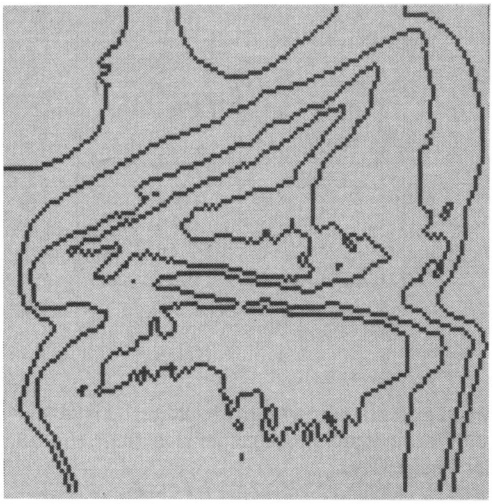


Fig. 12. Edge detected output using (13c) and $r = 8$ corresponding to Fig. 7(b).

play the role of creating different amounts of fuzziness in property domain. The intermediate smoother helps both in retrieving some pixel intensity lost by previous enhancement operations and in selecting the crossover points for the following final enhancement. Edge detection is done using min or max operators within the neighboring pixels. The edge intensity increases with the number of successive uses of the INT operator.

Investigations were also reported [14], [15] in which the pre-enhancement operation of block E_1 is replaced by the histogram equalization technique [2] (a standard existing enhancement operation for images like X-ray pictures and landscape photographs that are taken under poor illumination). But the contours of the resulting edge detected output image as compared to the present algorithm were seen to contain more spurious wiggles which, in turn, make the task of their description and interpretation more difficult.

ACKNOWLEDGMENT

Provision of X-ray films by Prof. J. M. Tanner is gratefully acknowledged by the authors. The authors' thanks are also due to Dr. A. A. Hashim for his valuable cooperation in this project.

REFERENCES

[1] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*. New York: Academic, 1976.
[2] R. C. Gonzalez and P. Wintz, *Digital Image Processing*. London, England: Addison-Wesley, 1977.
[3] J. K. Agarwal, R. O. Duda, and A. Rosenfeld, Eds., *Computer Methods in Image Analysis*. New York: IEEE Press, 1977.
[4] *IEE Proc. Computer and Digital Techniques* (*Special Issue on Image Restoration and Enhancement*), vol. 127, p. 5, Sept. 1980.
[5] *IEEE Trans. Pattern Analysis Machine Intell.* (*Special Issue on Biomedical Pattern Analysis*), vol. PAMI-2, p. 5, Sept. 1980.
[6] J. M. Tanner, R. H. Whitehouse, W. A. Marshall, M. J. R. Healy, and H. Goldstein, *Assessment of Skeletal Maturity and Prediction of Adult Height* (*TW2 Methods*). New York: Academic, 1975.
[7] L. A. Zadeh, K. S. Fu, K. Tanaka, and M. Shimura, Eds., *Fuzzy Sets and Their Applications to Cognitive and Decision Processes.* London: Academic, 1975.
[8] P. P. Wang and S. K. Chang, Eds., *Fuzzy Sets–Theory and Applications to Policy Analysis and Information Systems*. New York: Plenum, 1980.
[9] S. K. Pal, "Studies on the application of fuzzy set theoretic approach in some problems of pattern recognition and man–machine communication by voice," Ph.D. dissertation, Calcutta Univ., Calcutta, India, 1978.
[10] S. K. Pal and D. Dutta Majumder, "Fuzzy sets and decision making approaches in vowel and speaker recognition," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, pp. 625–629, Aug. 1977.
[11] A. D. Allen, "Measuring the empirical properties of sets," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-4, pp. 66–73, Jan. 1974.
[12] S. K. Pal and R. A. King, "Image enhancement using smoothing with fuzzy sets," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-11, pp. 494–501, July 1981.
[13] Y. Nakagawa and A. Rosenfeld, "A Note on the use of local min and max operations in digital picture processing," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 632–635, Aug. 1978.
[14] S. K. Pal and R. A. King, "Histogram equalization with S and π functions in detecting X-ray edges," *Electron. Lett.*, vol. 17, pp. 302–304, Apr. 16, 1981.
[15] S. K. Pal, "Fuzzy set theory in grey tone image processing," Ph.D./D.I.C. dissertation, Imperial College, University of London, London, England, 1982.

Optimal Quadtrees for Image Segments

WILLIAM I. GROSKY AND RAMESH JAIN

***Abstract*—Quadtrees are compact hierarchical representations of images. In this paper, we define the efficiency of quadtrees in representing image segments and derive the relationship between the size of the enclosing rectangle of an image segment and its optimal quadtree. We show that if an image segment has an enclosing rectangle having sides of lengths x and y, such that $2^{N-1} \times \max(x, y) \leqslant 2^N$, then the optimal quadtree may be the one representing an image of size $2^N \times 2^N$ or $2^{N+1} \times 2^{N+1}$. It is shown that in some situations the quadtree corresponding to the larger image has fewer nodes. Also, some necessary conditions are derived to identify segments for which the larger image size results in a quadtree which is no more expensive than the quadtree for the smaller image size.**

***Index Terms*—Blueprint, grid size, image translation, optimal quadtree, partial quadtree.**

I. INTRODUCTION

Quadtrees are receiving increasing attention from researchers in computer graphics, image processing, cartography, and related fields. The quadtree representation of a region is based on successive subdivisions of the array into quadrants. A uniform quadrant of the image is represented by a leaf in the tree; a nonuniform quadrant is represented by an internal node, preparatory to its being further divided into its quadrants. Thus, the entire array is represented by the root node, the four quadrants by the four sons of the root node. This process is iterated. The leaf nodes, being of uniform color, represent those blocks for which no further subdivision is required. As an example, the 8 X 8 region shown in Fig. 1 is represented by the quadtree exhibited in Fig. 2; a white region is represented by a white node □; and a black region is represented by a black node ●. Note that the coordinate system we are using has the origin at the northwest corner, and that the positive

Manuscript received July 22, 1981; revised August 9, 1982.

W. I. Grosky is with the Intelligent Systems Laboratory, Department of Computer Science, Wayne State University, Detroit, MI 48202.

