PERGAMON

Pattern Recognition 32 (1999) 825–843

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

Fuzzy partition of two-dimensional histogram and its application to thresholding

H.D. Cheng*, Yen-Hung Chen

Department of Computer Science, Utah State University, Logan, UT 84322-4205, USA

Received 16 March 1998; received in revised form 15 May 1998

Abstract

This paper proposes a thresholding approach by performing a fuzzy partition on a two-dimensional (2D) histogram of the image. The novel 2D fuzzy partition method and membership assignment are based on fuzzy relationship of two fuzzy sets characterized by a set of the best parameters (a, b, c), which makes the image have the maximum fuzzy entropy. The threshold is selected as the crossover point of the fuzzy region $[a, c]$. The preprocessed moment arrays are employed to reduce the computation time of image entropy from $O(8N^2)$ to $O(2N)$. The proposed approach has been tested on various images, and the results have demonstrated that the proposed 2D fuzzy approach outperforms the 2D nonfuzzy approach and the one-dimensional (1D) fuzzy partition approach. © 1999 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved

Keywords: 2D Histogram; Fuzzy partition; Fuzzy entropy; Thresholding; Bright extraction; Dark extraction

1. Introduction

Image thresholding is an important technique for image processing and pattern recognition. It is also regarded as the first step for image understanding. Generally, a good threshold will result in good visualization and increase the accuracy and efficiency of the subsequent processing. Various methods have been proposed to automatically select the thresholds [1–3]. In general, thresholding techniques can be classified as global or local according to the nature of the algorithms. In the global thresholding methods, the threshold is selected by optimizing (maximization or minimization) some criterion functions defined from images. In the local thresholding techniques, the threshold selection is based on local properties of the statistical histogram of the image by searching for the local maxima or minima in the histogram. Based on different definitions of the criterion functions, thresholding techniques can be further classified as point-dependent or region-dependent methods. For a point-dependent method, only the pixel intensity is considered in the criterion functions for threshold selection, while a region-dependent method considers both the pixel intensity and the local information among pixels. Most thresholding techniques can be extended to the case of multithresholding. Thus, we focus on the bi-level thresholding techniques. In bi-level thresholding, it is assumed that an image is composed of object class and background class, which can be distinguished by

*Corresponding author.

PII: S0031-3203(98)00080-6

comparing the gray level values with a suitably selected threshold value.

The motivation of this paper is to propose a fuzzy approach for image thresholding based on both intensity distribution and local information among pixels. The purpose of this paper is to automatically determine the fuzzy region and find the bi-level threshold based on the maximum fuzzy entropy principle. Our approach involves a novel fuzzy partition on a 2D histogram where a 2D fuzzy entropy is defined, and a genetic algorithm is employed to find the optimal result. We will show that it is necessary to include the local information for threshold selection, and prove that the results using the proposed 2D fuzzy thresholding method are better than those using 2D nonfuzzy approach and 1D fuzzy approach.

2. Fuzzy set theory

2.1. Fuzzy set and membership function

A *fuzzy set A* in the observed space X is characterized by a membership function $\mu_A(x)$ that associates each element x of X with a real number, $\mu_A(x)$, in the interval [0, 1].The value $\mu_A(x)$ indicates the degree of the element x belonging to A. Generally, a fuzzy set A is defined as a collection of ordered pairs and can be expressed by the following notations [4]:

$$\begin{aligned} A &= \{(\mu_A(x_i), x_i) \mid i = 1, 2, \dots, n\} \\ &= \frac{\mu_A(x_1)}{x_1} + \frac{\mu_A(x_2)}{x_2} + \cdots + \frac{\mu_A(x_n)}{x_n} \\ &= \sum_{i=1}^{n} \frac{\mu_A(x_i)}{x_i} \\ &= \bigcup_{i=1}^{n} \frac{\mu_A(x_i)}{x_i}, \end{aligned}$$

where the plus sign means the union.

The selection of membership function depends on the applications. We introduce two membership functions which are used in our experiments. One is the S-function and the other is the Z-function. Other membership functions can also be found in [5,6].

The S-function is defined as

$$\mu_A(x) = S(x, a, b, c) = \begin{cases} 0, & x \leqslant a \\ \dfrac{(x-a)^2}{(b-a)\,(c-a)}, & a \leqslant x \leqslant b, \\ 1 - \dfrac{(x-c)^2}{(c-b)\,(c-a)}, & b \leqslant x \leqslant c \\ 1, & c \leqslant x, \end{cases} \tag{1}$$

where x is the observed variable. Here, x is the gray level between 0 and 255. The parameters a, b, and c determine the shape of the S-function. The interval $[a, c]$ is called the *fuzzy region*, where b is usually set as the midpoint of interval $[a, c]$, but it is not necessary. If b is defined as the midpoint between a and c, then the S-function is called a *standard S-function*. Notice that a standard S-function is symmetric about point b and the crossover point has membership 0.5 occurring at point b.

A Z-function can be obtained from an S-function by the following equation:

$$Z(x, a, b, c) = 1 - S(x, a, b, c). \tag{2}$$

In the proposed approach, the crossover point corresponding to membership 0.5 in the fuzzy domain does not need to be point b. This asymmetric membership function is more flexible than the standard S-function which is a special case of the asymmetric S-function.

2.2. Fuzzy relation

The *Cartesian product* of two universes, X and Y, is defined as

$$X \times Y = \{(x, y) \mid x \in X, y \in Y\}.$$

If $X = \{1, 2, 3\}$ and $Y = \{a, b, c\}$, then

$$X \times Y = \{(1, a), (1, b), (1, c), (2, a), (2, b), (2, c), (3, a), (3, b), (3, c)\}.$$

For a full Cartesian product, every element in universe X is completely related to every element in universe Y. A subset of the Cartesian product $X \times Y$ is called a relation over X and Y.

For a *fuzzy relation*, the strength of the relationship between the elements of two universes is measured by a membership function which represents various "degrees" of strength of the relation on the interval [0, 1]. Let $\underset{\sim}{A}$ be a fuzzy set on universe X and $\underset{\sim}{B}$ be a fuzzy set on universe Y. The Cartesian product between fuzzy sets $\underset{\sim}{A}$ and $\underset{\sim}{B}$ will result in a fuzzy relation $\underset{\sim}{R}$, which is a subset of the full Cartesian product space, i.e.,

$$\underset{\sim}{A} \times \underset{\sim}{B} = \underset{\sim}{R} \subset X \times Y,$$

where the strength of the relationship for each pair of (x, y) from $\mathrm{A} \times \mathrm{B}$ is selected as the smaller membership of x and y [7], i.e.,

$$\mu_{\underset{\sim}{R}}(x, y) = \mu_{\underset{\sim}{A} \times \underset{\sim}{B}}(x, y) = \min(\mu_{\underset{\sim}{A}}(x), \mu_{\underset{\sim}{B}}(y)). \tag{3}$$

2.3. Fuzzy partition

Since the membership for an element in a fuzzy set could be any real number in [0, 1], an element can partially belong to each subset under a fuzzy partition. We use an example to demonstrate the fuzzy partition.

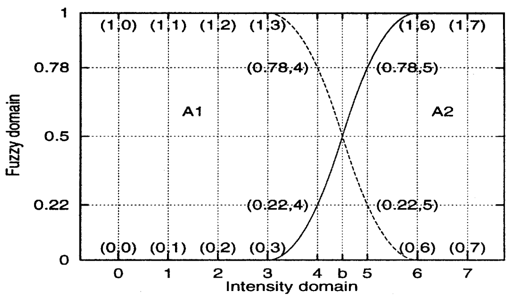


Fig. 1. 1D fuzzy partition characterized by (a, b, c)–(3, 4, 5, 6), where μ_{A_1} is a Z function and μ_{A_2} is the corresponding S-function. The location variable b will influence the value of μ_A and μ_A.

Let $X = \{0, 1, \ldots, 7\}$ be the intensity scale of an image where 0 means the most dark intensity and 7 means the most bright one. A bilevel fuzzy partition can be made on X to get two fuzzy subsets A_1 and A_2 by letting

$$A_1 = \sum_{i=0}^{7} \frac{\mu_{A_1}(x_i)}{x_i} = \frac{1}{0} + \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \frac{0.78}{4} + \frac{0.22}{5} + \frac{0}{6} + \frac{0}{7},$$

$$A_2 = \sum_{i=0}^{7} \frac{\mu_{A_2}(x_i)}{x_i} = \frac{0}{0} + \frac{0}{1} + \frac{0}{2} + \frac{0}{3} + \frac{0.22}{4} + \frac{0.78}{5} + \frac{1}{6} + \frac{1}{7},$$

where $\mu_{A_1}(x)$ is a Z-function and $\mu_{A_2}(x)$ is its corresponding S-function. The partition is shown in Fig. 1.

Basically, there are two tasks to perform for a fuzzy partition.

1. *Fuzzy region determination*: The fuzzy region is an important feature of a fuzzy partition. Unlike a crisp partition, the intersection of two fuzzy subsets, A and its complement, is not empty. In the above example of using the S-function (Z-function) as a membership function, the fuzzy region is the interval $[a, c] = [3,6]$.
2. *Membership assignment*: The elements in the fuzzy region have the uncertainty of belonging to fuzzy set A_1 or A_2. The location of point b can influence the determination of memberships μ_{A_1} and μ_{A_2}.

Therefore, a fuzzy partition can be entirely characterized by a combination of a, b, and c.

2.4. Fuzzy entropy

According to *information theory* [8], the *entropy* of a system is a measure of the amount of information of the system. Let $x_i, i = 1, \ldots, N$, be the possible outputs from source A with the probability $P(x_i)$. An entropy is defined as

$$H_{nonfuzzy}(A, P) = -\sum_{i=1}^{N} P(x_i) \log P(x_i), \tag{4}$$

where

$$\sum_{i=1}^{N} P(x_i) = 1.$$

The subscript "nonfuzzy" is used to distinguish from the fuzzy entropy. Larger entropy implies a larger amount of information. In fact, Eq. (4) describes the entropy in the ordinary domain. What we need is a fuzzy entropy which can measure the amount of information for an image with fuzzy nature.

