

Yang, Ching-Wen, Pau-Choo Chung, and Chein-I. Chang. "Hierarchical Fast Two-Dimensional Entropic Thresholding Algorithm Using a Histogram Pyramid." *Optical Engineering* 35, no. 11 (November 1996): 3227–41. <https://doi.org/10.1117/1.601061>.

<https://doi.org/10.1117/1.601061>

©1996 Society of Photo-Optical Instrumentation Engineers (SPIE). One print or electronic copy may be made for personal use only. Systematic reproduction and distribution, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper are prohibited.

Access to this work was provided by the University of Maryland, Baltimore County (UMBC) ScholarWorks@UMBC digital repository on the Maryland Shared Open Access (MD-SOAR) platform.

**Please provide feedback**

Please support the ScholarWorks@UMBC repository by emailing [scholarworks-group@umbc.edu](mailto:scholarworks-group@umbc.edu) and telling us what having access to this work means to you and why it's important to you. Thank you.

# Hierarchical fast two-dimensional entropic thresholding algorithm using a histogram pyramid

**Ching-Wen Yang**  
**Pau-Choo Chung**

National Cheng Kung University  
Department of Electrical Engineering  
Tainan, Taiwan 70101  
E-mail: pcchung@eembox.ncku.edu.tw

**Chen-I Chang**, MEMBER SPIE  
University of Maryland Baltimore County  
Department of Computer Science  
and Electrical Engineering  
Baltimore, Maryland 21228-5398  
E-mail: cchang@umbc2.umbc.edu

**Abstract.** Entropic thresholding provides an alternative view to the design rationale of conventional thresholding. Abutaleb proposed a second-order entropic thresholding approach to improve Pun's first-order entropic thresholding by introducing a 2-D gray-level histogram to take into account the spatial correlation. Abutaleb's (1989) method was further modified by Brink (1992). However, a major drawback of Abutaleb-Brink's method is very high computational cost. Recently, Chen et al. (1994) developed a fast efficient 2-D algorithm that reduces computational complexity from  $O(L^4)$  to  $O(L^{8/3})$ , where  $L$  is the total number of gray levels. A hierarchical fast 2-D entropic thresholding algorithm using a gray-level histogram pyramid is presented that can be viewed as a generalization of Chen et al.'s algorithm. The new algorithm consists of a 2-D gray-level histogram pyramid build-up procedure expanding Abutaleb's 2-D gray-level histogram to a histogram pyramid, and a thresholding process applying a modified version of Chen et al.'s algorithm to the histogram pyramid layer by layer from top to bottom. As a result, the computational complexity of Chen et al.'s algorithm can be further reduced to the optimal complexity,  $O(L^2)$ . The experiments show that the computer time of the new algorithm is only one tenth of that required for Chen et al.'s algorithm, which is a significant saving. © 1996 Society of Photo-Optical Instrumentation Engineers.

**Subject terms:** Abutaleb's two-dimensional gray-level histogram; co-occurrence matrix; two-dimensional gray-level histogram pyramid; entropic thresholding; quantization.

Paper 19115 received Nov. 17, 1995; revised manuscript received Mar. 28, 1996; accepted for publication June 4, 1996.

## 1 Introduction

Thresholding is an initial step of image segmentation. Many thresholding techniques have been studied in the past. Of particular interest is entropic thresholding, which uses Shannon's entropy of an image histogram as a thresholding criterion to segment an image. The first work to introduce the entropic approach into thresholding was investigated by Pun<sup>1</sup> and later improved by Kapur et al.<sup>2</sup> Due to the fact that the Pun-Kapur et al. approach is based on the image histogram, which is the first-order statistic, their method can be regarded as a first-order entropic thresholding. A major drawback of first-order thresholding is that the spatial correlation between pixels is not taken into account. As a result, the performance is generally not satisfactory. Pal and Pal<sup>3</sup> extended Pun-Kapur et al. method by using a gray-level co-occurrence matrix to account for correlation between pixels. Two methods were suggested in Ref. 3, called local entropy and joint entropy, both of which were also based on maximization of Shannon's entropy. Recently, Chang et al.<sup>4</sup> adopted a different approach that uses the concept of relative entropy defined on a gray-level co-occurrence matrix to minimize the discrepancy between an original image and a thresholded image. Since the gray-level co-occurrence matrix is basically a 2-D spatial gray-

level histogram, the methods in Refs. 3 and 4 can be thought of as second-order entropic thresholding approaches.

An alternative to considering a 2-D spatial gray-level histogram is one introduced by Abutaleb,<sup>5</sup> which is formed by the 1-D gray-level histogram and the 1-D local average gray-level histogram. Abutaleb's work was the first report to propose using such a 2-D (gray-level, local average gray-level) histogram, to be called Abutaleb's histogram, to improve the first-order entropic thresholding techniques. His approach is also a second-order entropic thresholding technique. However, unlike methods in Refs. 3 and 4, which generate only one threshold, Abutaleb found a pair of thresholds, one for each dimension, and his method was further improved by Brink.<sup>6</sup> The advantage of the Abutaleb-Brink approach is that the two thresholds can be simultaneously adjusted to obtain a better image quality than methods<sup>3,4</sup> using the same threshold value for both dimensions. Nevertheless, a principal disadvantage of the Abutaleb-Brink method is the computational cost, which exponentially increases with the image size and the total number of gray levels. To alleviate this problem, Chen et al.<sup>7</sup> proposed a fast 2-D thresholding algorithm that decomposes a thresholding process into two procedures. The first-stage procedure is to quantize gray levels of an image prior to thresholding, followed by a second-stage procedure to find a pair of thresholds based on this quantized image.

Although the two obtained thresholds may not be exactly the desired optimal threshold vectors, they should be very close. How close they are depends on how fine the gray levels are to be quantized. It was proved<sup>7</sup> that  $\lceil L^{2/3} \rceil$  is the optimal number of quantization levels, where  $L$  is the total number of gray levels to be used for the image, and  $\lceil x \rceil$  is defined as the smallest integer  $\geq x$ , and the required computing time is significantly reduced.

In this paper, a hierarchical fast 2-D entropic thresholding pyramid algorithm is presented that includes Chen et al.'s 2-D fast thresholding algorithm as a special case. The proposed algorithm is basically a hierarchical thresholding process that can be implemented in multiple stages using a gray-level histogram pyramid rather than only two stages, as described in Ref. 7. It expands Abutaleb's 2-D gray-level histogram into a 2-D gray-level histogram pyramid. Each layer of the histogram pyramid represents a reduced-size/resolution version of Abutaleb's 2-D gray-level histogram, with the number of gray levels gradually reduced as the layer number increases. Each layer in the histogram pyramid represents a 2-D gray-level histogram generated by merging into one gray level vector (gray level, local average gray level) a fixed number of gray level vectors in the histogram of its next immediate lower layer. In other words, the histogram of layer  $k$  can be also thought of as a quantized version of the histogram of layer  $k-1$  with gray level vectors viewed as quantization level vectors. Based on this pyramidal histogram structure, Chen et al.'s 2-D fast thresholding algorithm can be implemented by a two-layer histogram pyramid process. The top layer (i.e., the second layer) corresponds to a 2-D gray-level histogram obtained by uniformly quantizing the original Abutaleb 2-D gray-level histogram along each dimension in the bottom layer (i.e., the first layer). The uniform quantizer to be used is one merging  $\lceil L^{1/3} \rceil$  gray levels along each dimension in the bottom layer into one gray level in the top layer with  $\lceil x \rceil$  defined as the largest integer  $\leq x$ . More generally, assume that  $q$  is the number of gray levels to be merged along each dimension in a layer of the 2-D gray-level histogram pyramid,  $q^2$  gray level vectors (i.e.,  $q$  gray levels in each dimension) will be merged into one gray level vector and the size of the histogram will be reduced by a ratio  $q^2-1$  as the layer number is increased. As a result, the computer processing time can be significantly reduced to  $O(L^2 \log_q L)$  from  $O(L^4)$  of Abutaleb's algorithm and  $O(L^{8/3})$  of Chen et al.'s algorithm, where  $\log_q L$  is the number of layers in the pyramid. This means the new proposed algorithm can further reduce the complexity,  $L^{2/3}$  from  $O(L^{8/3})$  to  $O(L^2)$ , and achieves an optimal complexity because it requires at least  $O(L^2)$  to search for two optimal thresholds independently along each dimension. It can be also shown that the optimal choice for  $q$  is 2. Experimental results demonstrate that the proposed 2-D hierarchical pyramid thresholding algorithm requires only one tenth the computing time of Chen et al.'s algorithm and 1/600 to 1/800 of that for Brink's method. This is a substantial saving.

In Sec. 2, three 2-D entropic thresholding algorithms, Abutaleb's, Brink's, and Chen et al.'s algorithms are briefly reviewed. In Sec. 3, a new hierarchical entropic thresholding pyramid algorithm is described that is made

up of two algorithms, a build-up algorithm for Abutaleb's 2-D gray-level histogram pyramid and a modified version of Chen et al.'s algorithm. In Sec. 4, experimental results based on six test images are included to show that the new algorithm is indeed very efficient and effective. In Sec. 5, the computational complexity is discussed on the basis of trade-off between various quantization schemes used in the pyramid build-up algorithm and the new algorithm is shown to achieve optimal complexity. Also discussed in Sec. 5 is a further computational saving that can be achieved by narrowing the search areas for optimal threshold vectors in Abutaleb's histogram to a smaller feasible region. Finally, a brief conclusion is given in Sec. 6.