R. Jain was with the Intelligent Systems Laboratory, Department of Computer Science, Wayne State University, Detroit, MI 48202. He is now with the Department of Electrical and Computer Engineering, University of Michigan, Ann Arbor, MI 48109.

0162-8828/83/0100-0077$01.00 © 1983 IEEE

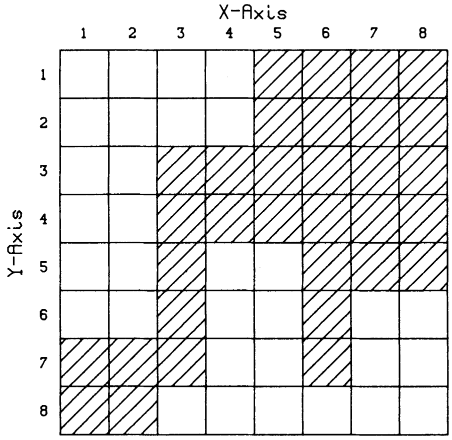


Fig. 1. An arbitrary region.

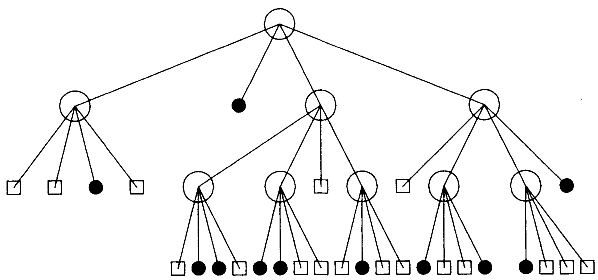


Fig. 2. The quadtree for the region of Fig. 1.

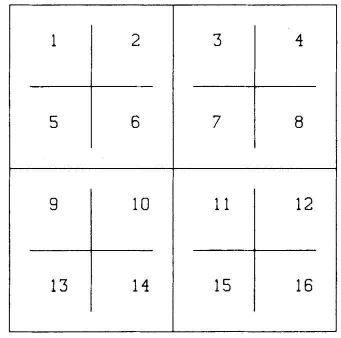


Fig. 3. A grid.

X and Y directions are to the right and downwards, respectively. Note also that the children of a node are in the order northwest, northeast, southeast, and southwest. This method of representation of an image is relatively compact [11] and is well suited to many operations in computer graphics and image processing [1]–[3], [7]–[12], [14]–[24].

Algorithms have been developed for conversion from other representations of regions, such as, boundary codes, binary arrays, and rasters, to quadtrees, and vice versa [3], [19]–[22]. The quadtrees constructed by the proposed algorithms do not necessarily have the minimum number of nodes. Samet [22] points out that one might be able to reduce the number of nodes by choosing a different location to serve as the origin.

In many raster graphics and image understanding applications, segments of an image need to be stored. Such segments are usually much smaller than the full image and must be stored efficiently. Conventionally, an array corresponding to the enclosing rectangle of the segment having a side parallel to the scan line is used. Such an enclosing rectangle, when converted to a quadtree, may result in a tree having too many nodes compared to the optimal tree.

This correspondence is concerned with optimal quadtrees for a given image. The optimality of quadtrees is judged on the basis of the number of leaves in the tree, as it can be shown that minimizing the number of leaves is the same as minimizing the number of nodes in a quadtree. Using this definition of optimality, we show that if an image segment has an enclosing rectangle having sides of lengths x and y, such that

$$2^{N-1} < \max(x, y) \leqslant 2^N,$$

then the optimal quadtree may be the one representing an image of size $2^N \times 2^N$ or $2^{N+1} \times 2^{N+1}$. It is shown that in some situations the quadtree corresponding to the larger image has fewer nodes. Also, some necessary conditions are derived to identify segments for which the larger image size results in a quadtree which is no more expensive than the quadtree for the smaller image size.

II. The Efficiency of a Quadtree

We will rank the efficiency of quadtrees by the number of their leaves. Notice that minimizing the number of leaves of a quadtree also minimizes its total number of nodes, as the number of nodes of a quadtree is directly proportional to the number of its leaves, as may be shown by a simple reworking of the proof of Lemma 5.2 in Horowitz and Sahni [6].

In this correspondence, we will only consider black figures on a white background. For this reason, if $T1$ and $T2$ are two quadtrees, we say that $T1 < T2$ iff the number of leaves of $T1$ is less than the number of leaves of $T2$. In this correspondence, we will be concerned with minimal quadtrees for image segments with respect to $<$.

III. Two Optimal Grid Sizes

Given IMAGE as defined above, we define its *enclosing rectangle* as the smallest rectangle which encloses IMAGE and whose sides are parallel to the coordinate axes. We let IMAGE$[X]$ and IMAGE$[Y]$ be the length of the horizontal and vertical sides, respectively, of its enclosing rectangle.

We will show the following.

Theorem 3.1: Let IMAGE be defined as above. Suppose $2^{N-1} < \max(\text{IMAGE}[X], \text{IMAGE}[Y]) \leqslant 2^N$ for $N \geqslant 1$. Let TREE$[j]$, $j \geqslant 1$, be the set of all quadtrees representing IMAGE based on a $2^{N+j} \times 2^{N+j}$ grid; that is, for any embedding of IMAGE in such a grid, the resultant quadtree is in TREE$[j]$. Then for any $T1 \in$ TREE$[p]$, $p \geqslant 2$, there exists a $T2 \in$ TREE$[p-1]$ such that $T2 < T1$.

Proof (Outline): Fig. 3 shows a $2^{N+p} \times 2^{N+p}$ grid divided into a 2×2 array of cells, each cell being a grid of size $2^{N+p-1} \times 2^{N+p-1}$. Each of the latter cells is also divided into a 2×2 array of cells, each cell being a grid of size $2^{N+p-2} \times 2^{N+p-2}$. IMAGE must lay within a square group of four contiguous cells, for example, cells 1, 2, 5, 6. In each case, it can be easily shown that for $T \in$ TREE$[p]$, there is a $T' \in$ TREE$[p-1]$ such that T has at least three more leaves than T', namely the tree whose root has the four subtrees in the square group. Hence, $T' < T$. Q.E.D.