Many definitions of fuzzy entropy are studied [4,9,10]. Most of them incorporate the membership $\mu_A(x_i)$ with the probabilities $P(x_i)$ in the measure of fuzzy entropy. In order to measure the information amount of a fuzzy set, Zadeh suggests a definition about the entropy of a fuzzy set which takes both probabilities and memberships of elements into consideration [4]. Let A be a fuzzy set with membership function $\mu_A(x_i)$, where x_i, $i = 1, \ldots, N$, are the possible outputs from source A with the probability $P(x_i)$. The fuzzy entropy of set A is defined as

$$H_{fuzzy}(A) = -\sum_{i=1}^{N} \mu_A(x_i)\, P(x_i) \log P(x_i). \tag{5}$$

The difference between Eqs. (5) and (4) is the term $\mu_A(x_i)$, which serves as a weighted multiplier in Eq. (5). Therefore, Zadeh's fuzzy entropy is also called a weighted entropy.

3. Proposed method

We propose a novel 2D fuzzy partition approach whose idea is that the optimal fuzzy partition divides all pixels of the image into two fuzzy subsets, i.e., object class and background class. The fuzzy partition also maximizes a posterior image entropy that is the sum of entropies of two fuzzy subsets. Since a fuzzy partition can be characterized by a combination of the three parameters a, b, and c, the major task of the proposed approach is to find a set of (a, b, c) which maximizes the fuzzy entropy of the image. This is an optimization prooblem which can be solved by any of the following method: heuristic searching, simulated annealing, genetic algorithm, etc. In this paper, we use genetic algorithm [11] to search for the optimal solution.

The proposed method consists of the following three steps:

1. Find 2D histogram of the image.

2. Make fuzzy partition on 2D histogram.
3. Compute fuzzy entropy.

Step 1 needs to be executed only once while steps 2 and 3 should be performed iteratively for each trial set of (a, b, c) . The optimum (a, b, c) determines the fuzzy region (i.e., interval $[a, c]$). The threshold is selected as the crossover point which has membership 0.5 implying the largest fuzziness.

3.1. 2D Histogram

In order to acquire 2D histogram of an image, the local average should be defined. Suppose a digital image with $M \times N$ pixels and L gray scales $(0, \ldots, L-1)$ is represented as a 2D intensity function $f(x, y)$. We define the local average of $f(x, y)$ as the average intensity of its four neighbors. The value is rounded to an integer and denoted by $g(x, y)$, i.e.,

$$g(x, y) = \lfloor \tfrac{1}{4}[f(x, y+1) + f(x, y-1) + f(x+1, y) + f(x-1, y)] + 0.5 \rfloor. \quad (6)$$

Actually, the averaging operation comes from a 3×3 lowpass spatial filter. Since the lowpass filters have the features of edge blurring and noise reduction [8], this causes the pixels to fall off the diagonal line of a 2D histogram. Therefore, the pixels, which have the same intensity but have different spatial features, can be identified in the second dimension (local average gray level) of a 2D histogram. A 2D histogram should be an array $(L \times L)$ with the entries representing the number of occurrences of the pair $(g(x, y), f(x, y))$.

3.2. Fuzzzy partition on 2D histogram

As described in Section 2.3, a fuzzy partition should have an overlap between the fuzzy subsets. Applying this concept to the 2D histogram of the image, there should be an overlapped block between the black block $Block_B$ and the white block $Block_W$. Based on the parameters a, b, and c of the S-function (Z-function), the $Block_B$ is defined as the block with the left top point $(0, 0)$ to the right bottom point (c, c), the $Block_W$ is the block from left top point (a, a) and the right bottom point $(L-1, L-1)$. The overlapped block is denoted by $Block_{fuzzy}$, which begins at point (a, a) and ends at point (c, c). The fuzzy partition is shown in Fig. 2.

After determining the fuzzy region in the 2D histogram of the image, the next task is to determine the membership for each entry of the 2D histogram. Since the entries in a 2D histogram are classified into two fuzzy subsets, there are two memberships to be defined for each entry. One is for the fuzzy set *Bright*, which describes the brightness for each entry, the other is for the fuzzy set *Dark*, which describes how dark the entry is. A 2D histogram

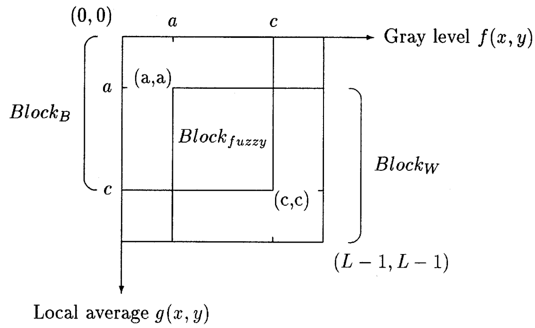


Fig. 2. A fuzzy on a 2D histogram, where $Block_B$ is the block from from $(0, 0)$ to (v, v) $Block_W$ is the block from (a, a) to $(L-1, L-1)$, and $Black_{fuzzy}$ is the overlap block from (a, a) to (c, c).

consists of all the possible pairs of (local average gray level, gray level), and it can be viewed as a full Cartesian product of two sets X and Y, where X represents the local average gray levels of the pixels and Y represents the gray levels of the pixels. Basically, both sets X and Y represent the same set composed of all possible intensities, i.e.

$$X = Y = \{0, 1, 2, \ldots, L-1\}.$$

In order to define two fuzzy sets *Bright* and *Dark* on the 2D histogram, four fuzzy sets are defined in advance based on the S-function and the corresponding Z-function with parameters (a, b, c). They are $BrightX$, $DarkX$, $BrightY$, and $DarkY$, respectively, where $BrightX$ and $DarkX$ are the fuzzy sets defined for the elements of X, and $BrightY$ and $DarkY$ are the fuzzy sets defined for the elements of Y. They are described as follows:

1. $BrightX$ is a fuzzy set describing the brightness of the observed variable $x \in X$. The membership function $\mu_{BrightX}$ is defined by an S-function. That is,

$$BrightX = \sum_{x \in X} \frac{\mu_{BrightX}(x)}{x} = \sum_{x \in X} \frac{S(x, a, b, c)}{x}.$$

2. $Dark\ X$ is a fuzzy set describing the darkness of the observed variable $x \in X$. The membership function μ_{DarkX} is defined by a Z-function. That is,

$$DarkX = \sum_{x \in X} \frac{\mu_{DarkX}(x)}{x} = \sum_{x \in X} \frac{Z(x, a, b, c)}{x}.$$

3. $BrightY$ is a fuzzy set describing the brightness of the observed variable $y \in Y$. The membership function $\mu_{BrightY}$ is defined by an S-function. That is,

$$BrightY = \sum_{y \in Y} \frac{\mu_{BrightY}(y)}{y} = \sum_{y \in Y} \frac{S(y, a, b, c)}{y}.$$

4. *DarkY* is a fuzzy set describing the darkness of the observed variable $y \in Y$. The membership function μ_{DarkY} is defined by a Z-function. That is,

$$DarkY = \sum_{y \in Y} \frac{\mu_{DarkY}(y)}{y} = \sum_{y \in Y} \frac{Z(y, a, b, c)}{y}.$$

The fuzzy relation *Bright* caused by the Cartesian product between fuzzy sets *BrightX* and *BrightY* can be used to describe the brightness of the entries in 2D histograms, which is a subset of the full Cartesian product space $X \times Y$, i.e.

$$Bright = BrightX \times BrightY \subset X \times Y.$$

The membership of the fuzzy relation *Bright* can be obtained by referring to Eq. (3) as follows:

$$\mu_{Bright}(x, y) = \mu_{BrightX \times BrightY}(x, y) = \min(\mu_{BrightX}(x), \mu_{BrightY}(y)). \quad (7)$$

Similarly, a fuzzy relation *Dark* can be obtained by the Cartesian product of fuzzy sets *DarkX* and *DarkY*:

$$Dark = DarkX \times DarkY \subset X \times Y,$$

where the fuzzy relation *Dark* has the membership function

$$\mu_{Dark}(x, y) = \mu_{DarkX \times DarkY}(x, y) = \min(\mu_{DarkX}(x), \mu_{DarkY}(y)). \quad (8)$$

The memberships for the fuzzy sets *Bright* and *Dark* are depicted as contour lines in Fig. 3.

Notice that the fuzzy sets *Bright* and *Dark* are not complementary to each other in the proposed 2D histogram approach, i.e., the sum of $\mu_{Bright}(x, y)$ and $\mu Dark(x, y)$ for every (x, y) of a 2D histogram is not necessarily equal to unity, i.e., in general,

$$\mu_{Bright}(x, y) \neq 1 - \mu_{Dark}(x, y).$$

3.3. Fuzzy entropy

After the 2D histogram is divided into object and background classes ($Block_B$ and $Block_W$) and the memberships μ_{Bright} and μ_{Dark} are determined, their entropies are calculated separately and the total image entropy is defined as the summation:

$$H(image) = H(Block_B) + H(Block_W). \quad (9)$$

As shown in Fig. 4, the dark block $Block_B$ can be divided into a nonfuzzy region R_B and a fuzzy region R_1 according to the membership value μ_{Dark} equal to one or not, i.e.

$$Block_B = R_B \cup R_1,$$

$$R_B = \{(x, y) \,|\, \mu_{Dark}(x, y) = 1, (x, y) \in Block_B\},$$

$$R_1 = \{(x, y) \,|\, \mu_{Dark}(x, y) < 1, (x, y) \in Block_B\}.$$

In a similar way, the bright block $Block_W$ is composed of a nonfuzzy region R_W and a fuzzy region R_2 according to the value of membership function μ_{Bright}, i.e.