## 2 2-D Entropic Thresholding

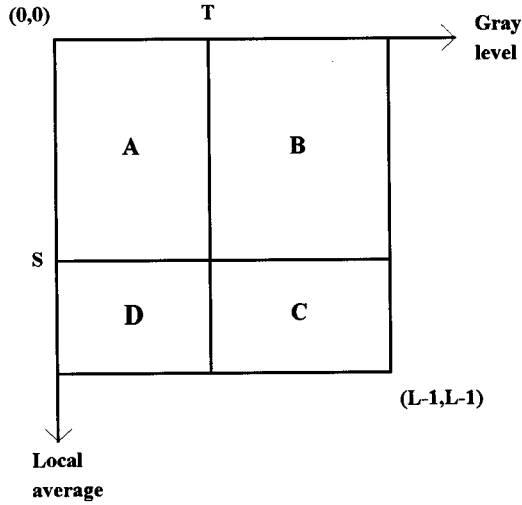
Entropy is an uncertainty measure first introduced by Shannon into information theory to describe how much information is contained in a source governed by a probability law. This concept becomes increasingly important in image processing, since an image can be interpreted as an information source with the probability law given by its image histogram. The first work applying the concept of entropy to thresholding was investigated by Pun,<sup>1</sup> who suggested selecting a threshold that maximizes the sum of entropies of the object and background classes. Although Pun's algorithm was later corrected and improved by Kapur et al.,<sup>2</sup> their results were generally not satisfactory. This is primarily due to the fact that the image histogram they considered was the first-order statistic and did not take into consideration the spatial correlation. To remedy this problem, several approaches<sup>3-5</sup> have been proposed to account for the spatial correlation between pixels.

Two types of 2-D gray-level histograms are of interest, one is the gray-level co-occurrence matrix suggested by Haralick et al.,<sup>8</sup> and the other was proposed by Abutaleb.<sup>5</sup> While the former has found a wide range of applications in texture analysis, the latter is almost entirely limited to thresholding applications. The work in Refs. 3 and 4 used the gray-level co-occurrence matrix of an image to define a 2-D gray-level histogram from which various second-order entropies can be defined. The difference between Refs. 3 and 4 is that the former maximized the second-order Shannon's entropy, while the latter minimized the relative entropy between an original image and a thresholded image.

Abutaleb's 2-D gray-level histogram is another type of 2-D gray-level histogram that differs from the gray-level co-occurrence matrix. It is formed by the Cartesian product of the original 1-D gray-level histogram and the 1-D local average gray-level histogram generated by applying a local window to each pixel of the image and calculating the average of the gray levels within the window. Since Abutaleb's 2-D gray-level histogram is the main focus in this paper, the gray-level co-occurrence matrix approach to thresholding is not discussed here (but refer to Refs. 3 and 4).

### 2.1 Abutaleb's Approach<sup>5</sup>

Suppose that a digital image  $I$  is represented by a set of  $N \times M$  pixels. Associated with a pixel at the spatial location  $(x, y)$  in image  $I$  is a gray level function  $f(x, y)$  taking values in a discrete set  $G = \{0, 1, \dots, L-1\}$ , called gray lev-



**Fig. 1** Abutaleb's 2-D gray-level histogram with threshold vector  $(T, S)$ .

els. For each pixel  $(x, y)$ , we can also consider a window with size  $n+1 \times n+1$  centered at the pixel  $(x, y)$  and calculate its local average gray level within the window by

$$g(x, y) = \frac{1}{(n+1)^2} \sum_{j=-n/2}^{n/2} \sum_{i=-n/2}^{n/2} f(x+i, y+j) \quad (1)$$

where  $n \leq N$  and  $n \leq M$ .

From Eq. (1) a pixel at  $(x, y)$  can be assigned a pair of gray levels  $(i, j)$ , where gray level  $i$  and local average gray level  $j$  are determined by  $f(x, y)$  and  $g(x, y)$  respectively. Let  $r_{ij}$  denote the frequency of occurrence of the pair  $(i, j)$  in the image  $I$ . The prior probability of the pair  $(i, j)$  is given by

$$p_{ij} = \frac{r_{ij}}{NM} \quad (2)$$

Based on Eq. (2), Abutaleb's 2-D gray-level histogram, denoted by  $W$ , can be depicted by Fig. 1 and defined by a matrix  $[W(i, j)]_{L \times L}$  with entries  $W(i, j) = r_{ij}$ , where  $i$  and  $j$  represent the horizontal and vertical axes, respectively, with  $(0, 0)$  at the left upper corner. Both gray level  $i$  and local average gray level  $j$  take values in  $G = \{0, 1, \dots, L-1\}$ . If we assume that the pair  $(T, S)$  is a threshold vector to be used for thresholding, the  $(T, S)$  divides Abutaleb's 2-D histogram into four quadrants. These quadrants can be further classified into the diagonal quadrants A and C and off-diagonal quadrants B and D, respectively, in Fig. 1. Since pixels belonging to either the object class or background class are expected to have small gray-level variations, diagonal quadrants A and C reflect local properties. On the other hand, off-diagonal quadrants B and D represent greater differences between gray levels and local mean, which reflect transitions between background and objects, thus, they are very likely to be edges. Never-

theless, this is not necessarily true in general. For instance, the diagonal of matrix in Fig. 1 generally represents perfectly homogeneous regions. As shown in Fig. 1, the diagonal line passes through quadrant B if the threshold pair  $(S, T)$  with  $S > T$  (or could be quadrant D if  $S < T$ ). In such a case, pixels represented by gray levels on the diagonal cannot be edges. Abutaleb's 2-D histogram may not do well, for example, the results in Fig. 8 in Sec. 4. However, if we allow a tolerance for small deviations from the diagonal, which may be caused by noise effects, the diagonal line can be expanded into a diagonal band so that if pixels represented by this band can be still thought of to be in a homogeneous region. A similar idea considered for spectral co-occurrence matrix<sup>9</sup> can be applied to this case. But, it is not our major concern and is not pursued here.

From Fig. 1, the probabilities of the object class and background class can be calculated by

$$P_B(T, S) = \sum_{j=0}^S \sum_{i=0}^T p_{ij}, \quad (3)$$

$$P_O(T, S) = \sum_{j=S+1}^{L-1} \sum_{i=T+1}^{L-1} p_{ij}.$$

Using Eqs. (3) as normalization factors, the normalized *a posteriori* probabilities of the object class and background class are functions of threshold vector  $(T, S)$  and defined as

$$p_B(i, j|T, S) = \frac{p_{ij}}{P_B(T, S)}, \quad (4)$$

$$p_O(i, j|T, S) = \frac{p_{ij}}{P_O(T, S)}.$$

From the *a posteriori* probabilities given by Eqs. (4) we can define the entropy for the background class, denoted by  $H_B(T, S)$  and the entropy of the object class, denoted by  $H_O(T, S)$  as follows:

$$H_B(T, S) = - \sum_{j=0}^S \sum_{i=0}^T p_B(i, j|T, S) \log p_B(i, j|T, S), \quad (5)$$

$$H_O(T, S) = - \sum_{j=S+1}^{L-1} \sum_{i=T+1}^{L-1} p_O(i, j|T, S) \log p_O(i, j|T, S).$$

Following the idea suggested in Ref. 6, Abutaleb used Eqs. (5) to define the total entropy sum given by

$$\Psi(T, S) = H_B(T, S) + H_O(T, S). \quad (6)$$

Abutaleb claimed that the best threshold vector  $(T^*, S^*)$  is the one that maximizes Eq. (6), namely,  $(T^*, S^*)$  is the solution to the following maximization problem.

$$(T^*, S^*) = \arg\{\max_{(T, S)} \Psi(T, S)\}. \quad (7)$$

## 2.2 Brink's Approach<sup>6</sup>

Instead of maximizing the total entropy sum of Eq. (6), Brink improved Abutaleb's method in Ref. 1 by proposing an alternative approach that is a max-min method. Brink first found the minimum of  $\{H_B(T,S), H_O(T,S)\}$  for each given threshold vector  $(T,S)$ , then maximized the minimum of  $\{H_B(T,S), H_O(T,S)\}$  over all possible threshold pairs  $(T,S)$ , namely,

$$(T^*, S^*) = \arg \left\{ \max_{\substack{T=0,1,\dots,L-1 \\ S=0,1,\dots,L-1}} [\min\{H_B(T,S), H_O(T,S)\}] \right\}. \quad (8)$$

Brink showed that in most experiments, the max-min method performed better than Abutaleb's maximization of the total entropy sum.