IV. A Larger Grid Size is Sometimes Better

We have just shown that multiplying the size of the underlying grid of an image by 4, 8, 16, $\cdots$ always worsens the cost of the respective quadtree. Thus, we should either use a grid whose side is the smallest power of 2 larger than or equal to the longest side of the enclosing rectangle or a grid twice this size. Our next theorem shows that there are connected images which have cheaper quadtree representations when the latter choice is made for the grid size.

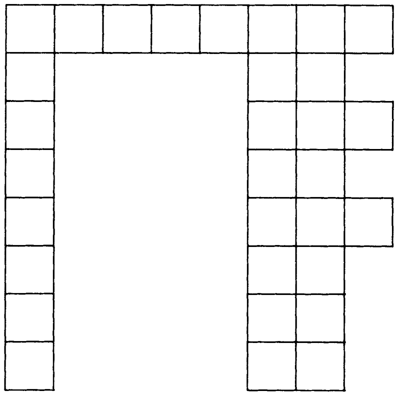


Fig. 4. IMAGE_K.

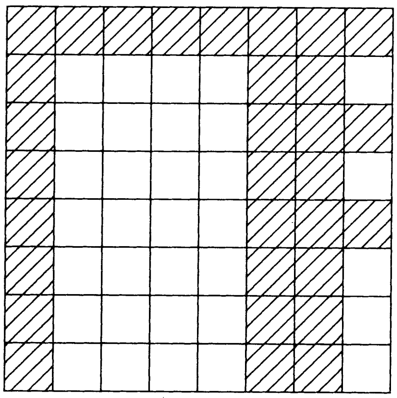


Fig. 5. The embedding of IMAGE_K.

Theorem 4.1: There exist connected images, IMAGE_k, for $k \geq 3$, which are such that

$$2^{k-1} < \max(\text{IMAGE}_k[X], \text{IMAGE}_k[Y]) \leq 2^k$$

and which have the property that there is a quadtree $T1$ of IMAGE_k based on a $2^{k+1} \times 2^{k+1}$ grid such that for all quadtrees $T2$ of IMAGE_k based on a $2^k \times 2^k$ grid, $T1 < T2$.

Proof: Consider IMAGE_k as in Fig. 4. Each cell shown is a $2^{k-3} \times 2^{k-3}$ grid of pixels. IMAGE_k, shown in Fig. 5, is the only embedding possible of IMAGE_k in a $2^k \times 2^k$ grid. Its quadtree has 31 black nodes and 24 white nodes. Consider the embedding of IMAGE_k in a $2^{k+1} \times 2^{k+1}$ grid as shown in Fig. 6. Its quadtree possesses 19 black nodes and 33 white nodes, thus proving our theorem. Q.E.D.

The following theorems are shown in a similar manner. See [4] for details.

Theorem 4.2: There exists a connected image, IMAGE_2, which is such that

$$2 < \max(\text{IMAGE}_2[X], \text{IMAGE}_2[Y]) \leq 4$$

and which has the property that there is a quadtree $T1$ of IMAGE_2 based on an 8×8 grid such that for all quadtrees $T2$ of IMAGE_2 based on a 4×4 grid, $T1$ and $T2$ have the same number of leaves, but $T1$ has fewer black leaves. Furthermore, there is no possible quadtree $T1$ which has fewer leaves than all possible $T2$.

Theorem 4.3: For $k \in \{0, 1\}$, all IMAGE_k such that

$$2^{k-1} < \max(\text{IMAGE}_k[X], \text{IMAGE}_k[Y]) \leq 2^k$$

and any quadtree $T1$ of IMAGE based on a $2^{k+1} \times 2^{k+1}$ grid,

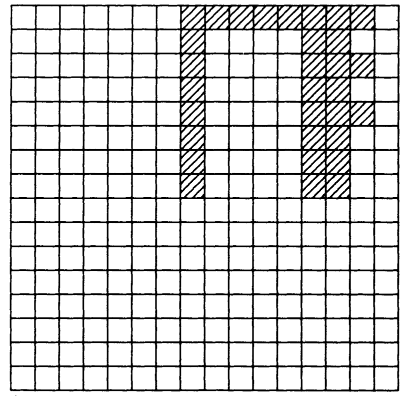


Fig. 6. Another embedding of IMAGE_K.

there exists a quadtree $T2$ of IMAGE based on a $2^k \times 2^k$ grid which is such that $T2 < T1$.

V. Conditions Where the Smaller Grid Size is Cheaper

We will now derive conditions under which the smaller grid size is cheaper. What this means is that for a particular image we will show when the cheapest possible quadtree using the larger grid size is still more expensive than the most costly quadtree possible using the smaller grid size. To derive these conditions, we will show how the number of leaves of a quadtree based on a $2^{N+1} \times 2^{N+1}$ grid changes when the embedded image, whose enclosing rectangle fits entirely within one quadrant of the grid, is translated by a given amount. It will be shown that if the number of rectangular blocks of certain given sizes consisting of either all black or all white pixels in our image is sufficiently small, any translation of the image cannot result in a quadtree with fewer leaves than the one for which the image lies completely within one of the grid's quadrants. This latter quadtree, in turn, will have more leaves than a quadtree based on a $2^N \times 2^N$ grid. We will also show something stronger than this. It has been previously noted [14] that small translations can perturb quadtree representations more so than large translations. The conditions we will derive on an image will ensure that a sequence of successively smaller translations of that image steadily worsens the resultant quadtree. Thus, if an image meets our conditions, it may be said to have an optimal grid size of $2^N \times 2^N$ in the *strong sense*.

Thus, let IMAGE be any image which is such that

$$2^{N-1} < \max(\text{IMAGE}[X], \text{IMAGE}[Y]) \leq 2^N.$$

Let *IMAGE* be any embedding of this image in a $2^N \times 2^N$ grid. We now assume, without loss of generality, that *IMAGE* is the northwest quadrant of a $2^{N+1} \times 2^{N+1}$ grid whose other quadrants consist only of white pixels.

We will show how the quadtree of this image on the larger grid changes when the grid is translated by $1, \cdots, 2^N - 1$ pixels in the positive X-direction. (It is obvious that translating the grid by 2^N pixels in the positive X-direction results in a quadtree of equal cost.)