$$Block_W = R_W \cup R_1,$$

$$R_W = \{(x, y) \,|\, \mu_{Bright}(x, y) = 1, (x, y) \in Block_W\}$$

$$R_2 = \{(x, y) \,|\, \mu_{Bright}(x, y) < 1, (x, y) \in Block_W\}.$$

$H(Block_B)$ is composed of two entropies. One is a nonfuzzy entropy from R_B, the other is a fuzzy entropy from R_1. The same situation is applied to the evaluation of $H(Block_W)$. Therefore,the entropies of $Block_B$ and $Block_W$ are defined as

$$H(Block_B) = H_{fuzzy}(R_1) + H_{nonfuzzy}(R_B), \quad (10)$$

$$H(Block_W) = H_{fuzzy}(R_2) + H_{nonfuzzy}(R_W). \quad (11)$$

Applying Eq. (5) for the fuzzy entropy evaluation of $H_{fuzzy}(R_1)$ and $H_{fuzzy}(R_2)$, and Eq. (4) for the nonfuzzy

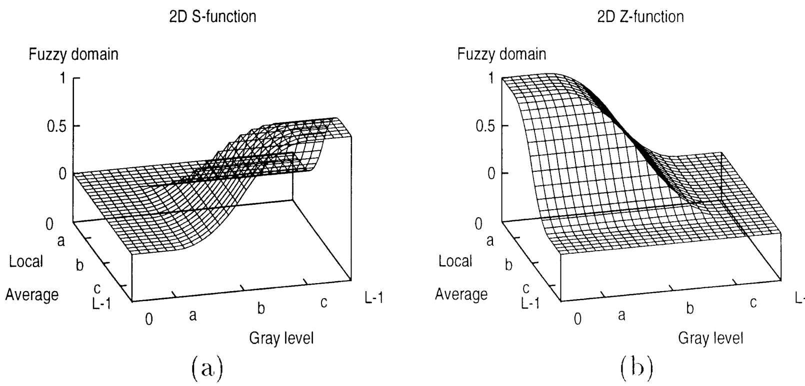


Fig. 3. Fuzzy membership assignment in 2D histograms: (a) μ_{Bright}. (b) μ_{Dark}.

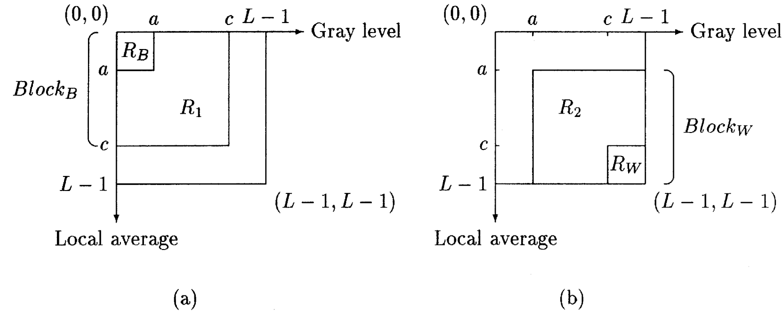


Fig. 4. Two fuzzy entropies from R_1 and R_2, and two nonfuzzy entropies from R_B and R_W (a) $Black_B$ in 2D histogram is divided into R_B and R_W. (b) $Black_B$ in 2D histogram into R_W and R.

entropy evaluation of $H_{nonfuzzy}(R_B)$ and $H_{nonfuzzy}(R_W)$, we get the following four entropies:

$$H_{fuzzy}(R_1) = -\sum_{(x,y)\in R_1} \mu_{Dark}(x,y)\frac{n_{xy}}{\sum_{(x,y)\in R_1} n_{xy}} \times \log\frac{n_{xy}}{\sum_{(x,y)\in R_1} n_{xy}}, \tag{12}$$

$$H_{nonfuzzy}(R_B) = -\sum_{(x,y)\in R_B} \frac{n_{xy}}{\sum_{(x,y)\in R_1} n_{xy}} \times \log\frac{n_{xy}}{\sum_{(x,y)\in R_R} n_{xy}} \tag{13}$$

$$H_{fuzzy}(R_2) = -\sum_{(x,y)\in R_2} \mu_{Bright}(x,y)\frac{n_{xy}}{\sum_{(x,y)\in R_2} n_{xy}} \times \log\frac{n_{xy}}{\sum_{(x,y)\in R_2} n_{xy}} \tag{14}$$

$$H_{nonfuzzy}(R_W) = -\sum_{(x,y)\in R_W} \frac{n_{xy}}{\sum_{(x,y)\in R_3} n_{xy}} \times \log\frac{n_{xy}}{\sum_{(x,y)\in R_1} n_{xy}} \tag{15}$$

where n_{xy} is the element in the 2D histogram which represents the number of occurrences of the pair (x, y). The membership functions $\mu_{Bright}(x, y)$ and $\mu_{Dark}(x, y)$ are defined in Eqs. (7) and (8), respectively. It should be noticed that the probability computations $n_{xy}/\sum n_{xy}$ in the four regions are independent of each other.

3.4. Thresholding

Once the thresholding value t is obtained from the proposed approach, we can conduct image thresholding based on the following formula:

$$f_t(x,y) = \begin{cases} g_0, & f(x,y) < t, \\ g_1, & f(x,y) \geqslant t. \end{cases} \tag{16}$$

where $f(x, y)$ stands for the intensity function of the image at (x, y).

Another thresholding method is based on 2D histograms. A threshold vector (s, t) divides the 2D histogram into four blocks, i.e., a dark block $Block_0$, a bright block $Block_1$, and two noise (edge) blocks, $Block_2$ and $Block_3$. Chen et al. [12] proposed a thresholding method which extracts the bright block from the image and leaves the noise blocks together with the dark block as the other class (dark). This bright extraction method is expressed as:

$$f_{s,t}(x,y,bright) = \begin{cases} g_1, & f(x,y) \geqslant t \wedge g(x,y) \geqslant s, \\ g_0, & \text{otherwise}, \end{cases} \tag{17}$$

where $g(x, y)$ is the local average of $f(x, y)$ defined in Eq. (6). Conversely, if extracting the dark portion is desired, Eq. (17) can be modified by

$$f_{s,t}(x,y,dark) = \begin{cases} g_0, & f(x,y) < t \wedge g(x,y) < s, \\ g_1, & \text{otherwise}. \end{cases} \tag{18}$$

4. Speed up the computation

For each set of (a, b, c), GA should calculate the corresponding fitness value $H(image)$. It is the summation of four entropies corresponding to Eqs. (12), (13), (14), and (15) respectively. Let n_{xy} be the number of occurrences of the pair (x, y) in the 2D histogram and N be the dimension of the region (R_1, R_B, R_2, or R_W). For computing each entropy, it needs $O(N^2)$ time units to obtain the summation of n_{xy} in that region, and another $O(N^2)$ time to collect the entropies contributed by each entries. Thus, it takes $O(2N^2)$ time to compute each of the four entropies. In order to get all four entropies, it requires $O(8N^2)$ computation time. The computation is too expensive even for a heuristic search algorithm like GA. In this section, we will derive an algorithm to compute the fuzzy entropy in $O(N)$ time and the nonfuzzy entropy in a constant time. Therefore, it needs only $O(2N)$ computation time to obtain all four entropies. Since the entropy computation in our approach involves the summation of a rectangle region in a 2D histogram which needs $O(N^2)$ time, the key is to reduce the rectangle summation time.

Ref. [13] precomputed the cumulative moments on all scan lines and saved them in an array so that the summation on a given segment of a scan line can be found through a subtraction. In our experiments, four moment arrays $M1$, $M2$, $M3$, and $M4$ were used to help find the rectangle summation in 2D histograms. All four moment arrays are 1D arrays with L entries. The entries at index l are denoted by $M1[l]$, $M2[l]$, $M3[l]$, and $M4[l]$, respectively, which are computed by the following equations:

$$M1[l] = \sum_{x=0}^{l} \sum_{y=0}^{l} n_{xy}, \quad M2[l] = \sum_{x=0}^{l} \sum_{y=0}^{l} n_{xy} \log n_{xy},$$

$$M3[l] = \sum_{x=l}^{L-1} \sum_{y=l}^{L-1} n_{xy}, \quad M4[l] = \sum_{x=l}^{L-1} \sum_{y=l}^{L-1} n_{xy} \log n_{xy} \tag{19}$$

where n_{xy} is the element at coordinate (x, y) of 2D histograms representing the number of occurrences of (x, y) pair, $l = 0, 1, 2, \ldots, L-1$. The elements of moment arrays are precomputed and saved to represent the cumulative amount of 2D histograms on the left top rectangular region (begins by $(0, 0)$ and ends at (l, l)) or the right bottom rectangular region (begins by (l, l) and ends at $(L-1, L-1)$). Using the moment arrays, the summation of n_{xy} (or $n_{xy}\log n_{xy}$) on the regions R_B, R_1, R_W, and R_2 of 2D histograms can be obtained by subtractions. That is,

$$\sum_{(x,y)\in R_B} n_{xy} = M1[a], \qquad \sum_{(x,y)\in R_1} n_{xy} = M1[c] - M1[a],$$

$$\sum_{(x,y)\in R_W} n_{xy} = M3[c], \qquad \sum_{(x,y)\in R_2} n_{xy} = M3[a] - M3[c]. \tag{20}$$

The same situation can be applied to the summation of $n_{xy} \log n_{xy}$ by using moment arrays $M2$ and $M4$ instead of $M1$ and $M3$. This means that the summation of n_{xy} (or $n_{xy}\log n_{xy}$) on regions R_B, R_1, R_W, or R_2 can be evaluated in a constant time independent of their dimensions.