## 2.3 Chen et al.'s Approach<sup>7</sup>

Since Brink's method requires brute force to solve Eq. (8) by exhaustively searching for all possible pairs of  $(T,S)$  with  $T$  and  $S$  ranging from 0 to  $L-1$ , the computational cost is very expensive and the complexity can be as high as  $O(L^4)$ . To alleviate this problem, Chen et al.<sup>7</sup> proposed a fast algorithm to find a solution to Eq. (8). Rather than directly finding an optimal threshold vector  $(T^*, S^*)$ , their idea was to decompose Brink's thresholding procedure into two processes. The first-stage process is to quantize the original image using a small number of gray levels from which a set of candidate threshold vectors  $C$ , called local optimal threshold vectors, can be generated using Eq. (8). This quantization procedure narrows the search area for an optimal threshold vector  $(T^*, S^*)$  to a candidate set of local optimal threshold vectors. This is followed by a second stage process in which Brink's method is applied again to this candidate set of local optimal threshold vectors to obtain a desired optimal threshold vector. It was shown<sup>7</sup> that the computer processing time required for Chen et al.'s fast algorithm was  $O(L^{8/3})$ , which is a significant saving in storage and time compared to  $O(L^4)$ . For instance, the saving could be reduced from 1 h to 1 min in some experiments conducted in this paper and is calculated based on total computing time and  $L = 256$ .

## 3 Hierarchical 2-D Entropic Thresholding Pyramid Algorithm

Despite the saving provided by Chen et al.'s algorithm, it is still possible to improve the algorithm in terms of computing time and memory space. In this section, we present a hierarchical pyramid algorithm that can further reduce the computational complexity of Chen et al.'s algorithm. The proposed algorithm is based on the idea of presenting Abutaleb's 2-D gray-level histogram in a pyramidal structure with each layer being a reduced-size/resolution Abutaleb 2-D gray-level histogram, and then applying a modified version of Chen et al.'s algorithm to each layer of the pyramid from top to bottom. In this way, the algorithm is carried out by two procedures, a preprocessing step to build up Abutaleb's 2-D gray-level histogram pyramid, and a modified Chen et al. 2-D thresholding processing using the generated histogram pyramid.

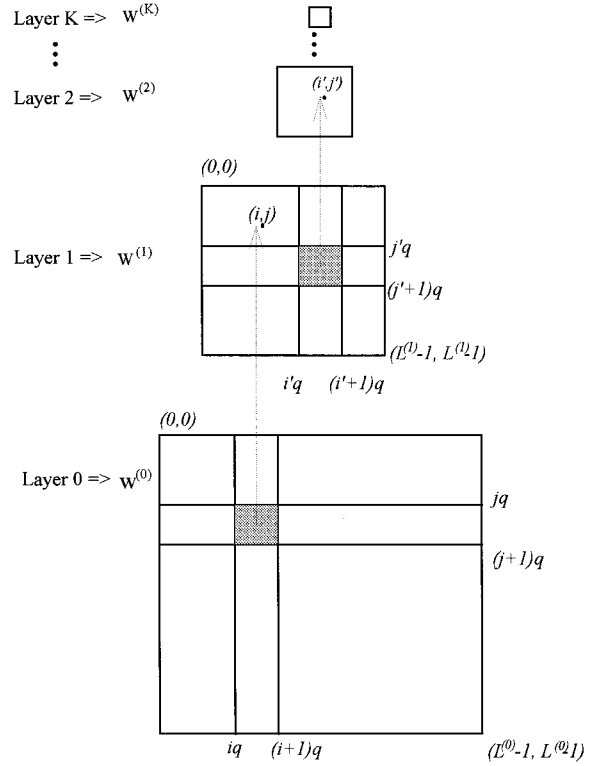


Fig. 2 Construction of Abutaleb's 2-D gray-level histogram pyramid.

### 3.1 Algorithm for Constructing Abutaleb's 2-D Gray-Level Histogram Pyramid

Let  $W^{(0)}$  denote Abutaleb's 2-D gray-level histogram of the original image obtained from the 1-D gray-level histogram and 1-D local average gray-level histogram given by Eq. (1), where  $i$  and  $j$  represent the 1-D gray-level histogram coordinate and local average gray-level histogram coordinate, respectively. Assume that  $q > 1$  is a fixed positive integer that is the number of gray levels to be merged as a layer moves up one its next higher layer. An Abutaleb 2-D gray-level histogram pyramid can be constructed as follows (see Fig. 2).

1. Calculate the local average gray level  $g(x,y)$  for each pixel over an  $n \times n$  square window, as shown in Eq. (1).
2. Layer 0: Construct the Abutaleb's 2-D gray-level histogram  $[W^{(0)}(i,j)]_{L^{(0)} \times L^{(0)}}$  for the original image. Let  $G_1^{(0)} = G = \{0, 1, \dots, L^{(0)} - 1\}$  be the range for gray level  $i$  and  $G_2^{(0)} = G = \{0, 1, \dots, L^{(0)} - 1\}$  be the range for local average gray level  $j$ , where  $L^{(0)} = L$ .
3. Layer  $k > 0$ : Construct a reduced-size/resolution Abutaleb 2-D gray-level histogram  $W^{(k)}$  at layer  $k$  based on the following equation:

$$r_{ij}^{(k)} = W^{(k)}(i,j) = \sum_{m=jq+1}^{(j+1)q} \sum_{l=iq+1}^{(i+1)q} r_{ml}^{(k-1)}, \quad (i,j) \in G_1^{(k)} \times G_2^{(k)}, \quad (9)$$

$$(m,l) \in G_1^{(k-1)} \times G_2^{(k-1)},$$

where  $r_{ij}^{(k)}$  is the frequency of occurrence of the gray level pair  $(i, j)$  in  $W^{(k)}$ ; and  $G_1^{(k)}$  and  $G_2^{(k)}$  are the sets of gray levels and local average gray levels in layer  $k$ , respectively, with the size equal to  $L^{(k)} = L/q^k$ . This gives the size of  $W^{(k)}$  to be  $L^{(k)} \times L^{(k)}$ .

4. Increase  $k$  by 1 and repeat the above process until  $L^{(k+1)} < q$ . That is, if  $K+1$  is the number of layers in the pyramid,  $L^{(K+1)} < q$ .

### 3.2 Algorithm for Finding Threshold Vectors from Abutaleb's 2-D Histogram Pyramid

Basically, the following algorithm is a modified version of Chen et al.'s algorithm. For any two successive layers  $k-1$  and  $k$  in Abutaleb's 2-D gray-level histogram pyramid, we can treat the histogram of layer  $k-1$  in Abutaleb's 2-D gray-level histogram pyramid as the original image's gray-level histogram and the histogram of layer  $k$  as that of the quantized image. As a result of this interpretation, Chen et al.'s algorithm can be viewed as a special version of the proposed hierarchical pyramid algorithm using a two-layer Abutaleb 2-D gray-level histogram pyramid.

1. Initialization: Let  $K+1$  be the total number of layers in the pyramid and  $W^{(K)}$  be the top layer of Abutaleb's 2-D histogram pyramid. The algorithm is initialized by letting the candidate set, i.e., the set of local optimal threshold vectors of layer  $K$ ,  $C^K$  equal to  $G_1^{(K)} \times G_2^{(K)}$ .
2. For any vector  $(T^{(k)}, S^{(k)}) \in C^{(k)}$  in layer  $k < K$ , first calculate the *a posteriori* probabilities

$$P_B^{(k)}(T^{(k)}, S^{(k)}) = \sum_{i \leq S^{(k)}} \sum_{j \leq T^{(k)}} p_{ij}^{(k)}, \quad i \in G_1^{(k)}, j \in G_2^{(k)},$$

$$P_O^{(k)}(T^{(k)}, S^{(k)}) = \sum_{i > S^{(k)}} \sum_{j > T^{(k)}} p_{ij}^{(k)}, \quad i \in G_1^{(k)}, j \in G_2^{(k)},$$

where  $p_{ij}^{(k)}$  is the probability of  $r_{ij}^{(k)}$  by normalization.

3. Find the probabilities of the object class and background class:

$$P_B^{(k)}(i, j | T^{(k)}, S^{(k)}) = \frac{p_{ij}^{(k)}}{P_B^{(k)}(T^{(k)}, S^{(k)})},$$

$$P_O^{(k)}(i, j | T^{(k)}, S^{(k)}) = \frac{p_{ij}^{(k)}}{P_O^{(k)}(T^{(k)}, S^{(k)})}. \quad (10)$$

4. Find the entropies of object class and background class in  $W^{(k)}$ :

$$H_B^{(k)}(T^{(k)}, S^{(k)}) = - \sum_{i \leq S^{(k)}} \sum_{j \leq T^{(k)}} p_B^{(k)}(i, j | T^{(k)}, S^{(k)}) \log p_B^{(k)}(i, j | T^{(k)}, S^{(k)}), \quad i \in G_1^{(k)}, j \in G_2^{(k)},$$

$$H_O^{(k)}(T^{(k)}, S^{(k)}) = - \sum_{i > S^{(k)}} \sum_{j > T^{(k)}} p_O^{(k)}(i, j | T^{(k)}, S^{(k)}) \log p_O^{(k)}(i, j | T^{(k)}, S^{(k)}), \quad i \in G_1^{(k)}, j \in G_2^{(k)}. \quad (11)$$