Let us define translations $PXTR_j$ and $NXTR_j$, $0 \leq j \leq N - 1$, as a cyclic translation[1] of our $2^{N+1} \times 2^{N+1}$ grid by 2^j pixels

[1] By a cyclic translation we mean a translation in which both the left and right as well as the top and bottom sides of the grid are identified. Thus, no subimages are lost during the translation process. For example, a subimage being translated to the south off the bottom side of the grid would reappear on the top side.

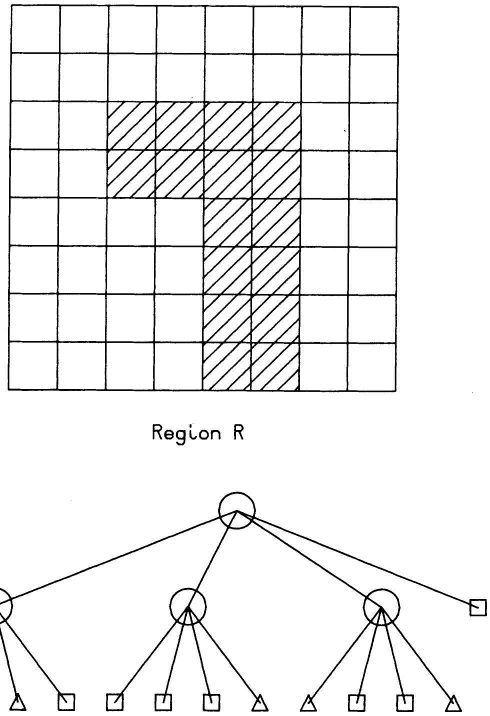


Fig. 7. A partial quadtree with respect to region R. (Green nodes are drawn as triangles.)

in the positive and negative X-directions, respectively. Also, let IDX be the null translation of our grid in the X-direction. Define an *X-translation vector* $VX = (XTr_1, \cdots, XTr_N)$ as an N element vector of translations such that for $1 \leqslant i \leqslant N$, $XTr_i \in \{PXTR_{N-i}, NXTR_{N-i}, IDX\}$ and, if $XTr_j = IDX$ for $1 \leqslant j \leqslant N$, then $XTr_k = IDX$ for all $j < k \leqslant N$. We speak of *applying* an X-translation vector to our grid, which connotes the act of first applying XTr_1, followed by $XTr_2, \cdots$, finally followed by XTr_N. The following result is easily shown by induction on N. See [4] for details.

Lemma 5.1: For $1 \leqslant d \leqslant 2^N - 1$, there is an X-translation vector $VX = (XTr_1, \cdots, XTr_N)$ such that applying VX to our grid results in an image which is the same as that resulting when the grid is translated d pixels in the positive X-direction.

Define the *order* of $VX = (XTr_1, \cdots, XTr_N)$, order (VX), as the number of non-IDX entries of VX. Define the jth *approximation* of VX as the X-translation vector.

$$(XTr_1, \cdots, XTr_j, IDX, \cdots, IDX).$$

Let us now define a *partial quadtree* as a quadtree in which leaves may be white, black, or green. Green nodes connote the fact that the finer structure at that node is unknown. A partial quadtree with respect to a region R of an image is a *partial quadtree with respect to a region R* of an image is a white regions of the image not in R, black leaves correspond to black regions of the image not in R, and green leaves correspond to that portion of the image which is contained in R. See Fig. 7 for an illustration of this concept. A *completion* of a partial quadtree is a quadtree which results upon substituting quadtrees for all green nodes, and then performing the appropriate reductions of coalescing four white or four black sons.

Now, let VX_0 be an X-translation vector of order 0. After applying VX_0 to our grid, the quadtree of the resulting image is seen to be a completion of the partial quadtree shown in Fig. 8(a). In general, suppose we have an X-translation vector VX_j of order $j \geqslant 1$. We will now show how to construct the partial quadtree QX_j of a completely white image with respect to the region into which the northwest quadrant is taken under the application of VX_j. Thus, the quadtree which results from an application of VX_j to our grid will be a completion of QX_j. This definition will be recursive. Let V^* be the $(j - 1)$st approximation of VX_j. (Note that V^* is of the order $j - 1$.) Suppose the partial quadtree QX_{j-1}, a completion of which is the quadtree which results from an application of V^* to our grid, is given. Then, QX_j results from QX_{j-1} by the following construction. Suppose the jth entry of VX_j is $PXTR_{N-j}$ or $NXTR_{N-j}$. Then, for every green or white node, respectively, on the highest numbered level, the root being of level 0, which has an ancestor which is the northwest son of the root, we substitute the tree shown in Fig. 8(b), while for every white or green node, respectively, on the highest numbered level which has an ancestor which is the northeast son of the root, we substitute the tree shown in Fig. 8(c). As an illustration, consider the X-translation vector $(PXTR_1, NXTR_0)$. We construct the required partial quadtree QX_2 by first constructing the partial quadtrees QX_0, QX_1 of (IDX, IDX) and $(PXTR_1, IDX)$, respectively. See Fig. 9(a)–(c) for sample images and their corresponding partial quadtrees QX_0, QX_1, QX_2.