Consider the fuzzy entropy $H_{fuzzy}(R_1)$ in Eq. (12) and suppose $\sum_{(x,y)\in R_1} n_{xy} = P$. We do the following derivation:

$$H_{fuzzy}(R_1)$$

$$= -\sum_{(x,y)\in R_1} \mu_{Dark}(x,y) \frac{n_{xy}}{\sum_{(x,y)\in R_1} n_{xy}} \log \frac{n_{xy}}{\sum_{(x,y)\in R_1} n_{xy}}$$

$$= \frac{-1}{P} \sum_{(x,y)\in R_1} \mu_{Dark}(x,y)\,(n_{xy} \log n_{xy} - n_{xy} \log P)$$

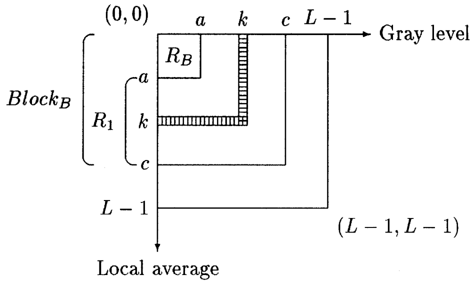


Fig. 5. R_1 is divided into $(c-a)$ subsets based on different value of membrship $\mu_{Dark}(x, y)$, where the stripped region is the set of elements having the same membership $Z(k, a, b, d)$ $\{(x, y) | \mu_{Dark}(x, y) = Z(k, a, b, c)\}$, and $a < k \leqslant r$.

$$= \frac{-1}{P} \sum_{k=a+1}^{c} Z(k) \Bigg[\sum_{\mu_{Dark}(x,y)=Z(k)} \times (n_{xy} \log n_{xy} - n_{xy} \log P) \Bigg]$$

$$= \frac{-1}{P} \sum_{k=a+1}^{c} Z(k) \Bigg[\sum_{\mu_{Dark}(x,y)=Z(k)} n_{xy} \log \times n_{xy} - \sum_{\mu_{Dark}(x,y)=Z(k)} n_{xy}(\log P) \Bigg] \tag{21}$$

where $Z(k)$ is an abbreviation of $Z(k, a, b, c)$. The above derivation is intended to group together the elements which have the same memberships. Therefore, the summation is based on the membership, not on the coordinates. In other words, R_1 can be viewed as a union of the following subsets with different memberships.

$$R_1 = \bigcup_{k=a+1}^{c} \{(x,y) \,|\mu_{Dark}(x,y) = Z(k,a,b,c)\}.$$

The elements with the same membership should be in a connected line as Fig. 5.

By using Eq. (20), we can define the following notations and easily get the values:

$$P \sum_{(x,y)\in R_1} n_{xy} = M1[c] - M1[a].$$

$$Q(k) = \sum_{\mu_{Dark}(x,y)=Z(k,a,b,c)} n_{xy} \log n_{xy}$$

$$= \sum_{(x,y)=(0,0)}^{(k,k)} n_{xy} \log n_{xy} - \sum_{(x,y)=(0,0)}^{(k-1,k-1)} n_{xy} \log n_{xy}$$

$$= M2[k] - M2[k-1]$$

$$R(k) = \sum_{\mu\mu_{\text{Dark}}(x,y)=Z(k,a,b,c)} n_{xy}$$
$$= \sum_{(x,y)=(0,0)}^{(k,k)} n_{xy} - \sum_{(x,y)=(0,0)}^{(k-1,k-1)} n_{xy}$$
$$= M1[k] - M1[k-1].$$

Therefore, the derivation of $H_{fuzzy}(R_1)$ in Eq. (21) can be rewritten as:

$$H_{fuzzy}(R_1) = \frac{-1}{P} \sum_{k=a+1}^{c} Z(k, a, b, c)\,[Q(k) - R(k)\log P].$$

This means that if the moment arrays in Eq. (19) are precomputed and saved, the fuzzy entropy in Eq. (12) can be evaluated in $O(N)$ time, where N is the dimension of R_1 from a to c. The same technique can be applied to compute the fuzzy entropy $H_{fuzzy}(R_2)$ in Eq. (14) and the nonfuzzy entropies $H_{nonfuzzy}(R_B)$ and $H_{nonfuzzy}(R_W)$ in Eqs. (13) and (15), respectively.

In order to make this method work, four moment arrays ($M1$, $M2$, $M3$, and $M4$) in Eq. (19) must be precomputed and saved. $M1$ and $M2$ are used to help the computation of $H(Block_B)$, while $M3$ and $M4$ are used to help the computation of $H(Block_W)$. For an image with L gray levels, the 2D histogram is an $L \times L$ array, therefore, the corresponding moment array is a 1D array with L entries. The required time to set up four moment arrays should be $O(2L^2)$. In our experiments, all the images have 256 gray levels, i.e. $L = 256$. After the moment arrays are saved, the computation time required for the fuzzy entropy in Eqs. (12) or (14) is $O(N)$. For the nonfuzzy entropy of Eqs. (13) or (15), it needs only a constant computation time. Consequently, the image entropy evaluation can be done in $O(2N)$ time for any set of (a, b, c). Notice that the setup of moment arrays needs to be done only once before the computation of the entropies. However, the image entropy evaluation should be done for every set of (a, b, c), which is represented by the individual chromosome in GA. Hence, the preprocessing time $O(2L^2)$ for the setup of moment arrays does not affect the image entropy computation time $O(2N)$.

5. Experimental results

In this section, we test the performance of the proposed approach on various images. Each image has 256 gray levels from 0 (the darkest) to 255 (the brightest). Due to the non-deterministic property of GA, the results will not be the same when we run the GA program several times. Therefore, for each image we run the GA 20 times and choose the best one from these 20 runs. After some iterations of GA, the proposed algorithm can find a combination of a, b, and c which maximizes the image's total entropy. The threshold is selected at the crossover point from the fuzzy region $[a, c]$. All the bilevel threshold images are expressed in two intensities 0 and 255.

5.1. 2D fuzzy approach vs 1D fuzzy approach

In order to prove the importance of spatial information in the selection of threshold, we compare our results with the results of the 1D fuzzy approach [14]. In Table 1, the fuzzy regions and threshold values of the proposed 2D fuzzy approach are shown in the second and third columns, respectively, while the 1D fuzzy approach results are shown in the fourth and fifth columns.

Some of the thresholds found by the proposed method are the same as those from the 1D method. Here we only show a few different results. Three of the thresholded images by Eq. (16) are shown in Figs. 6, 7, and 8, respectively, where part a is the original image, parts b and

Table 2
Threshold vectors using the proposed 2D fuzzy approach and 2D nonfuzzy approach

Image	2D fuzzy threshold vector (t, t)	2D Crisp threshold vector (s, t)
"*masuda2*"	(74, 74)	(174, 182)
"peppers"	(120, 120)	(86, 91)
"soccer"	(88, 88)	(148, 159)
"tiffany"	(187, 187)	(160, 151)

Table 1
Fuzzy regions and thresholds using the proposed 2D fuzzy approach and 1D fuzzy approach

	2D fuzzy		1D fuzzy	
Image	(a, b, c)	Threshold t	(a, c)	Threshold b
"buildings"	(100, 114, 134)	116	(10, 244)	127
"couple"	(13, 46, 47)	37	(10, 41)	26
"tiles"	(57, 81, 117)	84	(18, 171)	95

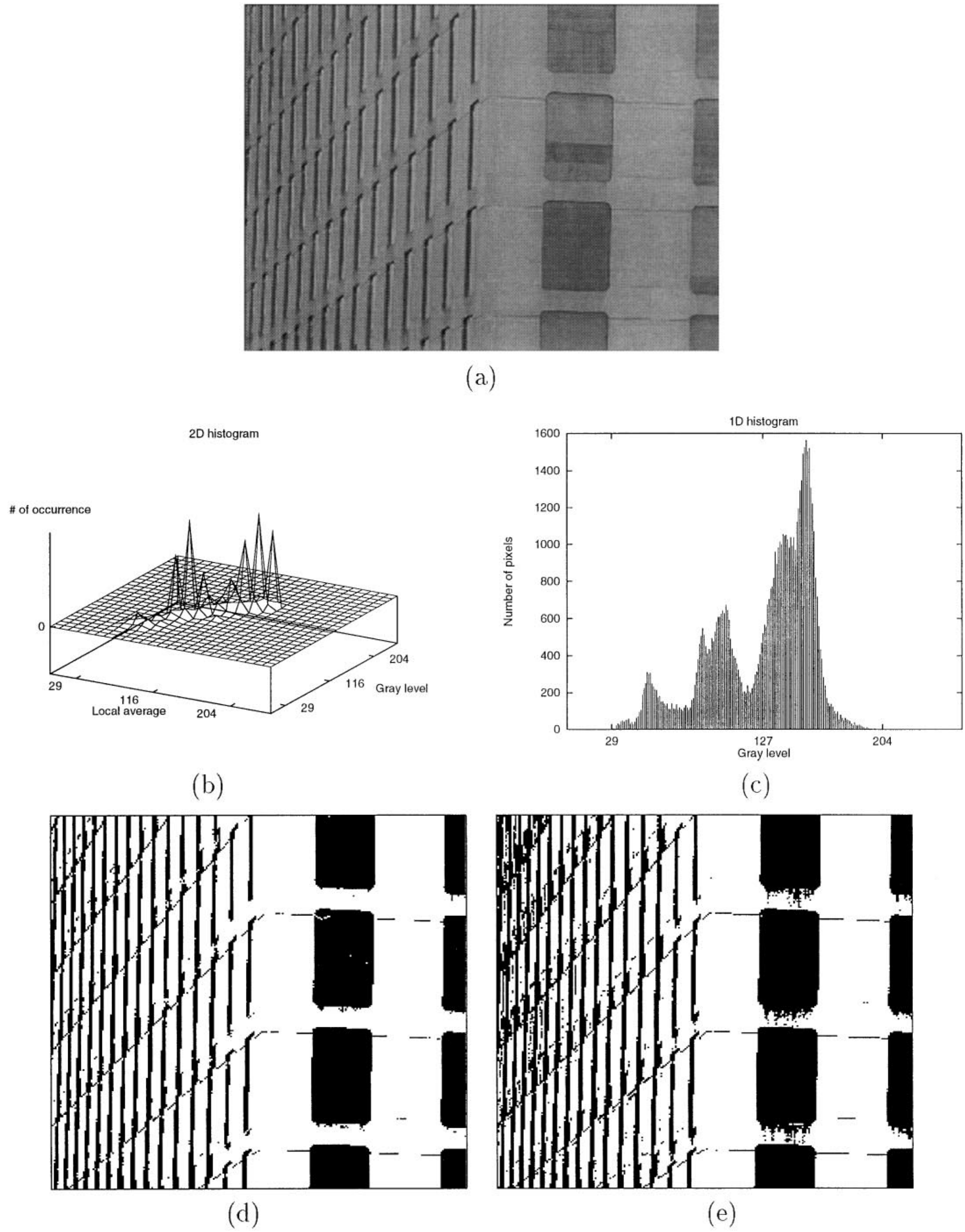


Fig. 6. Comparison between 2D fuzzy thresholding and 1D fuzzy thresholding. (a) Original image "buildings." (b) 2D histogram. (c) 1D histogram. (d) Thresholded image by 2D fuzzy approach with threshold 116. (e) Thresholded image by 1D fuzzy approach with threshold 127.