5. Find the optimal threshold vector for layer  $k$  using Brink's method, i.e.,

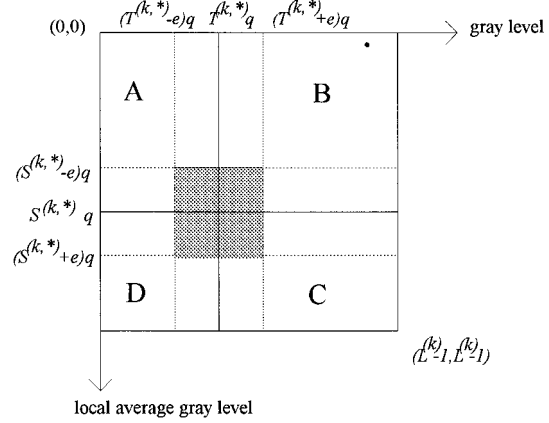


Fig. 3 Candidate set of threshold vectors for layer  $k-1$  in the histogram pyramid.

$$(T^{(k,*)}, S^{(k,*)}) = \arg \left\{ \max_{(T^{(k)}, S^{(k)}) \in G_1^{(k)} \times G_2^{(k)}} \left[ \min_{\alpha \in \{B, O\}} \{H_a^{(k)}(T^{(k)}, S^{(k)})\} \right] \right\}. \quad (12)$$

6. Find  $C^{(k-1)}$  (shown in Fig. 3), the set of candidate threshold vectors for layer  $k-1$  resulting from  $(T^{(k,*)}, S^{(k,*)})$  given by Eq. (12) in step 5:

$$C^{(k-1)} = \{(T^{(k-1)}, S^{(k-1)}) | (T^{(k,*)} - e)q \leq T^{(k-1)} \leq (T^{(k,*)} + e)q, (S^{(k,*)} - e)q \leq S^{(k-1)} \leq (S^{(k,*)} + e)q\}, \quad (13)$$

where  $e$  is the tolerance of quantization error.

7. Reduce  $k$  by 1 and go to step 2 until  $k < 0$ .

## 4 Experimental Results

In this section, results of experiments conducted to make a comparative study of various thresholding algorithms described in this paper are presented. In addition, the advantages that the proposed algorithm can offer in terms of time and memory saving and an extension into three-level thresholding are also discussed. Six images were used for comparison (shown in Figs. 4 to 9), which were  $256 \times 256$  pixels, with gray level values of  $G = \{0, 1, \dots, 255\}$  and  $L = 256$ . The window size used to generate local average gray levels was  $3 \times 3$ . The number of pixels merged for the bins in each layer,  $q$  was chosen to be 2 and the tolerance of quantization error  $e = 1$ . The total number of layers constructed in Abutaleb's 2-D histogram pyramids was 8, i.e.,  $K = 7$ . All images were run on a Sun Sparc 20 workstation with OS version 5.4 and 32M memory using C language. The histogram size of two successive layers was reduced by a factor of 4 as the layer was incremented. More specifically, for each  $k$ , the histogram size of layer  $k$  is only a quarter of that of layer  $k-1$ . Four methods were studied for computational complexity: Brink's method, Chen et al.'s algorithm, diagonal search, and the proposed hierarchical pyramid algorithm. The diagonal search is a special version of Brink's method that lim-

**Table 1** Comparison of computing time (unit in seconds) between the hierarchical pyramid algorithm and the other three algorithms.

Images	Algorithms			
	Brink's Method	Chen et al.'s Algorithm	Diagonal Search	Pyramid Algorithm
"Peppers"	3407	44	21	4
"Text1"	2258	34	13	4
"Text2"	2184	35	13	4
"Rose"	2804	37	17	3
"Lena"	3310	43	21	4
"F16jet"	2418	31	15	3

its the optimal threshold vector ( $T^*, S^*$ ) to the diagonal line of Abutaleb's 2-D gray-level histogram ( $T^*, T^*$ ), i.e.,  $T^* = S^*$ . As expected, the computing time of the diagonal search is reduced at least one order lower than that required for ( $T^*, S^*$ ) by Chen et al.'s algorithm, since only 256 gray levels are compared for the diagonal search instead of  $256 \times 256$  gray level pairs required for Brink's algorithm. Experiments using the diagonal search were included in our comparative study to demonstrate the efficiency of the hierarchical pyramid algorithm. The results tabulated in Tables 1 and 2 show the comparison in terms of computing time between the hierarchical pyramid algorithm and the other three algorithms. The new method requires only 1/10 the time required by Chen et al.'s algorithm, approximately 1/5 to 1/4 of that for the diagonal search, and 1/800 to 1/600 for Brink's max-min method.

Six images were used for experiments: Fig. 4 ("Peppers"), Figs. 5 and 6 ("Text1" and "Text2"), Fig. 7 ("Rose"), Fig. 8 ("F-16 jet"), and Fig. 9 ("Lena"). Parts (a) of all figures are original images, parts (b) are three-level thresholded images obtained by Chen et al.'s algorithm, parts (c) are three-level thresholded images obtained by the new hierarchical pyramid algorithm, parts (d) are three-level thresholded images obtained by Brink's method, and parts (e) are three-level thresholded images obtained by the diagonal search. All the three-level thresholded images are generated by setting the gray levels in quadrant A to 0, quadrant C to 255, and quadrants B and D (i.e., corresponding to edges and noise) to 127, respectively. Thresh-

**Table 2** Comparison of threshold vectors between the hierarchical pyramid algorithm and the other three algorithms.

Images	Algorithms			
	Brink's Method	Chen et al.'s Algorithm	Diagonal Search	Pyramid Algorithm
"Peppers"	(109,110)	(109,110)	(110,110)	(119,79)
"Text1"	(53,51)	(53,51)	(52,52)	(39,53)
"Text2"	(40,37)	(40,37)	(39,39)	(15,27)
"Rose"	(183,184)	(175,167)	(184,184)	(157,127)
"Lena"	(107,94)	(107,94)	(103,103)	(93,106)
"F16jet"	(99,112)	(75,119)	(108,108)	(68,111)

\*Threshold vectors=(gray level, local average gray level).

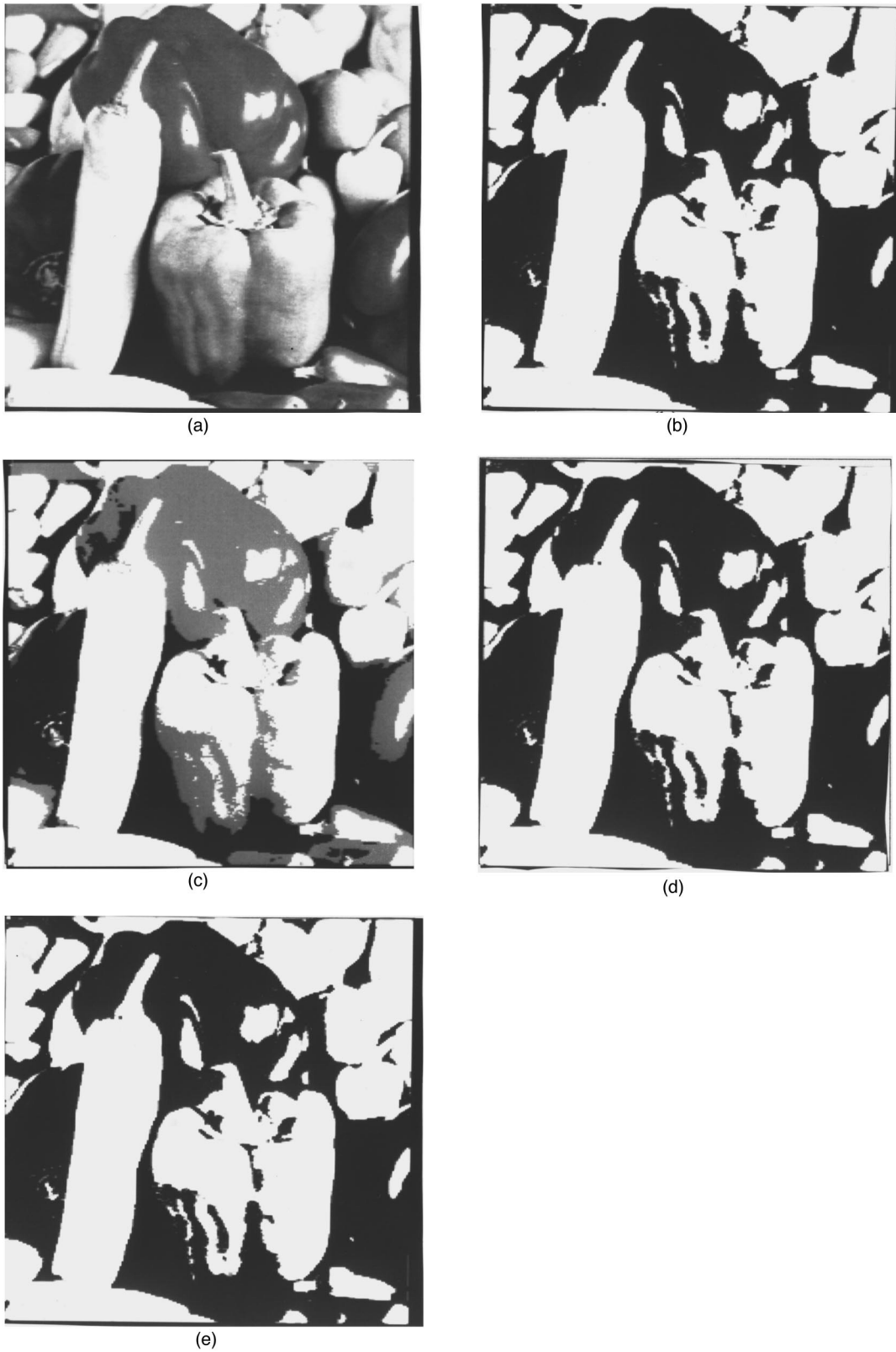
old vectors generated by these four methods are tabulated in Table 2. As shown in the table, the threshold vectors produced by the new algorithm are very different from those computed by the other three methods (which differ only slightly). This is indeed an advantage of the new algorithm over the other three methods, as demonstrated in Figs. 4 to 7, where the images thresholded by the new algorithm are of better quality than the other thresholded images. In images of Figs. 4(b), 4(d), and 4(e) the front pepper is classified as foreground and the other two peppers are classified as background since they correspond to quadrant C and quadrant A, respectively. This is not the case for the image of Fig. 4(c), where the rear pepper emerges from the background and appears in the foreground as an object. This is attributed to the fact that the rear pepper is identified by quadrants B and D. From Table 2, we can see the threshold vector (119,79) generated by the new pyramid algorithm is far different from (109,110) by the other three methods. This shows that by allowing a certain deviation from the diagonal line in Abutaleb's 2-D histogram, a better thresholded image can be produced. Similar evidence is also demonstrated in Figs. 5 to 7. In particular, for the "Text2" image, Fig. 6(c) is obviously much better than Figs. 6(b), 6(d), or 6(e). In Fig. 7, although all thresholded images miss the rose, Fig. 7(c) picks up more portions of the leaves of the rose than Figs. 7(b), 7(d), or 7(e). For the images in Figs. 8 and 9, all thresholded images produced by these four methods look very much alike with no significant differences. Note that the six images included in this paper are only representative image among many images examined in our laboratory. Based on our experiments, the new pyramid algorithm generally performs better than the other three methods. Most impressively, the computational saving is enormous.