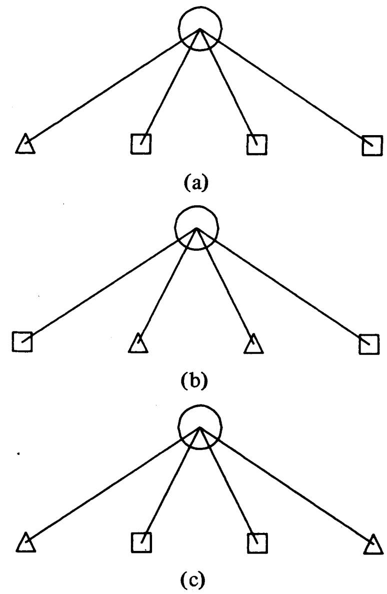
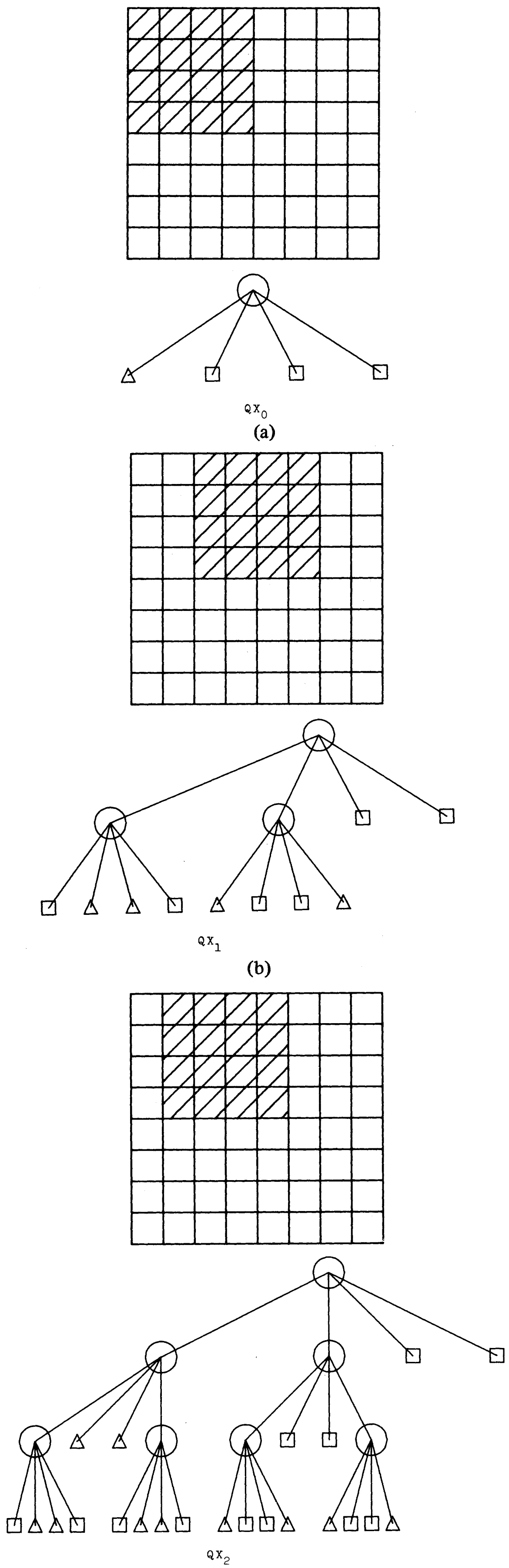


Fig. 8. Partial quadtrees.

Suppose we have two X-translation vectors VX_j and VX_k of orders j and k, respectively. Let TX_j and TX_k be the quadtrees which result after applying VX_j and VX_k, respectively. We would like to derive some condition on our image under which the number of leaves of TX_k is guaranteed to be greater than that of TX_j. This condition will turn out to be a bound on the number of regions of certain sizes of either all white or all black pixels.

We define a *block of size* $2^x \times 2^y$ as any rectangular area of pixels in our grid which extends for x pixels in the X-direction and y pixels in the Y direction. A *collection of disjoint blocks of size* $2^x \times 2^y$ is a set of such blocks, each pair of which contains no pixel in common. A *blueprint* of IMAGE is the sequence $(p_0, \cdots, p_N)$ where, for $0 \leqslant i \leqslant N$, p_i is the number of disjoint blocks of size $2^i \times 2^i$ of one color which are not contained in any block of the same color of size $2^{i+j} \times 2^{i+j}$, for $j \geqslant 1$. Informally, a blueprint is a sequence of counts of the number of maximal sized blocks of one color whose sides are powers of 2. We also define, for $n \geqslant 0$, $g(n) \triangleq n - \sum_{i=0}^{H} b_i$, where $b_H b_{H-1} \cdots b_0$ is the base 4 representation of n, and $f(n) \triangleq 3(n - \sum_{i=0}^{L} a_i)$, where $a_L a_{L-1} \cdots a_0$ is the base 2 representation of n.

We then have the following.



(c)
Fig. 9. Images and their partial quadtrees.

Theorem 5.1: For $1 \leqslant k \leqslant N$, let WE and WW be the cardinalities of the largest collections of disjoint blocks of size $2^{N-k} \times 2^{N-k+1}$ which are adjacent to the eastern and western boundaries of IMAGE, respectively, and which consist entirely of white pixels. Let W and B be the cardinalities of the largest collections of disjoint blocks of size $2^{N-k+1} \times 2^{N-k+1}$ which consist entirely of white pixels and entirely of black pixels, respectively, and which are at least 2^{N-k} pixels away from both the eastern and western boundaries of *IMAGE*. Define $M = \max(WE, WW)$ and $m = \min(WE, WW)$. For $0 \leqslant j < k$, suppose that

$$f(2M) + g(4m + 4W) + g(4B) < 3(2^k - 2^j) + 2 \sum_{i=1}^{k-1} p_{N-k+i+1}(4^i - 1).$$

Then we have $TX_j < TX_k$.

Proof (Outline): We will show that if the hypotheses of the theorem are true, then no matter what the shape of our image may be, the minimum number of leaves in TX_k is greater than the maximum number of leaves in TX_j.

Define T_k and T_j as the partial quadtrees which result from QX_k and QX_j, respectively, by substituting for each green node on level d, $0 < d \leqslant k$, the complete partial quadtree of height $k + 1 - d$, all of whose leaves are green. Thus, all green leaves of T_k and T_j occur on level $k + 1$. Note that TX_k and TX_j are completions of T_k and T_j, respectively.

As developed in [4], the number of leaves in TK_k is of the form $2^k 3 + L1 - L2$, where $2^k 3$ is the number of white leaves in T_k, $L1$ is the number of leaves in the quadtrees which are substituted for the green leaves of T_k in the process of going from T_k to TX_k, and $L2$ is the number of leaves saved through the coalescence of four white or four black nodes to one node in this process. Similarly, the number of leaves in TX_j is of the form $2^j 3 + L1 - L3$.

