern will allow more neurons being selected as neighbors. The local topology information of the input character pattern is enforced by this neighborhood selection. The scaling factor of the ellipse decreases as the convergence progress.

Two examples are shown in Fig. 3. Fig. 3a shows the evolution when the input character pattern and standard template pattern are both 'a'. Fig. 3b displays the case that the input handprinted character pattern is 'a' and the template pattern is 'b'. The circles denote the neuron support and the squares denote the convergence result. We can easily observe that both neuron supports effectively match this input character pattern. The dashed lines connecting circles and squares show the matching correspondence during evolution.

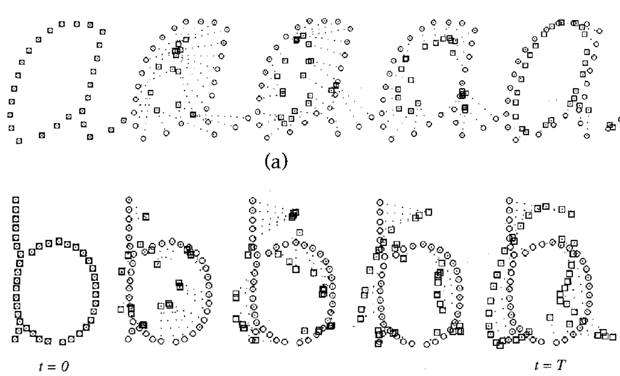


Fig. 3. The evolution of the matching using SOM.

With this correspondence, we can easily define a proper distance measurement which measures the similarity between the input feature pattern and the neuron support. The distance measurement is defined as

$$D(\mathbf{x}, \mathbf{c}_j) = \frac{1}{c_j} \sum_{1 \le k \le c_j} \left|\mathbf{w}'_{j_k}(0) - \mathbf{w}'_{j_k}(T)\right|, \tag{2}$$

where $\mathbf{w}'_{j_k}$ is a 2D vector which contains the x_i and y_i components of $\mathbf{w}_{jk}$. Remember that the neuron support is initialized by the 4D features of the standard template pattern, so $\mathbf{w}'_{j_k}(0)$ represents the geometric shape of the jth standard character. After time T, the weights of the neuron support contain values of matched input patterns. The distance is just the mean difference of the neurons' locations before and after elastic stretching. The two parameters r_i and θ_i are designed to assist the stretching.

The unknown character pattern is classified as the standard character pattern j^* if

$$D(\mathbf{x}, \mathbf{c}_{j^*}) = \min_j D(\mathbf{x}, \mathbf{c}_j), \quad 1 \le j \le M \tag{3}$$

The algorithm can be further improved by adding other constraints to isolate the unconnected strokes of the standard character. Such constraints are applied when the neurons are close in geometrical distance, but they are in separated strokes. This can be seen in Fig. 4. The neurons separated by a constraint line should not be considered as linked neighbors even if they are close geometrically.

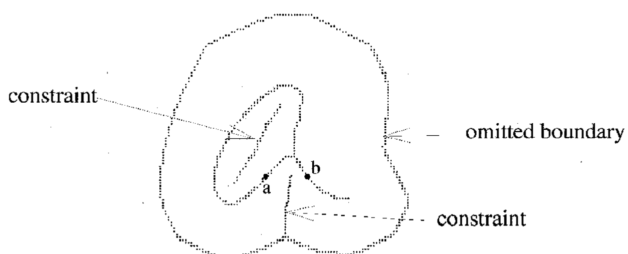


Fig. 4. Constraint to isolate the unconnected strokes.

We constructed a constraint set for each standard character to separate unconnected strokes in the standard character pattern. This constraint prevents the neurons in unconnected strokes being updated even they are close together and within the scaled ellipse's range. A proper constraint set can be obtained by finding the Voronoi boundary of the standard character strokes. An example of the Voronoi boundary is also shown in Fig. 4. The outer most boundary is omitted in our case. If two neurons are located at different sides of a Voronoi boundary (or more than one boundary), they cannot be considered as neighbors.

4 Simulations

In this section we give detailed simulations and compare the performance of the methods with and without constraints. The input handprinted character patterns are transformed to their 4D and 2D representations. The input patterns are randomly selected from a standard database.[1] There are total 62 character classes and we select 50 input patterns for each class. We manually discard those input patterns with poor resolution or cannot be easily verified by humans. Each character may have several standard template patterns. Here we use only one standard template for each character to simplify the simulations. Using grammars to improve the potential candidates is also omitted in our simulations. The performance is listed in Table 1. In this table we name the method in this work as spatial topology distance method in 4-dimensional space or STD(4D)

1. NIST Special Database 19.

method. The C-STD(4D) means the STD(4D) method with constraint. There is a similar method [5] which does not contain the r_i and θ_i parameters. We name the method in the reference [5] as STD(2D). In Table 1, the rejection rate is calculated using the credit threshold for each standard character pattern. Apparently the duration of the recognition process using 4D features are twice longer than that using 2D features.