## 5 Discussion of Computational Complexity

Recall that in step 3 of the histogram pyramid build-up algorithm,  $q$  gray levels along each dimension are merged into one gray level by averaging. In general, this step can be replaced with any 2-D vector quantizer to achieve a better quantization at the expense of complexity.<sup>10</sup>

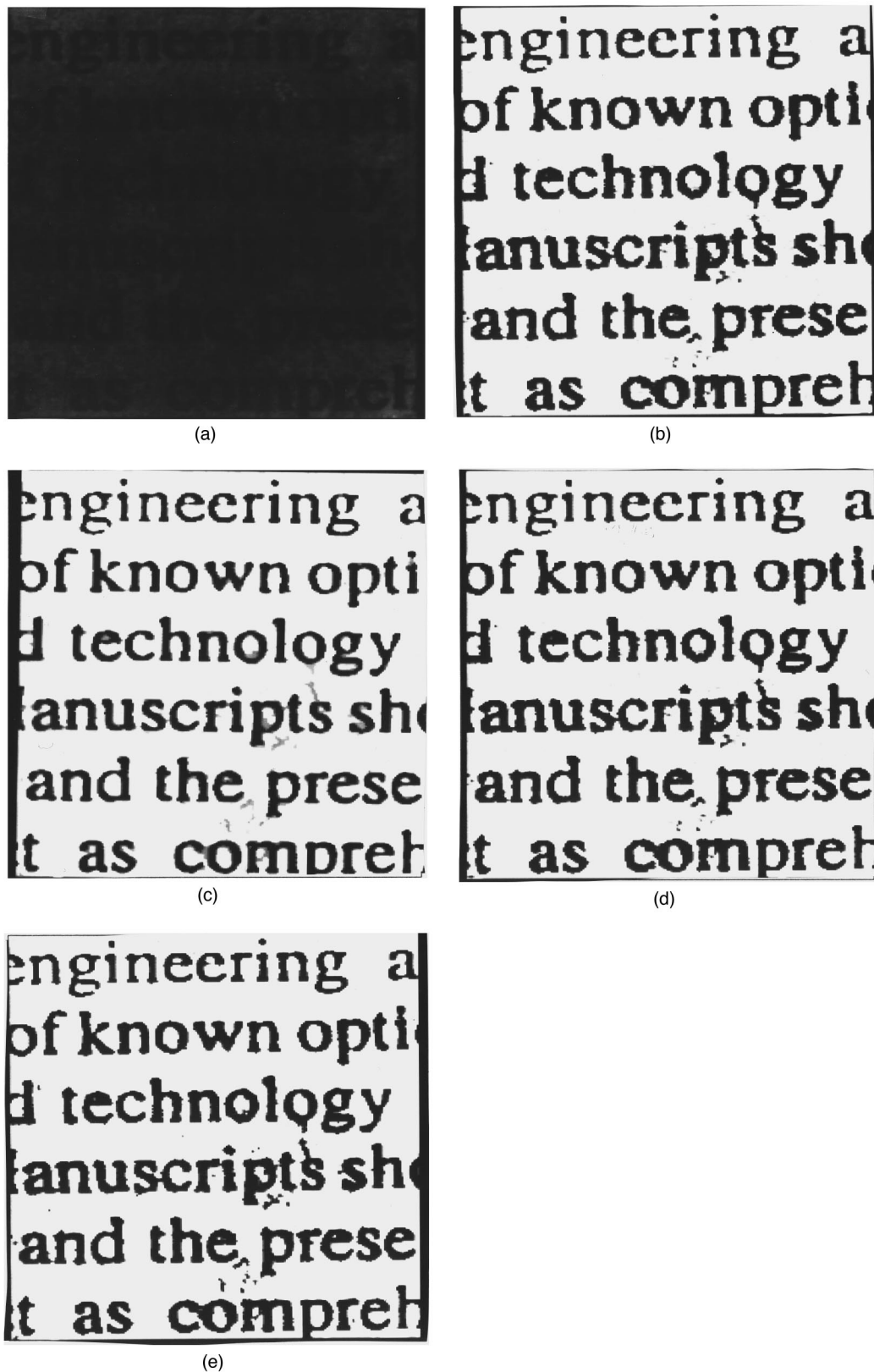
### 5.1 Complexity on Quantization

As mentioned previously, Chen et al.'s algorithm is a special version of the new hierarchical pyramid algorithm because the algorithm can be implemented as a two-layer pyramid algorithm in which the histogram of the top layer in the pyramid is obtained by a 2-D uniform vector quantizer using a quantization threshold square  $\lfloor L^{1/3} \rfloor \times \lfloor L^{1/3} \rfloor$  to quantize the original Abutaleb 2-D gray-level histogram in the bottom layer. However, the threshold vectors found by the uniform quantization are generally suboptimal. Of course, this can be further improved by designing an optimal vector quantizer<sup>11</sup> instead of the uniform vector quantizer, but it may not be worthwhile due to complexity of finding global extrema at tremendous computational cost. The averaging used in step 3 of the histogram pyramid construction is probably the simplest quantization scheme to achieve the best compromise between computational complexity and implementation. This is shown by the fact that the computer processing time required for the new al-