In [4], it is shown that the number of leaves saved through the coalescence of nodes on levels greater than $k + 1$ is the same for the process of going from T_k to TX_k as in going from T_j to TX_j. Thus, we may write $L2 = L + L4$ and $L3 = L + L5$, where $L4$ and $L5$ are the number of leaves saved through the coalescence of nodes on levels less than or equal to $k + 1$ in the processes of going from T_k to TX_k and T_j to TX_j, respectively.

Suppose that $L4 \leqslant F$ and $L5 \geqslant G$ for some $F, G \geqslant 0$ and that $F < 3(2^k - 2^j) + G$. We then have that

$$\begin{aligned} 2^j 3 + L1 - L3 &= 2^j 3 + L1 - L - L5 \leqslant 2^j 3 - G + L1 - L \\ &< 2^k 3 - F + L1 - L \leqslant 2^k 3 + L1 - L - L4 \\ &= 2^k 3 + L1 - L2. \end{aligned}$$

Thus, if we can show that $L4 \leqslant f(2M) + g(4m + 4W) + g(4B)$ and $L5 \geqslant 2 \sum_{i=1}^{k-1} p_{N-k+i+1}(4^i - 1)$, our result will follow.

There are $2M$ disjoint regions of white in *IMAGE*, each of size $2^{N-k} \times 2^{N-k}$. Assume, without loss of generality, that each of these regions is adjacent to the eastern boundary of *IMAGE*. There are then $2m$ disjoint regions of white in IMAGE, each of size $2^{N-k} \times 2^{N-k}$ and each adjacent to the western boundary of *IMAGE*. Also, there are $4W$ and $4B$ disjoint regions of which and black, respectively, in *IMAGE*, each of size $2^{N-k} \times 2^{N-k}$ and each at least 2^{N-k} pixels from both the eastern and western boundaries. Call the former two collections of regions *boundary regions* and the latter two collections *internal regions*.

As shown in [4], an upper bound on the coalescence of QX_k achievable with such regions may be computed as follows. For every 2^h such eastern boundary regions, we can decrease the number of white leaves by a maximum of $3(2^h - 1)$. For every $4^h - 2^{h+1}$ such internal white regions along with 2^h such western boundary regions, for $h \geqslant 1$, another decrease of up to $4^h - 1$ white leaves is possible. Finally, for every 4^h such internal white or internal black regions, a decrease of up to $4^h - 1$ white or black leaves, respectively, can occur. From these facts, one can show that $f(2M) + g(4m + 4W) + g(4B)$ is an appropriate upper bound for $L4$.

On the other hand, $L5 \geqslant 2 \sum_{i=1}^{k-1} p_{N-k+i+1}(4^i - 1)$ follows from the fact that for each block contained in *IMAGE* of size $2^t \times 2^t$ consisting of all white or all black pixels, any translation of *IMAGE* in the x-direction results in a minimum of two monocolored blocks, each of size $2^{t-1} \times 2^{t-1}$ and thus each represented by a leaf node. Q.E.D.

As an example of this theorem, consider the image of Fig. 5. Its blueprint is (28, 6, 1, 0). In the computation of TX_k, for $k \in \{1, 2\}$, we have that $WE = WW = W = B = 0$. Thus, $TX_0 < TX_1$ and $TX_0 < TX_2$, which may be easily verified. However, for $k = 3$, $WE = 1$, $WW = 0$, $W = 6$, and $B = 4$. Thus, the sufficient condition for $TX_0 < TX_3$ is $f(2) + g(24) + g(16) < 27$, which is not true. This does not mean that $TX_0 \not< TX_3$, however, as our condition is not necessary. It is, although, a sign that we should check it out and, indeed, we do get that $TX_3 < TX_0$.

We then have the following corollaries.

Corollary 5.1: Define WE and WW as the cardinalities of the largest collections of disjoint blocks of size 1×2 which are adjacent to the eastern and western boundaries of *IMAGE*, respectively, and which consist entirely of white pixels. Let W and B be the cardinalities of the largest collection of disjoint blocks of size 2×2 which consist entirely of white pixels and entirely of black pixels, respectively, and which are at least 1 pixel away from both the eastern and western boundaries of IMAGE. Define $M = \max(WE, WW)$ and $m = \min(WE, WW)$. Suppose that $f(2M) + g(4m + 4W) + g(4B) < 3(2^N - 1)$. Then $TX_0 < TX_k$ for $1 \leqslant k \leqslant N$.

Proof: Let hypotheses$_d$ consist of the following assumptions.

1) $\lfloor WE/2^{N-d} \rfloor$ and $\lfloor WW/2^{N-d} \rfloor$ are the cardinalities of the largest collections of disjoint blocks of size $2^{N-d} \times 2^{N-d+1}$ which are adjacent to the eastern and western boundaries of *IMAGE*, respectively, and which consist of all white pixels.

2) $\lfloor W/4^{N-d} \rfloor$ and $\lfloor B/4^{N-d} \rfloor$ are the cardinalities of the largest collections of disjoint blocks of size $2^{N-d+1} \times 2^{N-d+1}$ which consist of all white and all black pixels, respectively, and which are at least 2^{N-d} pixels away from both the eastern and western boundaries of *IMAGE*.

3) $M_d = \max(\lfloor WE/2^{N-d} \rfloor, \lfloor WW/2^{N-d} \rfloor)$, and

$$m = \min(\lfloor WE/2^{N-d} \rfloor, \lfloor WW/2^{N-d} \rfloor).$$

4) $f(2M_d) + g(4m_d + \lfloor 4W/4^{N-d} \rfloor) + g(\lfloor 4B/4^{N-d} \rfloor) < 3(2^d - 1)$.

Now, each entry of hypotheses$_N$ is true by the assumptions of the corollary. Suppose each element of hypotheses$_d$ is true for $1 \leqslant d' < d \leqslant N$. It is easy to see that clauses 1, 2, and 3 of hypotheses$_{d'}$ are true. From the fact that $f(\lfloor x/2 \rfloor) \leqslant (f(x) - 3)/2$ for $x > 1$, $g(\lfloor x/2 \rfloor) \leqslant g(x)/2$ for $x \geqslant 0$, and $g(\lfloor x/4 \rfloor) \leqslant (g(x) - 3)/4$ for $x > 3$, we can easily show that clause 4) of hypotheses$_{d'}$ is true. Thus, hypotheses$_d$ is true for $1 \leqslant d \leqslant N$ by induction, and our result follows from Theorem 5.1. Q.E.D.