c show the 2D and 1D histograms of the image, respectively, part d is our thresholded result, and part e is the result from the compared 1D approach. In comparing the two thresholded images of "buildings" in Fig. 6d and e the image in Fig. 6d shows clear shapes for the windows of the building, while the image in Fig. 6e mixes the window and its shadow together. Another example is the image "couple" shown in Fig. 7. The image in Fig. 7d has

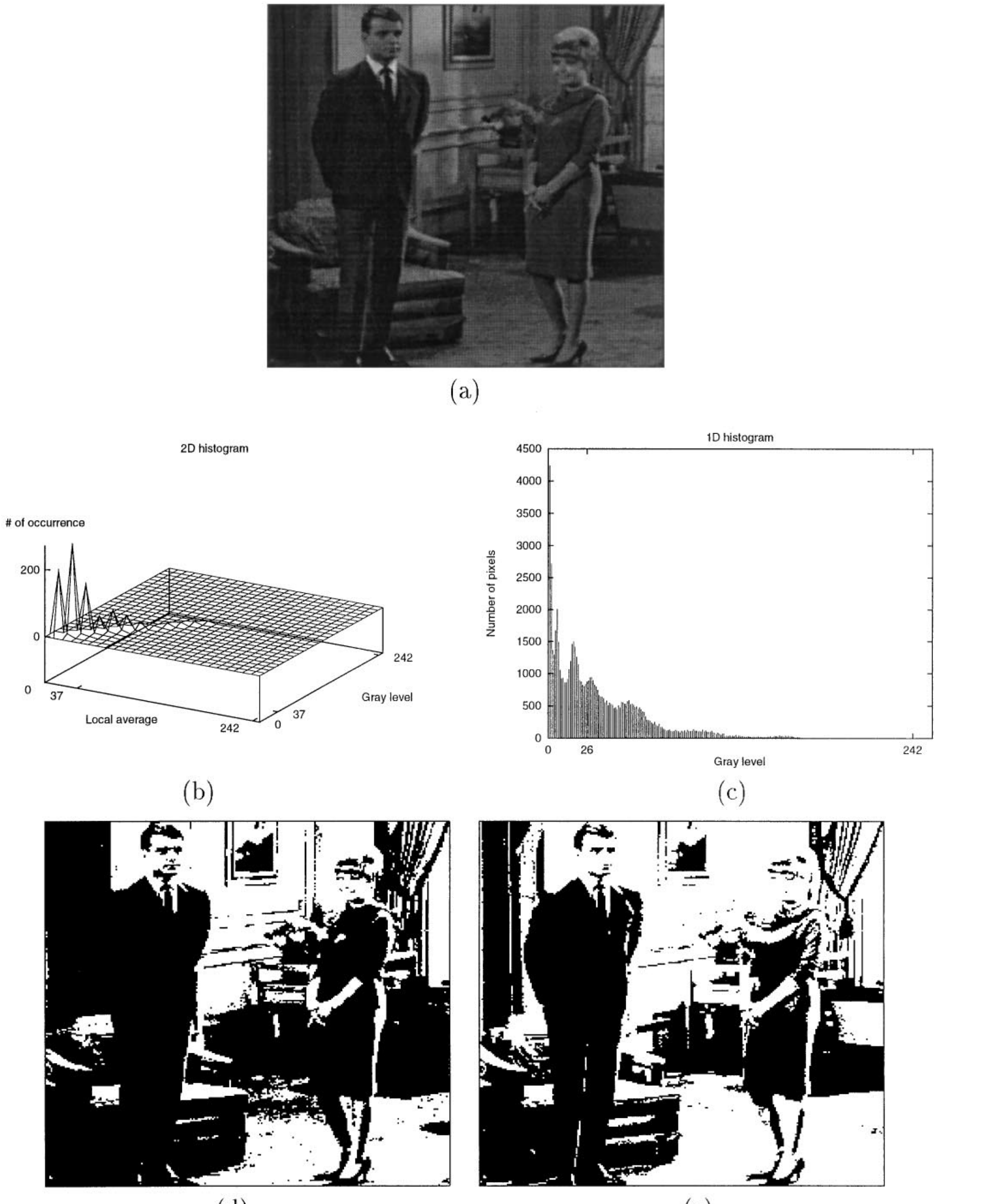


Fig. 7. Comparison between 2D fuzzy thresholding and 1D fuzzy thresholding: (a) Original image "couple"; (b) 2D histogram; (c) 1D histogram; (d) Thresholded image by 2D fuzzy approach with threshold 37; (e) thresholded image by 1D fuzzy approach with threshold 26.

clearer features of the people than those in Fig. 7e. Besides the faces, the picture frame on the wall is clearly depicted in image d than in image e. The third example is shown in Fig. 8. The image from the 1D approach has distortion in the upper left and upper right corners in Fig. 8e, compared with image in Fig. 8d. The results show that the proposed 2D fuzzy approach outperforms the 1D fuzzy approach.

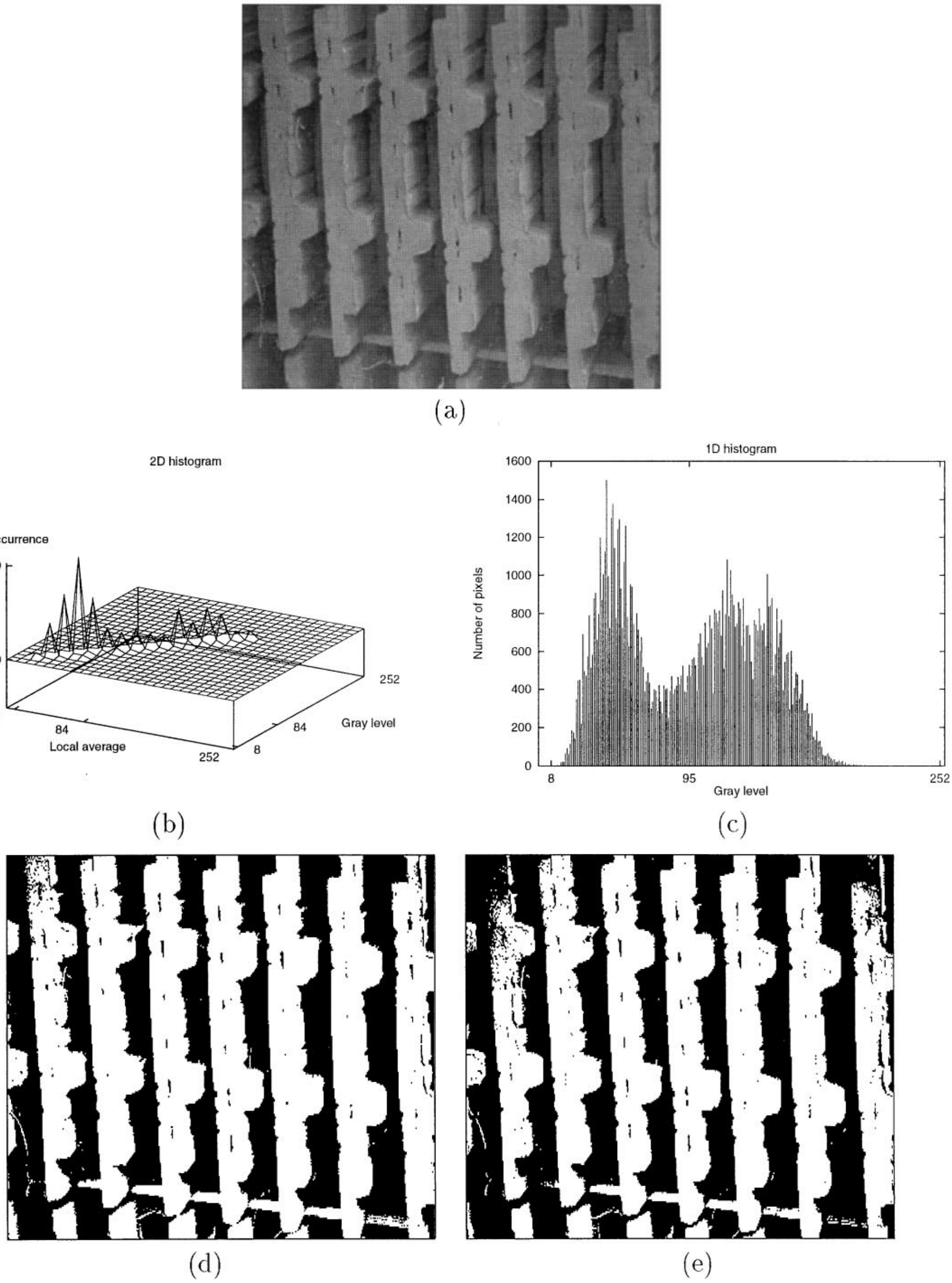


Fig. 8. Comparison between 2D fuzzy thresholding and 1D fuzzy thresholding; (a) original image “tiles” (b) 2D histogram; (c) 1D histogram; (d) thresholded image by 2D fuzzy approach with threshold 84; (e) thresholded image by 1D fuzzy approach with threshold 95.