TABLE 1
THE PERFORMANCE

Algorithm	Correct Rate	Reject Rate	Execution Time(PC)
STD(4 D)	92	1	3.5 sec
STD(2D)	88	2	1.8 sec
C-STD(4D)	95	1	3.6 sec
C-STD(2D)	92	1	1.9 sec

The accumulated deformation vectors after elastic matchings are shown in Fig. 5. The total 50 handprinted character patterns 'a' are presented to the neuron supports of standard character patterns 'a' and 'b' separately. The deformation vectors are drawn with line segments and are overlaid to obtain the figure. Long line segments denote serious deformations and will lie across large area. By the density of the overlaid deformation vectors we can easily evaluate the correctness of matching. The high density areas indicate a large amount of overlaid lines. In Fig. 5a the handprinted character pattern 'a' is presented to the neuron support of the standard character pattern 'a'. In Fig. 5b handprinted character 'a' is presented to neuron support of the standard character pattern 'b'. It is obvious that the area of high density in (b) is much larger than that in (a). This suggests that (b) contains many long line segments and is resulted from very bad distortion matching for wrong topology. In opposite, the high density area in Fig. 5a is localized along the contour of the neuron support 'a'. In this case, the standard character pattern 'a' tends to be the choice of the classification.

(a) (b)

Fig. 5. The accumulated deformation vectors of neuron supports.

The correctness of this classification can be further ensured by investigating the probability distribution of distances measured after many various matchings. The distribution of the measured distances is shown in Fig. 6. For each standard character pattern we show the distance distribution for both correct and incorrect input handprinted character patterns. Here a 'correct matching' is by performing elastic matching between the handprinted and standard character patterns of the same class, e.g., handprinted character pattern and standard character patterns are both 'A'. On the other hand, 'incorrect matching' is matched between patterns of different classes, e.g., 'A' and 'B'. The left column of line segments for each standard character depicts the distance distribution of correct matchings. The right column depicts the distribution of incorrect matchings. The length of each horizontal line segment displays the probability distribution of the distance. We observe that there is a gap in between the maximum of the correct matchings and the minimum of the incorrect matchings. The credit threshold is in the middle of this gap. This suggests the handprinted character pattern can always be classified. The simulations provide a sound support for the developing of the 4D representations and the proposed SOM method.

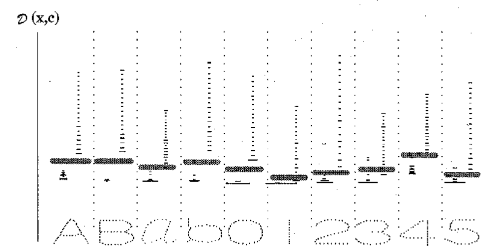


Fig. 6. The distributions of distance.

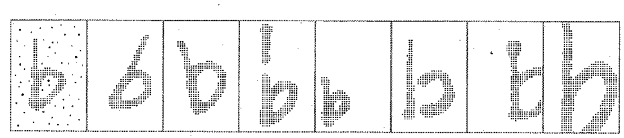
5 Discussions

As for the biological interpretations of our method, the ellipses are very similar to the 'logons' proposed by D. Gabor [6], [7]. Logons are two-dimensional Gaussian-weighted sinusoids which are used as the basis functions for analyzing image signals. The logon is a natural choice for use in image analysis since its compatibility with human visual system [8] and the projection of a logon is a scaled ellipse in the plane. This provides the biologically plausible foundations for the 4D features. Furthermore, the receptive fields of human eyes' visual cells are generally ellipses. So these ellipses can be considered as the receptive fields of certain pseudo visual cells.

The ellipse with long major axis is also similar to the directional bar excitation in the brain [9]. The ellipses with long r_i configure the main structure of a character. Our method may provide bases for the perception of a character structure in our visual system. For complicated images, such as color images, the ellipses for the green, blue, red color can be the bases for extracting sophisticated features. Our method can be extended to capture other intra-ellipse and even inter-ellipse information.

The proposed classification methods have been used to develop a prototype of automatic mailing address recognition system. This system recognizes isolated handprinted character patterns. The character patterns are written within formatted boxes to simplify the segmentation. Grammar rules of the addresses is incorporated into the recognition process to improve the accuracy and reduce the search. Fig. 7a shows the standard templates for the character 'a' used in the system. Fig. 7b shows several typical handprinted characters which can be classified by the proposed method.

(a)



(b)

Fig. 7. (a) Templates for character 'a'. (b) handprinted characters which can be classified.