We also have the following, whose proof is similar to that of Corollary 5.1.

Corollary 5.2: Let WE, WW, W, B, M, and m be the same as in Corollary 5.1. Suppose that $f(2M) + g(4m + 4W) + g(4B) < 2^{N-1}3$. Then $TX_{k-1} < TX_k$ for $1 \leqslant k \leqslant N$.

Corollaries 5.1 and 5.2 give sufficient conditions for the smaller grid size to be cheaper and strongly cheaper, respectively, than the larger grid size, with respect to translations of the image in the X-direction only. This is due to the fact that a quadtree based on the image being embedded in the northwest quadrant of the larger grid has three more leaves than the quadtree of this quadrant itself, which is the smaller grid. Similar results hold for translations in the Y-direction. The results of these corollaries put somewhat restrictive conditions on the images which satisfy them. For example, they do not allow there to be an X-translation of the given image which is still contained in the northwest quadrant of the larger grid to result in a cheaper or equal cost quadtree than the original image. Informally, the image should completely fill the smaller grid. That is, the enclosing rectangle of the image should have sides which are $2^N \times 2^N$. These conditions are reflected in the hypotheses. Suppose, for example, that each black pixel of the image is at least 1 pixel from the eastern boundary. Then, $M \geqslant 2^{N-1}$. Thus, $f(2M) + g(4m + 4W) + g(4B) \geqslant f(2^N 3) = 3(2^N - 1)$.

We would now like to extend Theorem 5.1 so that the latter embedding is arbitrary. Before we do this, however, we will need some more preliminary results.

Let VX_j be any X-translation vector of order $0 \leqslant j \leqslant N$, and let VY_k be any Y-translation vector of order $0 \leqslant k \leqslant N$. We define $QXY_{j,k}$ as the partial quadtree of a completely white image with respect to the region into which the northwest quadrant is taken under the application of VX_j followed by the application of VY_k. Also, we let $TXY_{j,k}$ be the quadtree which results after applying VX_j followed by the application of VY_k. Thus, $TXY_{j,k}$ is a completion of $QXY_{j,k}$.

We then have the following theorem and its corollaries, whose proofs may be found in [5].

Theorem 5.2: For $1 \leqslant k \leqslant j \leqslant N$, let WE and WW be the cardinalities of the largest collections of disjoint blocks of size $2^{N-j} \times 2^{N-j+1}$ which are adjacent to the eastern and western boundaries of *IMAGE*, respectively, and which consist entirely of white pixels. Let WN and WS be the cardinalities of the largest collections of disjoint blocks of size $2^{N-k+1} \times 2^{N-k}$ which are adjacent to the northern and southern boundaries of *IMAGE*, respectively, and which also consist entirely of white pixels. Define $MEW = \max(WE, WW)$ $mew = \min(WE, WW)$, $MNS = \max(WN, WS)$ and $mns = \min(WN, WS)$. Let $W1$ be the sum of the cardinality of the largest collection of disjoint blocks of size $2^{N-j} \times 2^{N-j+1}$ on the *mew* border of IMAGE and the cardinality of the largest collection of disjoint blocks of size $2^{N-j+1} \times 2^{N-j}$ on the *mns* border of IMAGE, both collections consisting entirely of white pixels. Finally, suppose $W2$ and B are the cardinalities of the largest collections of disjoint blocks of size $2^{N-j+1} \times 2^{N-j+1}$ which consist entirely of white pixels and entirely of black pixels, respectively, and which are at least 2^{N-j} pixels from the *MEW*, *mew*, and *mns* border of IMAGE and at least 2^{N-k} pixels from the *MNS* border of IMAGE.

Assume $1 \leqslant q \leqslant p \leqslant N$, $q \leqslant k$, and $p \leqslant j$. Then $f(2MWE) + f(2MNS + 2) + g(4B) + 4W1 + 4W2 + 1 < 3(2^j + 2^k + k) - 3(2^p + 2^q - q) + \sum_{i=1}^{j-1} p_{N-j+i+1}(4^i - 1)$ implies that $TX_{p,q} < TX_{j,k}$.

A similar result also holds for $1 \leqslant j \leqslant k \leqslant N$.

Corollary 5.3: Define WE and WW as the cardinalities of the largest collections of disjoint blocks of size 1×2 which are adjacent to the eastern and western boundaries of IMAGE, respectively, and which consist entirely of white pixels. Let WN and WS be the cardinalities of the largest collections of disjoint blocks of size 2×1 which are adjacent to the northern and southern boundaries of *IMAGE*, respectively, and which also consist entirely of white pixels. Define MEW, mew, MNS, and mns as in Theorem 5.2. Let $W1$ be the sum of the cardinality of the largest collection of disjoint blocks of size 1×2 on the *mew* border of *IMAGE* and the cardinality of the largest collection of disjoint blocks of size 2×1 on the *mns* border of *IMAGE*, both collections consisting entirely of white pixels. Suppose $W2$ and B are the cardinalities of the largest collections of disjoint blocks of size 2×2 which consist entirely of white pixels and entirely of black pixels, respectively, and which are at least 1 pixel from each border of *IMAGE*.

Then $f(2MWE) + g(4B) + 4W1 + 4W2 < 3(2^N - 1)$ and $f(2MNS + 2) \leqslant 3(2^{N-1} - 1) - 1$ imply that $TX_{0,0} < TX_{j,k}$, for $1 \leqslant k \leqslant j \leqslant N$. A similar result also holds for $1 \leqslant j \leqslant k \leqslant N$.

Corollary 5.4: Let WE, WW, WN, WS, MNS, mns, MEW, mew, $W1$, $W2$, and B be as in Corollary 5.3. Then $f(2MWE) + g(4B) + 4W1 + 4W2 < 2^{N-1}3$ and $f(2MNS + 2) \leqslant 3(2^{N-2} - 1)$ imply that $TX_{j,k} < TX_{j+1,k}$ for $1 \leqslant k \leqslant j \leqslant N$. A similar result also holds for $1 \leqslant j \leqslant k \leqslant N$.