5.2. 2D fuzzy vs 2D crisp

The proposed 2D fuzzy thresholding results are compared with the 2D nonfuzzy (crisp) thresholding results [15]. We want to prove that the proposed fuzzy partition method can get better thresholded images than the crisp partition method. Table 2 shows the threshold vectors found by both approaches. The comparisons are

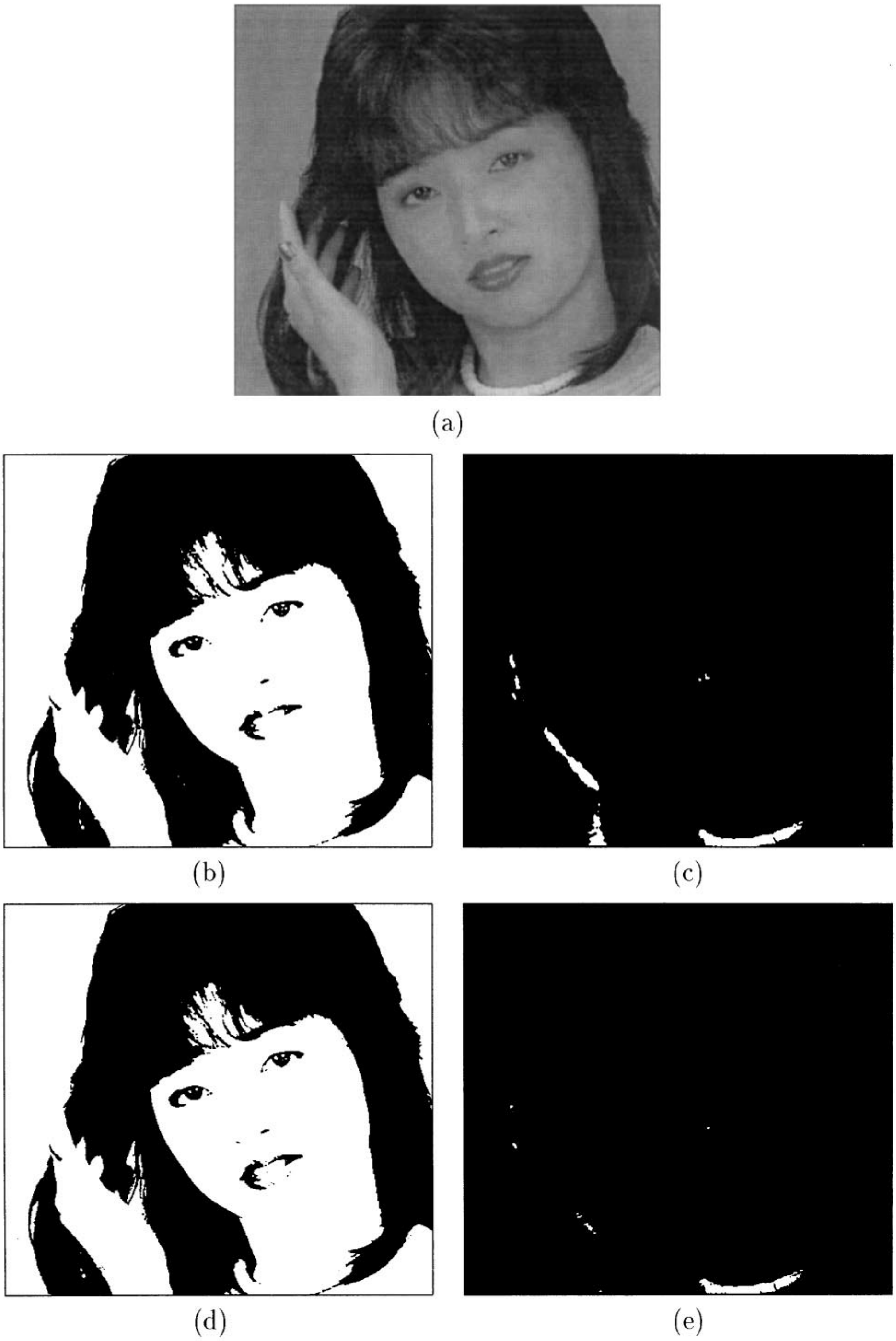


Fig. 9. 2D thresholding on image "masuda2": masuda2 (a) original image; (b) dark extraction by 2D fuzzy approach with threshold vector $(t, t) = (74, 74)$; (c) dark extraction by 2D crisp approach with threshold vector $(s, t) = (174, 182)$; (d) bright extraction by 2D fuzzy approach with threshold vector $(t, t) = (74, 74)$; (e) bright extraction by 2D crisp approach with threshold vector $(s, t) = (174, 182)$.

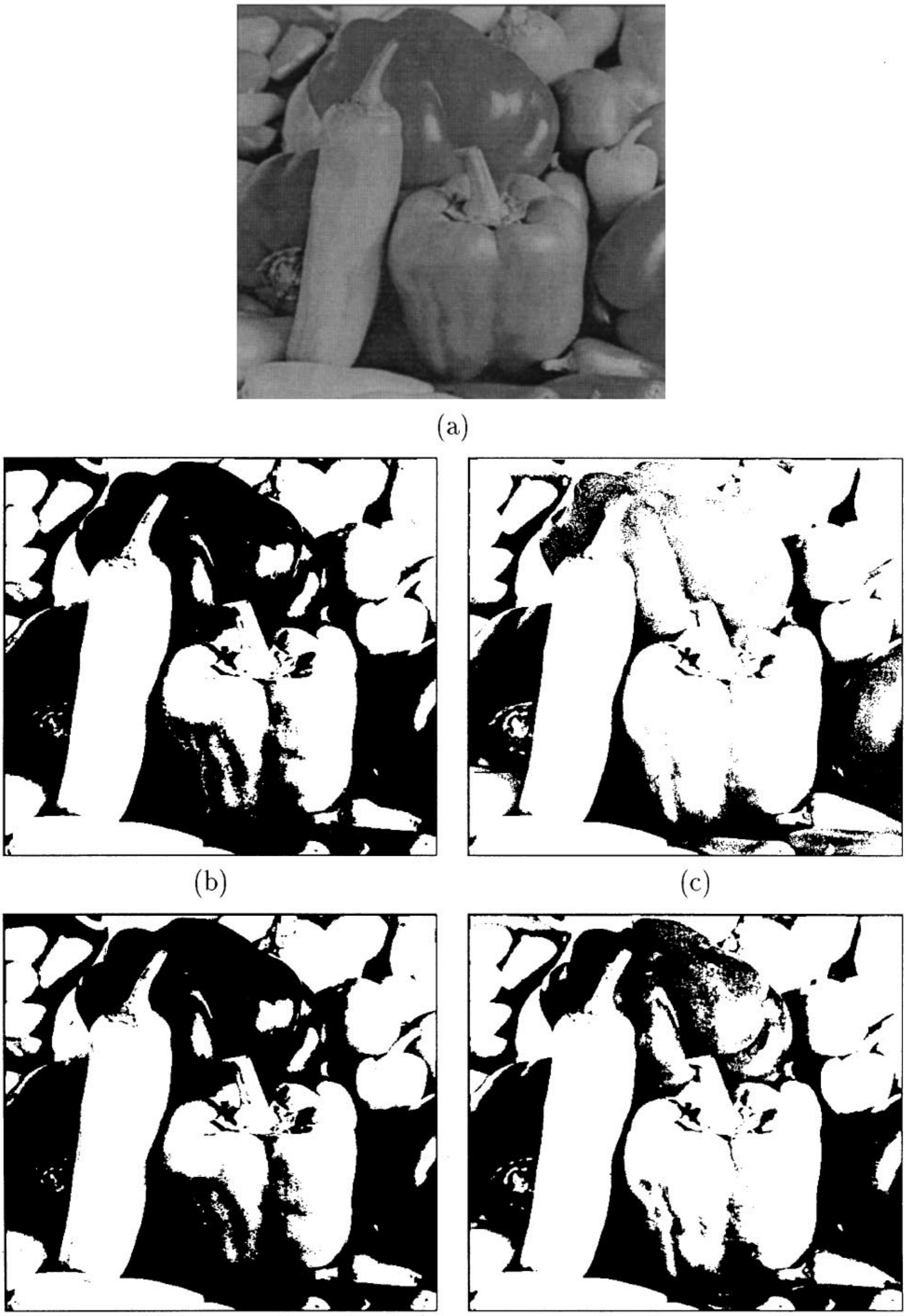


Fig. 10. 2D thresholding on image "peppers": peppers: (a) original image; (b) dark extraction by 2D fuzzy approach with threshold vector $(t, t) = (120, 120)$; (c) dark extraction by 2D crisp approach with threshold vector $(s, t) = (86, 91)$; (d) bright extraction by 2D fuzzy approach with threshold vector $(t, t) = (120, 120)$; (e) bright extraction by 2D crisp approach with threshold vector $(s, t) = (86, 91)$.