ACKNOWLEDGMENT

This work was supported by the National Science Council.

REFERENCES

[1] D.J. Burr, "Elastic Matching of Line Drawings," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 3, pp. 708-713, Nov. 1981.
[2] D.J. Burr, "Designing a Handwriting Reader," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 5, pp. 554-559, Sept. 1983.
[3] G.E. Hinton, C.K.I. Williams, and M.D. Revow, "Adaptive Elastic Models for Hand-Printed Character Recognition," J.E. Moody, S.J. Hanson, and R.P. Lippmann, eds., *Advances in Neural Information Processing Systems* 4. Morgan Kauffmann, 1992.
[4] T. Kohonen, Self-Organization and Associative Memory, 3rd edition. Berlin, Heidelberg, Germany: Springer-Verlag, 1989.
[5] C.Y. Liou and H.C. Yang. "Spatial Topology Distance for Hand-printed Character Recognition," *Proc. Int'l Conf. Artificial Neural Networks*, Amsterdam, 1993.
[6] J.G. Daugman, "Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression," *IEEE Trans. Acoustics, Speech & Signal Processing*, vol. 36, no. 7, pp. 1,169-1,179, 1988.
[7] D. Gabor, "Theory of Communication," *J. IEE*, vol. 93, pp. 429-457, 1946.
[8] D.H. Hubel and T.N. Wiesel, "Brain Mechanisms of Vision," I. Rock, ed, *The Perceptual World: Readings from Scientific American Magazine*, pp. 3-24, 1990.
[9] G.G. Blasdel and G. Salama, *Nature*, vol. 321, pp. 579-585, 1986.

Document Image Decoding by Heuristic Search

Anthony C. Kam and Gary E. Kopec, *Member, IEEE*

Abstract—This correspondence describes an approach to reducing the computational cost of document image decoding by viewing it as a heuristic search problem. The kernel of the approach is a modified dynamic programming (DP) algorithm, called the *iterated complete path* (ICP) algorithm, that is intended for use with separable source models. A set of heuristic functions are presented for decoding formatted text with ICP. Speedups of 3–25 over DP have been observed when decoding text columns and telephone yellow pages using ICP and the proposed heuristics.

Index Terms—Document image decoding, Markov models, heuristic search, dynamic programming.

1 INTRODUCTION

DOCUMENT image decoding (DID) is an approach to document recognition that is based on an explicit communication theory view of the processes of document creation, transmission, and recognition [2]. In the DID model, a stochastic *message source* selects a finite string M from a set of candidate strings according to a prior probability distribution. An *imager* converts the message into an ideal binary image Q. A *channel* maps the ideal image into an observed image Z by introducing distortions due to printing and scanning, such as skew, blur, and additive noise. Finally, a *decoder* receives image Z and produces an estimate $\hat{M}$ of the original message according to a maximum a posteriori (MAP) decision criterion.

Much of the recent work in DID has focussed on a class of combined source/imager models called separable Markov sources [4]. Loosely, a separable source is one that may be factored into a product of one-dimensional models that represent horizontal and vertical structure, respectively. MAP decoding of an image with respect to a separable model can be implemented using a nested dynamic programming (DP) algorithm called the *separable Viterbi algorithm* [4].

The time complexity of separable Viterbi decoding is $\mathbf{O}(B_h \times H \times W)$, where B_h is the number of branches in the horizontal models and H and W are the image height and width, respectively, in pixels. The factor $B_h \times W$ represents the cost of using the horizontal models to decode a single image row while the factor H arises because decoding is repeated at every row. Although the computation grows only linearly with the number of image pixels, in absolute terms it can be prohibitive. For example, decoding an 8.5 in × 11 in image scanned at 300 ppi using a simple text column model requires about 40 *minutes*. Thus, methods for decreasing the required computation are essential if DID is to become a widely-used approach to document image analysis.

The basic source of complexity in image decoding is the need to decode every image row. If the horizontal models were run only along the actual text baselines, the cost would decrease by the factor $\frac{H}{L}$, where L is the number of text lines. For example, with

- *A.C. Kam is with Caliper Corporation. This work was performed while he was a graduate student at the Massachusetts Institute of Technology.*
- *G.E. Kopec is with Xerox Corp., Palo Alto Research Center, Xerox PARC, 3333 Coyote Hill Rd., Palo Alto, CA 94304. E-mail: kopec@parc.xerox.com.*

Manuscript received Feb. 23, 1995; revised Dec. 4, 1995. Recommended for acceptance by J.J. Hull.
For information on obtaining reprints of this article, please send e-mail to: transpami@computer.org, and reference IEEECS Log Number P95182.

0162-8828/96$05.00 ©1996 IEEE