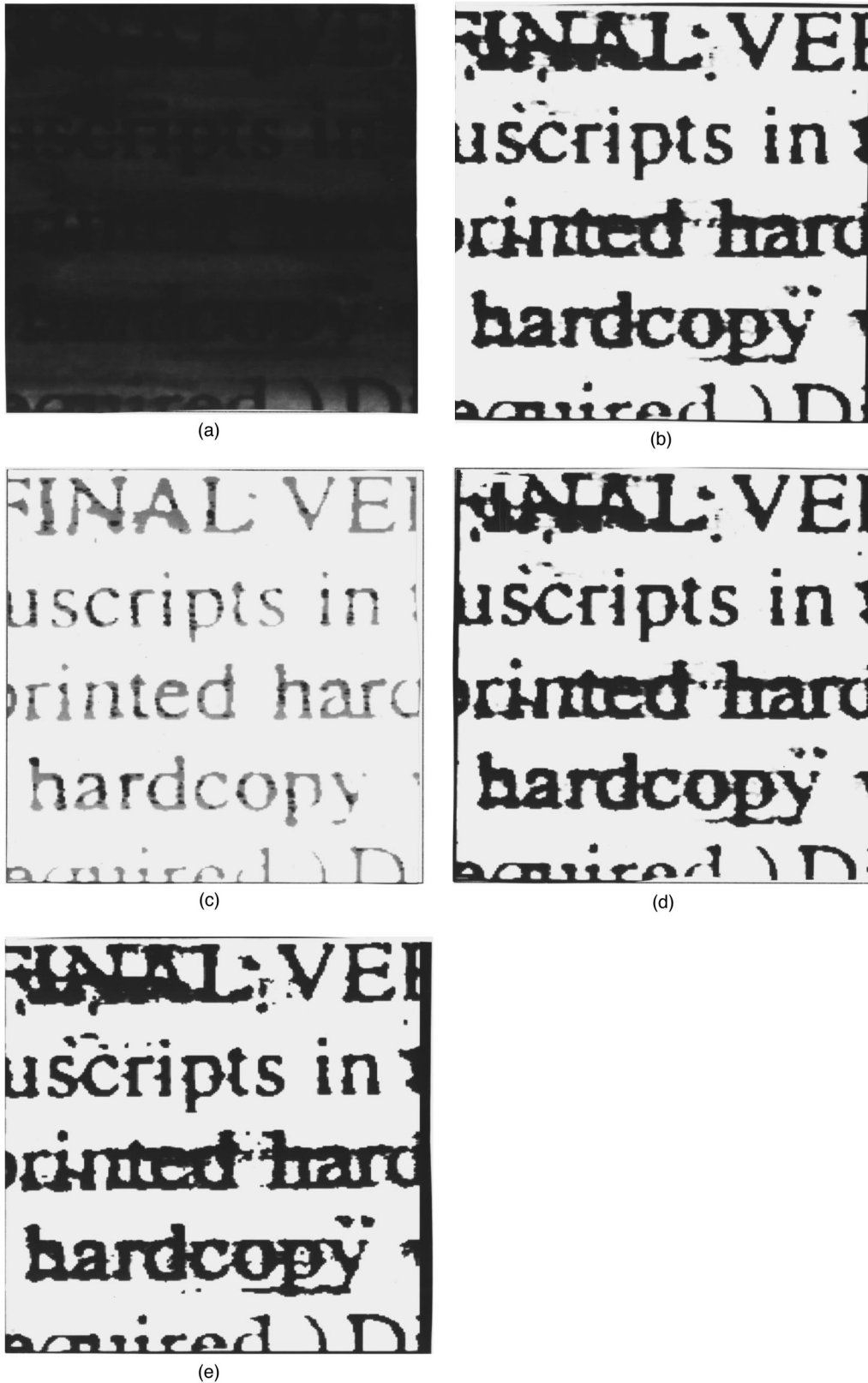


**Fig. 4** (a) Original image of "peppers," (b) three-level thresholded image obtained by Chen et al.'s algorithm, (c) three-level thresholded image obtained by the new hierarchical pyramid algorithm, (d) three-level thresholded image obtained by Brink's method, and (e) three-level thresholded image obtained by diagonal search.

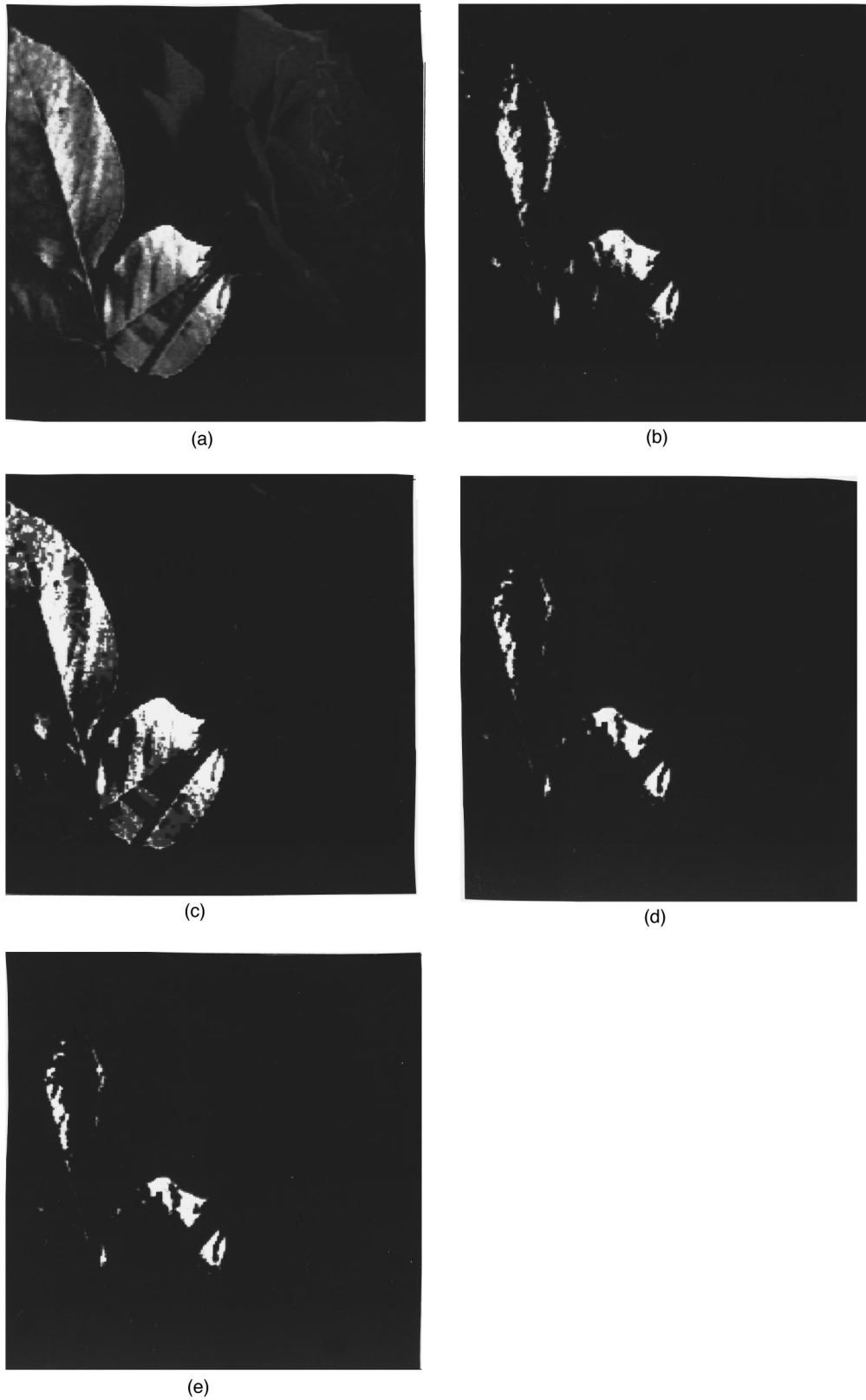




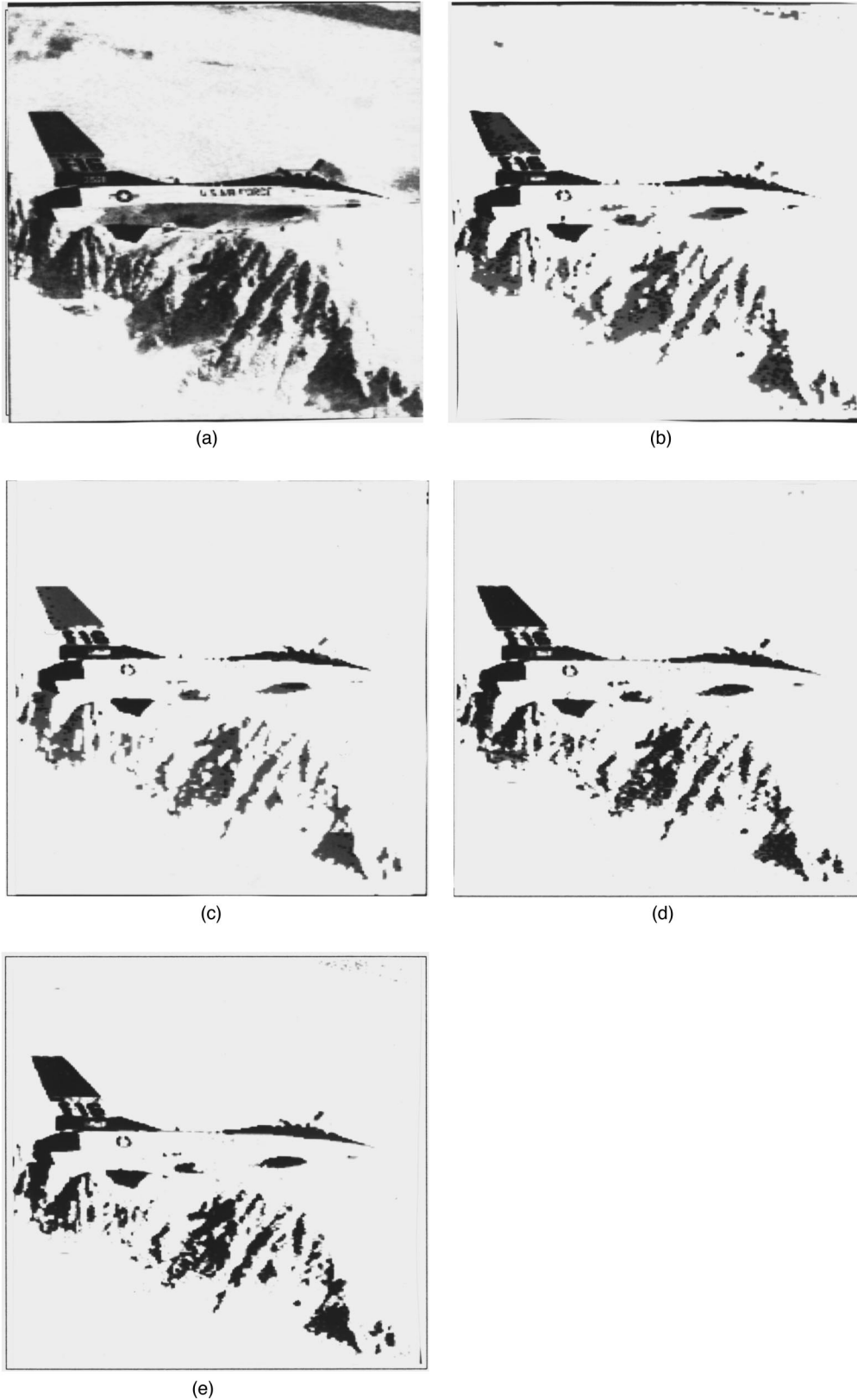
**Fig. 5** (a) Original image of "Text1," (b) three-level thresholded image obtained by Chen et al.'s algorithm, (c) three-level thresholded image obtained by the new hierarchical pyramid algorithm, (d) three-level thresholded image obtained by Brink's method, and (e) three-level thresholded image obtained by diagonal search.