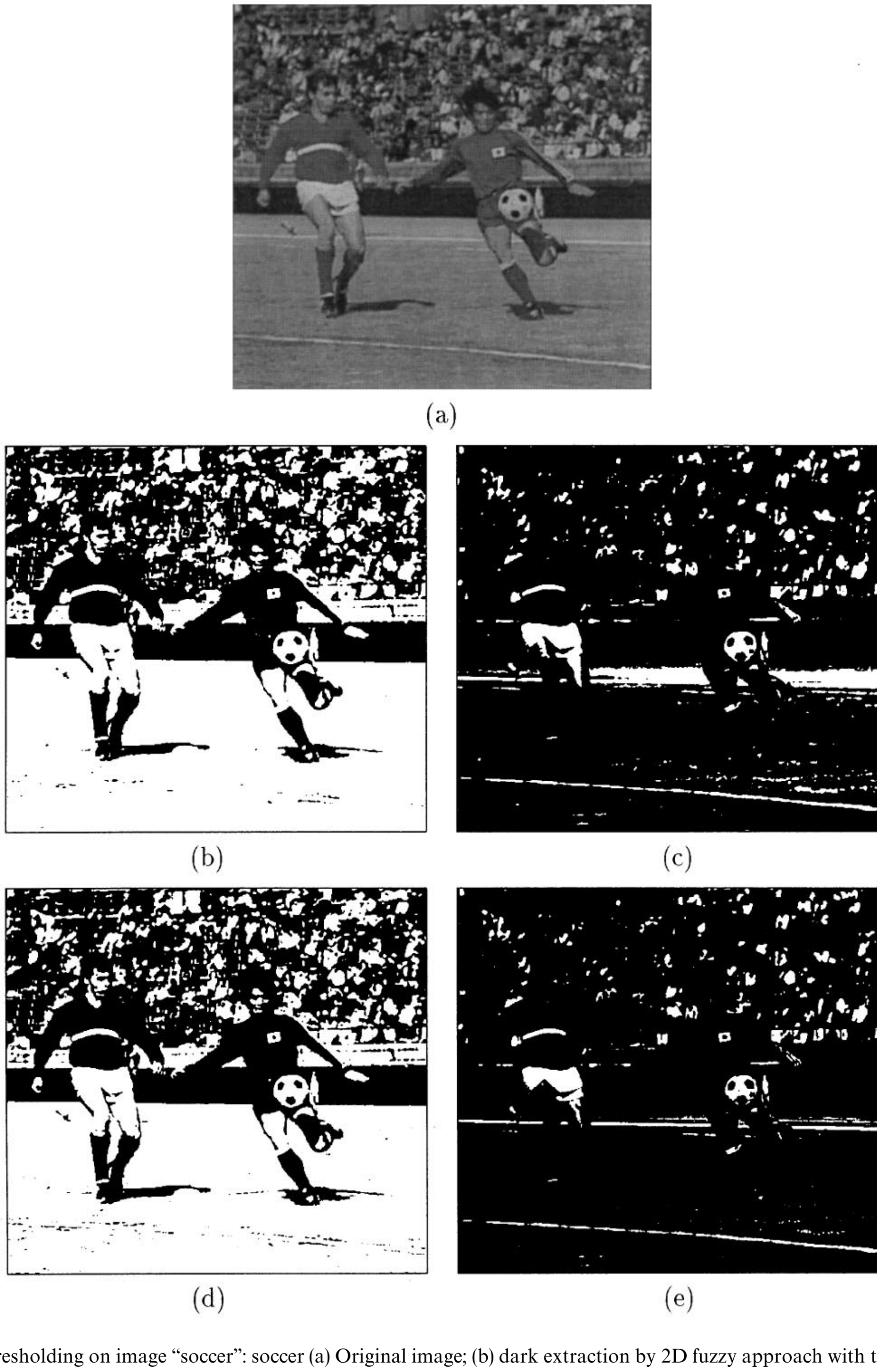


Fig. 11. 2D thresholding on image "soccer": soccer (a) Original image; (b) dark extraction by 2D fuzzy approach with threshold vector $(t, t) = (88, 88)$; (c) dark extraction by 2D crisp approach with threshold vector $(s, t) = (148, 159)$ (d) bright extraction by 2D fuzzy approach with threshold vector $(t, t) = (88, 88)$; (e) bright extraction by 2D crisp approach with threshold vector $(s, t) = (148, 159)$.

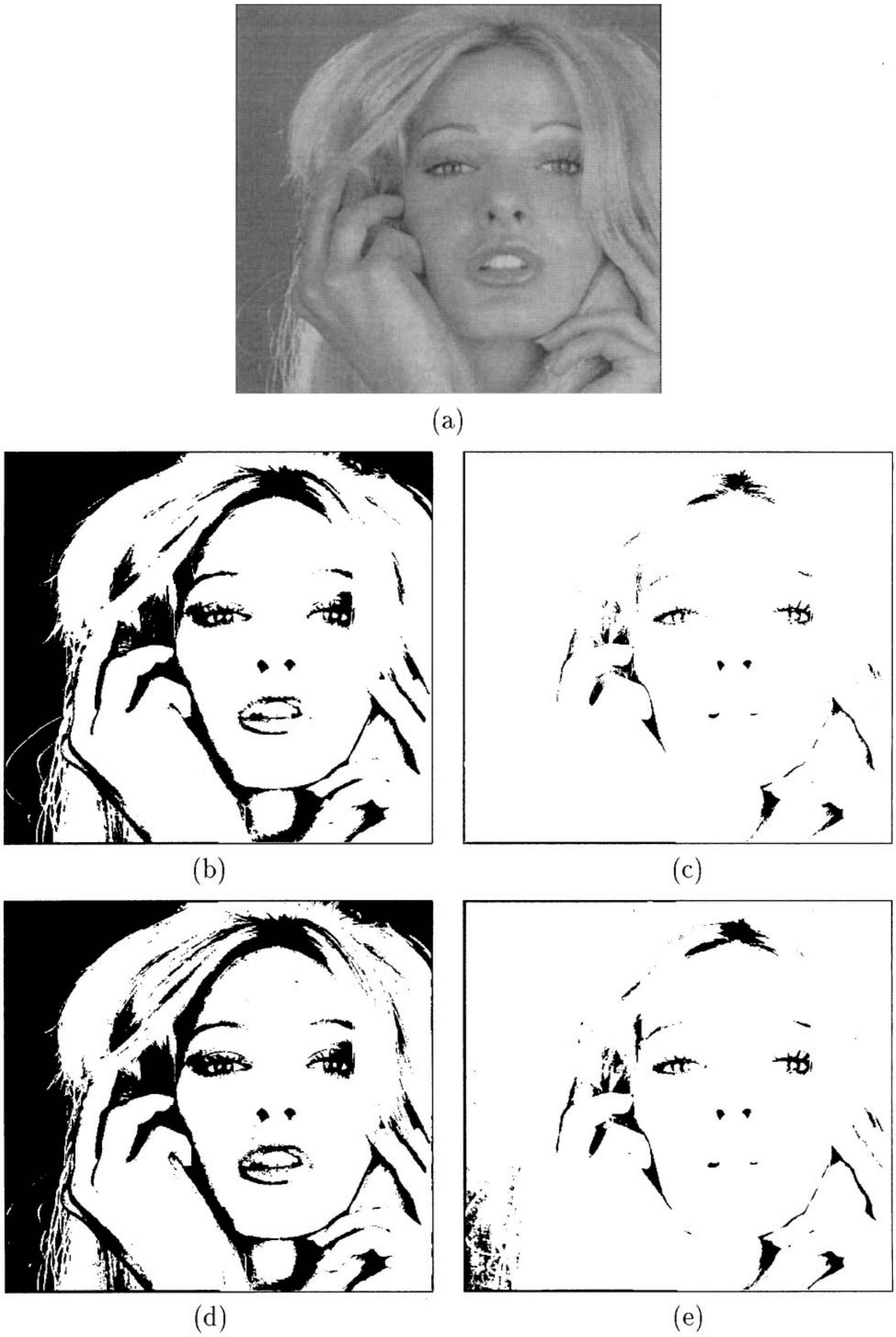


Fig. 12. 2D thresholding on image "tiffany": tiffany: (a) original image; (b) dark extraction by 2D fuzzy approach with threshold vector $(t, t) = (187, 187)$; (c) dark extraction by 2D crisp approach with threshold vector $(s, t) = (160, 151)$; (d) bright extraction by 2D fuzzy approach with threshold vector $(t, t) = (187, 187)$; (e) bright extraction by 2D crisp approach with threshold vector $(s, t) = (160, 151)$.

Table 3
Computational time (in seconds) required to calculate entropies for a given (a, b, c)

	Preprocess	Eq. (12)	Eq. (13)	Eq. (14)	Eq. (15)	Total
No moment	0	0.252	0.052	0.247	0.055	0.606
Moment	0.833	0.001	0.000	0.001	0.000	0.835

performed by dark extraction and bright extraction experiments. Figs. 9–12 show the comparisons where part a is the original image, parts b and c are the results of dark extraction using the proposed approach and the 2D nonfuzzy approach, respectively, and parts d and e are the results of bright extraction using the proposed approach and the 2D nonfuzzy approach, respectively.The operations of dark extraction and bright extraction are explained in Eqs. (18) and (17).

1. Image "masuda2" in Fig. 9: The results are obvious because nothing can be identified in the images in Fig. 9c and e. It is even hard to tell what the images are.
2. Image "peppers" in Fig. 10: The peppers can be seen clearly in images in Fig. 10b and d, but some of them cannot be identified in the compared images in Fig. 10c and e. The rugged surface of peppers has been clearly expressed on the front pepper in images in Fig. 10b and d.
3. Image "soccer" in Fig. 11: The comparison results are obvious. Images in Fig. 11b and d show that two soccer players are in the field and the audience is in the background. In the images in Fig. 11c and e, only the soccer ball can be clearly recognized.
4. Image "tiffany" in Fig. 12: This is quite a bright image. In the images in Fig. 12b and d, the bright object has been successfully extracted from the dark back ground. In the images in Fig. 12c and 6e, the dark background has been misclassified to be bright gray level. Therefore, the hair and hands of the object have disappeared.

The proposed approach is robust for either the dark images, such as "masuda2," and "soccer" or the bright image, "tiffany." From the above examples, our threshold vectors can generate better binary images than those obtained by using 2D crisp partition method. Therefore, the proposed 2D fuzzy approach is better than 2D nonfuzzy approach.

5.3. Computation time comparison

In order to verify that the moment array can reduce the computational time of entropy evaluation, the entropy evaluation is executed in two ways. One uses the moment array to calculate the entropy, and the other uses the original formula to calculate the entropy without the help of moment array. Table 3 shows the time required to calculate the four entropies for a given (a, b, c). The processing time for moment array is shown in the first column. The programs were run on a HP 9000/710 and the computing time was measured in seconds.