VI. Conclusions

In this paper, we have discussed the optimal image size for quadtrees. We have shown that two image sizes may give us an optimal quadtree and that sometimes the larger size is better.

We next gave sufficient conditions for the smaller image size to give us cheaper quadtrees. Turning these conditions around gives some necessary conditions under which the larger image size results in a quadtree which is no more expensive than the smaller image size.

These conditions can certainly be improved, as the shape of our image was not taken into account beyond a count of various monocolored regions. A more detailed study of how shape affects the quadtree of an image should lead to more powerful results.

References

[1] N. Alexandridis and A. Klinger, "Picture decomposition, tree data-structures, and identifying directional symmetries as node combinations," *Comput. Graphics Image Processing*, vol. 8, pp. 43–77, 1978.

[2] C. R. Dyer, "Computing the Euler number of an image from its quadtree," Univ. Maryland, College Park, Comput. Sci. Rep. TR-769, May 1979.

[3] C. R. Dyer, A. Rosenfeld, and H. Samet, "Region representation: Boundary codes from quadtrees," *Commun. Ass. Comput. Mach.*, vol. 23, no. 3, pp. 171–179, 1980.

[4] W. I. Grosky and R. Jain, "Optimal quadtrees for image segments," Dep. Comput. Sci., Wayne State Univ., Detroit, MI, Tech. Rep. CSC 81-010, Sept. 1980.

[5] —, "Optimal quadtrees for image segments II," Dep. Comput. Sci., Wayne State Univ., Detroit, MI, Tech. Rep. CSC 82-007, Feb. 1982.

[6] E. Horowitz and S. Sahni, *Fundamentals of Data Structures.* Potomac, MD: Computer Science Press, 1976.

[7] G. M. Hunter, "Efficient computation and data structures for graphics," Ph.D. dissertation, Dep. Elec. Eng. and Comput. Sci., Princeton Univ., Princeton, NJ, 1978.

[8] G. M. Hunter and K. Steiglitz, "Linear transformations of pictures represented by quadtrees," *Comput. Graphics Image Processing*, vol. 10, pp. 289–296, 1979.

[9] —, "Operations on images using quadtrees," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-1, pp. 145–153, 1979.

[10] A. Klinger, "Data structures and pattern recognition," in *Proc. 1st IJCPR*, 1973, pp. 497–498.

[11] A. Klinger and C. R. Dyer, "Experiments in picture representation using regular decomposition," *Comput. Graphics Image Processing*, vol. 5, pp. 68–105, 1976.

[12] A. Klinger and M. L. Rhodes, "Organization and access of image data by areas," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-1, pp. 50–60, 1979.

[13] D. Knuth, *The Art of Computer Programming.* Reading, MA: Addison-Wesley, 1975.

[14] S. Ranade, A. Rosenfeld, and H. Samet, "Shape approximation using quadtrees," Univ. Maryland, College Park, Comput. Sci. Rep. TR-847, Dec. 1979.

[15] H. Samet, "Computing perimeters of images represented by quadtrees," Univ. Maryland, College Park, Comput. Sci. Rep. TR-755, 1979.

[16] —, "Connected component labelling using quadtrees," Univ. Maryland, College Park, Comput. Sci. Rep. TR-756, Apr. 1979.

[17] —, "A distance transform for images represented by quadtrees," Univ. Maryland, College Park, Comput. Sci. Rep. TR-780, July 1979.

[18] —, "A quadtree medial axis transform," Univ. Maryland, College Park, Comput. Sci. Rep. TR-803, Aug. 1979.

[19] —, "Region representation: Quadtree-to-raster conversion," Univ. Maryland, College, Park, Comput. Sci. Rep. TR-768, June 1979.

[20] —, "Region representation: Raster-to-quadtree conversion," Univ. Maryland, College Park, Comput. Sci. Rep. TR-766, May 1979.

[21] —, "Region representation: Quadtrees from binary arrays," *Comput. Graphics Image Processing*, vol. 13, pp. 88–93, 1980.

[22] —, "Region representation: Quadtrees from boundary codes," *Commun. Ass. Comput. Mach.*, vol. 23, no. 3, pp. 163–170, 1980.

[23] M. Shneier, "Linear-time calculations of geometric properties using quadtrees," Univ. Maryland, College Park, Comput. Sci. Rep. TR-770, May 1979.

[24] —, "A path-length distance transform for quadtrees," Univ. Maryland, College Park, Comput. Sci. Rep. TR-794, July 1979.

Application of a Multilayer Decision Tree in Computer Recognition of Chinese Characters

Y. X. GU, Q. R. WANG, AND C. Y. SUEN

***Abstract*—A multistage classifier with general tree structure has been developed to recognize a large number of Chinese characters. A simple and efficient method of classifying the characters was achieved by choosing the best feature at each stage of the tree. The features used are Walsh coefficients obtained from two profiles of a character projected onto the X-Y orthogonal axes. Some algorithms for aligning the characters were compared and one of them was adopted in this recognition scheme. A high recognition rate of about 99.5 percent was obtained in an experiment with more than 3000 different Chinese characters.**

***Index terms*—Character recognition, Chinese characters, multistage classifier, tree classifier, Walsh coefficients.**

I. Introduction

Chinese characters have a history of several thousand years and they are used by a large number of people. Research on automatic recognition of Chinese characters is a very attractive field and is a great challenge. A considerable amount of work has been done in this area since 1966 [1].

The large number of character categories and the structural complexity of character patterns are the main difficulties in Chinese character recognition. To overcome the first difficulty, several preliminary classification schemes have been proposed to reduce the number of candidates in the recognition stage and success has been reported on the recognition of up to 2000 characters [1], [2].

In a two-stage classification scheme, some orthogonal transforms may be used in the second stage to measure the similiarities between the input character and the selected candidates, the number of which generally falls between 50 and 100. In order to reduce the candidate number from several thousand to less than a hundred, some distinctive features are needed,

Manuscript received October 28, 1981; revised September 9, 1982. This work was supported by a research grant from the Kung Chung Wo Company, Ltd., Hong Kong.

The authors are with the Department of Computer Science, Concordia University, Montreal, P.Q., H3G 1M8, Canada.