**Fig. 6** (a) Original image of "Text2," (b) three-level thresholded image obtained by Chen et al.'s algorithm, (c) three-level thresholded image obtained by the new hierarchical pyramid algorithm, (d) three-level thresholded image obtained by Brink's method, and (e) three-level thresholded image obtained by diagonal search.



**Fig. 7** (a) Original image of "Rose," (b) three-level thresholded image obtained by Chen et al.'s algorithm, (c) three-level thresholded image obtained by the new hierarchical pyramid algorithm, (d) three-level thresholded image obtained by Brink's method, and (e) three-level thresholded image obtained by diagonal search.



**Fig. 8** (a) Original image of "F-16jet," (b) three-level thresholded image obtained by Chen et al.'s algorithm, (c) three-level thresholded image obtained by the new hierarchical pyramid algorithm, (d) three-level thresholded image obtained by Brink's method, and (e) three-level thresholded image obtained by diagonal search.



**Fig. 9** (a) Original image of "Lena," (b) three-level thresholded image obtained by Chen et al.'s algorithm, (c) three-level thresholded image obtained by the new hierarchical pyramid algorithm, (d) three-level thresholded image obtained by Brink's method, and (e) three-level thresholded image obtained by diagonal search.

gorithm is  $O(L^2 \log_q L)$  and  $O(L^{8/3})$  for Chen et al.'s algorithm, both of which are reduced from  $O(L^4)$  by simple quantization techniques. Note that since the averaging and uniform quantizations are simple arithmetic operations, their computing times should be negligible and will not affect the computational complexity to be considered here. On the other hand, this is not true if an optimal vector quantizer is used in each layer. This is because the computing time of generating optimal vector quantizers using the Linde-Buzo-Gray (LBG) algorithm<sup>11</sup> cannot be neglected and, in the worse case, may even dominate the total processing time.

## 5.2 Comparison between Chen's et al.'s Algorithm and the New Algorithm

In Chen et al.'s algorithm, it was proved that the optimal number of the quantized gray levels to be used in the uniform quantizer was  $\lceil L^{2/3} \rceil$ , which resulted in a complexity of  $O(L^{8/3})$  because  $\lceil L^{1/3} \rceil \times \lceil L^{1/3} \rceil$  gray-level vectors were merged into one gray-level vector. Similarly,  $\log_q L$  in the complexity of  $O(L^2 \log_q L)$  is the number of layers in the pyramid that is the result of averaging  $q^2$  gray-level vectors. In the Appendix, we prove that the computational complexity of the new algorithm is  $O(L^2 \log_q L) = O(L^2)$ , which means,  $\log_q L$  does not really contribute any complexity and is absorbed in  $O(L^2)$ . On the other hand, it also requires at least  $L^2$  comparisons to search for an optimal threshold vector  $(T^*, S^*)$ , because each dimension of Abutaleb's 2-D gray-level histogram has  $L$  gray levels, and is independent of the other dimensions. This implies a surprising result. Despite the fact that the averaging is not an optimal quantization method, the new hierarchical pyramid algorithm actually achieves the optimal complexity  $O(L^2)$ . Furthermore, the optimal value of  $q$  required for the new algorithm is also proved to be 2 in the Appendix.

One disadvantage of this new algorithm is that the final threshold vector generated is a suboptimal solution. It is true that a better or an optimal quantization scheme to find the desired optimal threshold vectors can be used. In this case, each layer in the histogram pyramid needs a different optimal quantizer to find the optimal threshold vector, at the expense of significantly increased computational cost. Nevertheless, according to our experiments (Figs. 4 to 9), the images thresholded by optimal threshold vectors are about the same as those thresholded by the new algorithm and the improvement in image quality is negligible. As described in Subsec. 5.1, such a trade-off is computationally too expensive and not worth the effort.

## 5.3 Comparison between Diagonal Search and New Algorithm

It is interesting to note from Table 1 that the computing time of the new algorithm was approximately about 1/5 of that of the diagonal search, despite  $256 \times 256$  gray level vectors needed to be searched for the former and only 256 gray levels for the latter. The diagonal search must examine all 256 gray levels to find an optimal threshold value, while the new algorithm takes advantages of the pyramidal structure of the histogram, and only a certain set of local optimal threshold vectors, called the candidate set, is processed for

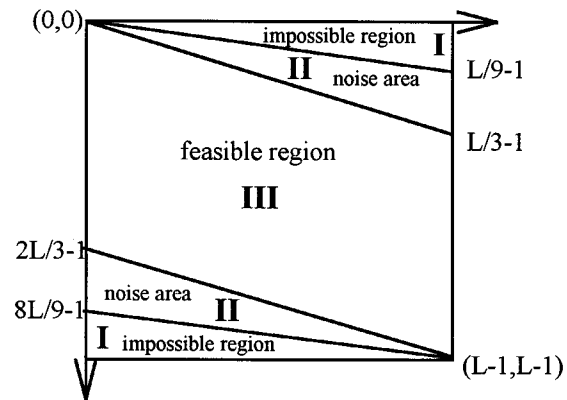


Fig. 10 Various regions for threshold vectors.

each layer. As a result, the computational load of finding an optimal threshold vector among  $256 \times 256$  gray level vectors is distributed over the candidate sets in all layers, each of which has relatively small size. On the other hand, the number of gray level vectors in each layer of the pyramid as well as the candidate set are exponentially decreased as the layer is increased. Consequently, the increase of overhead for overall computing time is still limited. The computing time shown in Table 1 includes the time for constructing histogram pyramids (about 1 s) and the time for finding local optimal threshold vectors for all layers. It is clear that all six test images require nearly the same computing time, ranging from 3 to 4 s. For instance, for the "Lena" image, the new algorithm requires about 2 s to construct the histogram pyramid and about 2 s to find the desired threshold vector.

## 5.4 Feasible Region for Searching Optimal Threshold Vectors

Another computational saving can be achieved by eliminating unnecessary (gray level, local average gray level) pairs in Abutaleb's 2-D gray-level histogram prior to thresholding. Since the local average gray level is calculated based on a  $3 \times 3$  window, some regions in Abutaleb's 2-D gray-level histogram will never be chosen for threshold vectors. For instance, if a pixel has gray level  $i$ , the local average gray level obtained from Eq. (1) must be either  $< i/9$  or  $\geq (8L+i)/9$ , as shown in Fig. 10, where the two regions labeled by I are impossible regions to be chosen for threshold vectors. The right upper triangle area, obtained by the line connecting  $((0,0), (L-1, L-1))$ , contains gray-level pairs with the local average gray levels  $< i/9$ . Similarly, the left lower triangle area obtained by the line connecting  $((8L/9-1, 0), (L-1, L-1))$ , contains gray-level pairs with the local average gray levels  $\geq (8L+i)/9$ . The regions labeled by II in Fig. 10 are designated as noise areas. They are obtained by assuming the surrounding pixels within a  $3 \times 3$  window with gray levels  $\leq 1/4$  of that of the central pixel in the window. In this case, the local average gray levels will be in the range of either  $[i/9, L/3]$  or  $[(2L+i)/3, 8L/9]$ , where  $i$  is the gray level of the central pixel. For instance, if the central pixel has gray level 255

and the gray levels of the surrounding pixels are less than 64, i.e.,  $1/4$  of 255, the corresponding local average gray level of the pixel will be less than  $255/3$ , which lies in the region II. As a result, the pixel will be considered to be noise. This will further reduce the searching area for the candidate threshold vectors to the region indicated by III in Fig. 10. Consequently, the feasible area of finding a desired threshold vector can be limited to region III without wasting efforts in regions I and II. Note that the noise areas can be also adjusted based on practical needs by changing the gray-level ratio of the central pixel to the surrounding pixels such as from  $1/4$  to  $1/3$  or  $1/5$ .

## 6 Conclusion

Second-order entropic thresholding has been proven useful by taking advantage of spatial correlation. However there is a trade-off of exponential growth of computing time. Although a fast 2-D algorithm recently proposed by Chen et al. can reduce the computational complexity to a certain extent, the computing cost is still high. In this paper, Chen et al.'s algorithm was improved by introducing a pyramidal structure into Abutaleb's 2-D gray-level histogram. The proposed new hierarchical pyramid algorithm consists of an algorithm to build a bottom-up histogram pyramid, and a modified Chen et al. algorithm, implemented iteratively using histograms from the top layer to the bottom layer of the pyramid. By doing so, the computational load is distributed over different layers, which saves storage and computing time. This new algorithm has been shown to achieve optimal complexity  $O(L^2)$ . The optimal number of gray-level vectors to be averaged in each layer in the pyramid is also shown to be 4, i.e., two for each dimension. Experiments demonstrate that the time required for the new algorithm is substantially low compared to that of Chen et al.'s algorithm by a factor of  $1/10$ .

## Appendix: Proofs of Optimal Complexity and Optimal $q$

In the following, we derive and prove that the computational complexity of the new algorithm is  $O(L^2)$ , and the optimal choice for  $q$  is 2.

From Figs. 2 and 3, we obtain

$$\text{time}(q) = q^2 + (2eq)^2 q^2 + (2eq)^2 q^4 + \cdots + (2eq)^2 q^{2i} + \cdots + (2eq)^2 L^2$$

and

$$\begin{aligned} \text{time}(q) - q^2 + (2eq)^2 &= (2eq)^2 + (2eq)^2 q^2 + (2eq)^2 q^4 \\ &+ \cdots + (2eq)^2 q^{2i} + \cdots \\ &+ (2eq)^2 L^2 = \sum_{i=0}^{\log_q L} (2eq)^2 q^{2i} \\ &= (2eq)^2 \frac{q^{2(\log_q L + 1)} - 1}{q^2 - 1} \\ &= (2eq)^2 \frac{L^2 q^2 - 1}{q^2 - 1}, \end{aligned} \quad (14)$$

which results in

$$\text{time}(q) = q^2 - (2eq)^2 + (2eq)^2 \frac{L^2 q^2 - 1}{q^2 - 1}. \quad (15)$$

Thus, the computational complexity is  $O(\text{time}(q)) = O(L^2)$ .

To find optimal  $q^*$ , let  $w = q^2$ , differentiate Eq. (15) with respect to  $w$ , and set it to zero:

$$\begin{aligned} \left. \frac{d \text{time}(q)}{dw} \right|_{w^*} &= 0 \Rightarrow 1 - 4e^2 + 4e^2 \frac{L^2 w^* - 1}{w^* - 1} + (4e^2 w^*) \\ &\times \left[ \frac{L^2(w^* - 1) - (L^2 w^* - 1)}{(w^* - 1)^2} \right] = 0 \\ &\Rightarrow (w^* - 1)^2 (1 - 4e^2) + 4e^2 (w^* - 1) \\ &\times (L^2 w^* - 1) + (4e^2 w^*) [L^2 (w^* - 1) \\ &- (L^2 w^* - 1)] = 0 \\ &\Rightarrow (1 - 4e^2 + 4e^2 L^2) (w^*)^2 \\ &- 2(1 - 4e^2 + 4e^2 L^2) (w^*) + 1 = 0. \end{aligned} \quad (16)$$

Solving the quadratic equation in  $w^*$  given by Eq. (16) yields

$$w^* = 1 \pm \left( 1 - \frac{1}{1 - 4e^2 + 4e^2 L^2} \right)^{1/2}. \quad (17)$$

To see whether  $w^*$ , given by Eq. (17), is a minimum point, differentiate Eq. (15) twice or Eq. (16) one more time and obtain

$$\begin{aligned} \left. \frac{d^2 \text{time}(q)}{dw^2} \right|_{w^*} &= \frac{d}{dw} \{ [w^2 - 1] [(1 - 4e^2 + 4e^2 L^2) w^2 \\ &- 2(1 - 4e^2 + 4e^2 L^2) w^* + 1] \} \Big|_{w^*} \\ &= 2w^* [(1 - 4e^2 + 4e^2 L^2) (w^*)^2 \\ &- 2(1 - 4e^2 + 4e^2 L^2) (w^*) + 1] \\ &+ [(w^*)^2 - 1] [2(1 - 4e^2 \\ &+ 4e^2 L^2) w^* - 2(1 - 4e^2 + 4e^2 L^2)] \quad (18) \\ &= [(w^*)^2 - 1] [2(1 - 4e^2 + 4e^2 L^2) w^* \\ &- 2(1 - 4e^2 + 4e^2 L^2)] \\ &= [(w^*)^2 - 1] [2(1 - 4e^2 + 4e^2 L^2) (w^* \\ &- 1)]. \end{aligned} \quad (19)$$

Since  $L$  is much larger than  $e$ , either  $w^* \approx 2$  or  $w^* \approx 0$ . Obviously, the case of  $w^* \approx 0$  is of no practical interest, which leaves the case that  $w^* \approx 2$ . Furthermore, Eq. (19)  $> 0$  because the first term of Eq. (18) is 0 due to Eq. (16),

$w^* > 1$  due to Eq. (17), and  $(1 - 4e^2 + 4e^2L^2) > 0$  as long as  $L$  is much larger than  $e$ . As a result, the optimal  $q^* = \sqrt{w^*} = \sqrt{2}$  is a minimum point of Eq. (15). However,  $q$  is the number of pixels to be merged in each layer and must be greater than 1, which implies that the best choice for  $q^*$  approximating  $\sqrt{2}$  is 2.

### Acknowledgment

Pau-Chung Chung and Chein-I Chang would like to thank the National Science Council, for supporting this work under the Grant numbers NSC 84-2213-E-006-087 and NSC 84-2213-E-006-086.

### References

1. T. Pun, "A new method for gray-level picture thresholding using the entropy of histogram," *Signal Process.* **2**, 223–237 (1980).
2. J. N. Kapur, P. K. Sahoo, and A. K. C. Wong, "A new method for gray-level picture thresholding using the entropy of histogram," *Comput. Graph. Vis. Image Process.* **29**, 273–285 (1985).
3. N. Pal and S. Pal, "Entropic thresholding," *Signal Process.* **16**, 97–108 (1989).
4. C.-I. Chang, K. Chen, J. Wang, and M. Althouse, "A relative entropy-based approach to image thresholding," *Pattern Recog.* **29**, 1275–1289 (1994).
5. A. S. Abutaleb, "Automatic thresholding of gray-level pictures using two-dimensional entropy," *Comput. Graph. Vis. Image Process.* **47**, 22–32 (1989).
6. A. D. Brink, "Thresholding of digital images using two-dimensional entropies," *Pattern Recog.* **25**, 803–808 (1992).
7. W.-T. Chen, C.-H. Wen, and C.-W. Yang, "A fast two-dimensional entropic thresholding algorithm," *Pattern Recog.* **27**, 885–893 (1994).
8. R. M. Haralick, K. Shanmugan, and I. Dinstein, "Texture features for image segmentation," *IEEE Trans. Syst. Man Cybernet.* **3**, 610–621 (1973).
9. M. L. G. Althouse and C.-I. Chang, "Target detection in multispectral images using the spectral co-occurrence matrix and entropy thresholding," *Opt. Eng.* **34**, 2135–2148 (1995).
10. A. Gersho and R. G. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, Boston (1992).
11. Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.* **28**, 84–95 (1980).



**Pau-Choo Chung** received BS and MS degrees from the Department of Electrical Engineering of the National Cheng Kung University, Taiwan, in 1981 and 1983, respectively, and the PhD degree in electrical engineering from Texas Tech University in 1991. From 1983 to 1986 she was with the Chung Shan Institute of Science and Technology, Taiwan. She is currently an associate professor in the Department of Electrical Engineering of National Cheng-Kung University, Taiwan. Her research is in the areas of neural networks with applications to medical image processing, computed tomography/magnetic resonance image analysis, and mammography.



**Chein-I Chang** received his BS, MS, and MA degrees from Soochow University, Taipei, Taiwan, in 1973, the National Tsing Hua University, Hsinchu, Taiwan, in 1975, and the State University of New York at Stony Brook in 1977, respectively, all in mathematics, and MS and MSEE degrees from the University of Illinois at Urbana-Champaign in 1980 and 1982, respectively, and PhD degree in electrical engineering from the University of Maryland, College Park, in 1987. He was a visiting assistant professor from January 1987 to August 1987, assistant professor from 1987 to 1993, and is currently an associate professor in the Department of Computer Science and Electrical Engineering at the University of Maryland, Baltimore County. He was a visiting specialist in the Institute of Information Engineering at National Cheng Kung University, Taiwan, from 1994 to 1995. His research interests include information theory and coding, signal detection and estimation, multispectral/hyperspectral remote sensing, neural networks, and pattern recognition. Dr. Chang is a senior member of IEEE and a member of SPIE, INNS, Phi Kappa Phi, and Eta Kappa Nu.



**Ching-Wen Yang** received a BS degree from the Department of Information Engineering Sciences of the Feng-Chia University, Taiwan, in 1987 and an MS degree in information engineering in 1989 from National Cheng Kung University, where he is currently a PhD candidate. His research interests include image processing, biomedical image processing, and computer network.