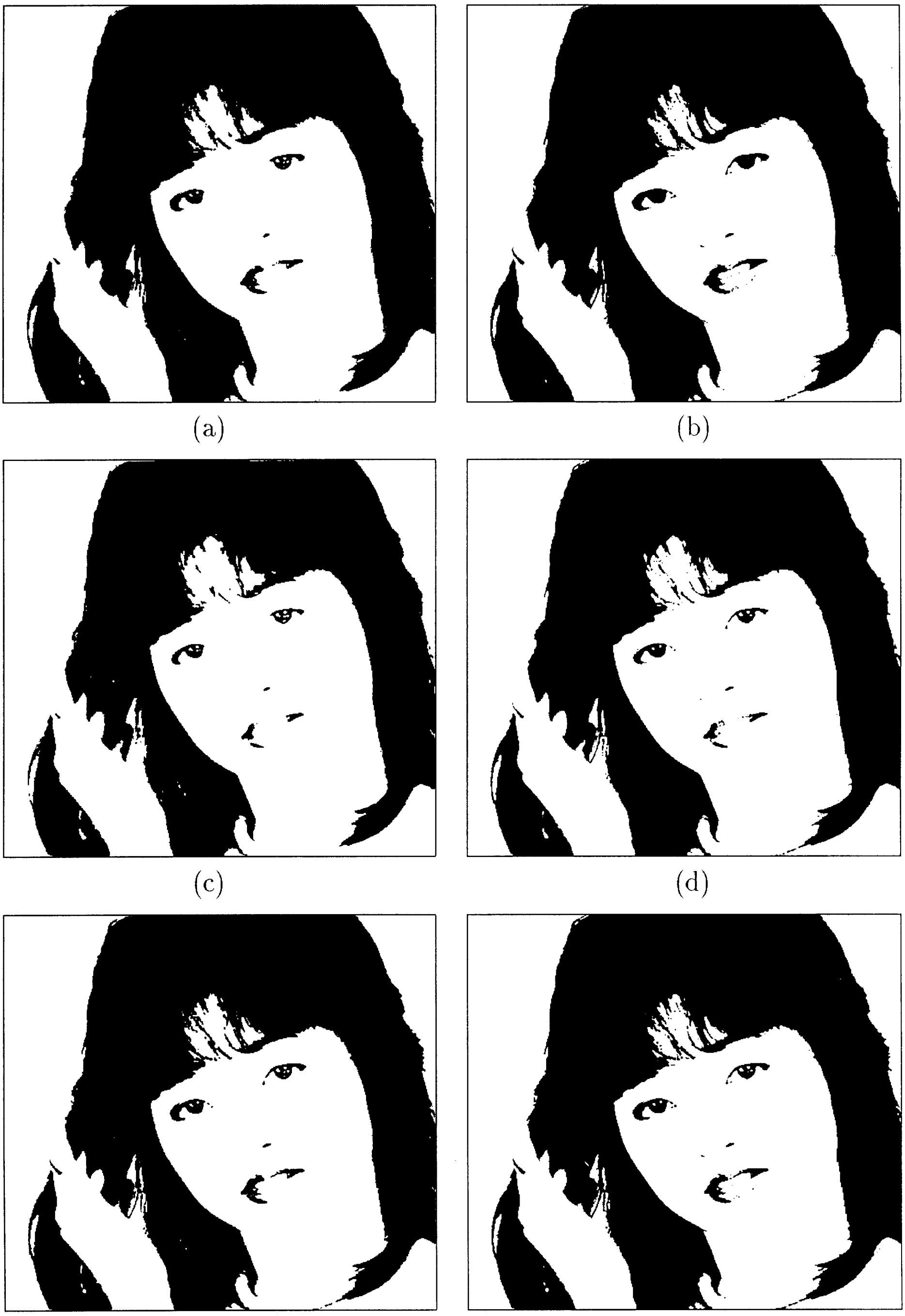
For the program not using the moment array, the computation time is shown in the first row of Table 3. The time needed to calculate two fuzzy entropies, Eqs. (12) and (14), is 0.252 and 0.247 s, respectively. The time needed to calculate two nonfuzzy entropies, Eqs. (13) and (15), is 0.052 and 0.055 s, respectively. Fuzzy entropies require more time than nonfuzzy entropies, because they

Table 4
Computational time (in s) for image entropy evaluation based on different generation numbers with population size 100

Number of generations	1	10	100
No moment	55.833	502.5	4973.333
Moment	1.467	5.333	31.333

Table 5
The relation between window size and corresponding threshold

	"masuda2"		"tiffany"		"peppers"	
d	Entropy	t	Entropy	t	Entropy	t
3	33.784	74	33.526	187	33.933	120
5	33.843	66	33.512	182	33.841	122
7	33.893	75	33.502	183	33.817	121
9	33.942	67	33.481	182	33.785	121
11	33.991	70	33.458	182	33.764	121
13	34.041	68	33.445	183	33.752	122
15	34.088	67	33.431	189	33.739	123
17	34.131	69	33.354	189	33.738	122
19	34.183	67	33.405	182	33.731	121
21	34.229	69	33.396	182	33.734	122
23	34.275	69	33.380	182	33.738	123
25	34.319	71	33.380	182	33.750	121
27	34.371	70	33.371	181	33.772	121
29	34.427	68	33.370	182	33.794	121
31	34.487	69	33.361	182	33.813	122
33	34.534	70	33.376	181	33.829	121
35	34.592	69	33.374	182	33.849	122
37	34.637	72	33.375	182	33.872	121
39	34.692	70	33.379	181	33.893	121
41	34.743	70	33.373	181	33.913	123



(a) (b)

(c) (d)

(e) (f)

Fig. 13. 2D thresholding with different window size d: (a) dark extraction by $d = 7$ with threshold vector $(t, t) = (75, 75)$; (b) bright extraction by $d = 7$ with threshold vector $(t, t) = (75, 75)$; (c) dark extraction by $d = 5$ with threshold vector $(t, t) = (66, 66)$; (d) bright extraction by $d = 5$ with threshold vector $(t, t) = (66, 66)$; (e) dark extraction by $d = 3$ with threshold vector $(t, t) = (74, 74)$; (f) bright extraction by $d = 3$ with threshold vector $(t, t) = (74, 74)$.

need to do extra multiplication operations. The total time for calculating four entropies is 0.606 s.

For the program using the moment array, the computation time is shown in the second row of Table 3. It needs to set up two moment arrays before evaluating the entropy. The task requires going through two 256×256 arrays to add the elements which takes 0.833 s. For evaluating fuzzy entropies, it needs 0.001 s for each fuzzy entropy. For evaluating nonfuzzy entropies, it needs only a constant number of subtractions and additions, which need approximately zero computing time. That yields a total 0.835 s.

From the results, the moment array approach needs more time to evaluate the four entropies for a given (a, b, c). However, the computation of moment arrays is required only once in the entire execution of GA, the time saving is seen by increasing the population size and generation number of GA. Let the population size of GA be 100. To produce the next generation, the GA will do the image entropy evaluation 100 times and some operations including selection, crossover, mutation, etc., which are required for running the GA. Table 4 shows the computation time in seconds for different generation numbers. From Table 4, a conspicuous time saving occurs when the number of generations becomes large. The moment array approach really saves computational time compared with the original calculation method.

5.4. Window size determination

A 2D histogram is the extension of a 1D histogram just by adding a second dimension local average. The purpose of the second dimension is for collecting the local information among pixels. In 2D histogram, we use the average of four neighbors to describe the spatial information. Actually, other lowpass filters can be utilized. In the following experiment, we analyze the effects of window size on the determination of threshold. The filters with different sizes may result different 2D histograms which may have different entropies and thresholds. Therefore, we should decide what size filter should be used to acquire the 2D histogram. The hypothesis is based on the maximal entropy principle, i.e. the optimal window size should maximize the image entropy.

The window size of a 3×3 filter is denoted by $d = 3$. We tried many filters with different window sizes to compute the corresponding 2D histograms, and the maximal entropies and optimal thresholds for each 2D histogram. The window size d is set at an odd number for symmetry. the range of d is from 3 to 41. When the filter is shifted out of the image boundary, the image is viewed as wrap-around rows (or columns).

The experiment was done on many images, and only a portion of the results is shown in Table 5. By investigating the data in the table, we can conclude that the window size does change the values of the image entropy without a definite pattern. With the increasing d values, image "masuda2" has the increasing entropy values while image "kids" has the decreasing entropy values; the entropies of image "peppers" decreased first then increased after $d = 19$, while the entropies of image "tiffany" have an unpredictable vibration.

Note that window size d has little influence on the determination of the threshold. All the thresholds, with respect to various d values, are almost the same for each of the tested images. The threshold vibration ranges for the four images "masuda2", "tiffany", and "peppers" are $[66 \cdots 75]$, $[181 \cdots 189]$, and $[120 \cdots 123]$, respectively. Image "masuda2" has the maximal vibration range 9. The minimal threshold value 66 occurred at $d = 5$ while the maximal threshold value 75 occurred at $d = 7$. By comparing the corresponding thresholded images with that from $d = 3$, the results, shown in Fig. 13, do not have big difference. Therefore, in most cases the filter with window size 3 should be used by the proposed approach to select the threshold for its simplicity and adequate accuracy.

6. Conclusions

An automatic thresholding of gray-level pictures using 2D fuzzy entropy is proposed. The proposed approach is based on the 2D histogram of images, which considers both the local information as well as the pixel intensity.

A novel 2D fuzzy partition characterized by parameters a, b, and c is proposed to partition a 2D histogram into two fuzzy subsets "dark" and "bright". For each fuzzy subset, one fuzzy entropy and one nonfuzzy entropy are defined based on the fuzziness of the regions. Thus, the image's entropy is obtained by summing up the four subentropies. The best fuzzy partition was found by a genetic algorithm based on the maximal entropy principle, and the corresponding parameters a, b, and c determines the fuzzy region $[a, c]$. Then, the threshold is selected as the crossover point from the fuzzy region. The membership assignment for the elements in the 2D histogram is based on the fuzzy relation of fuzzy sets, which is actually the intersection of the fuzzy sets. The moment array is introduced to help the evaluation of entropies. The key is to precompute and save the values required for the evaluation of image entropies. The computation expense for total entropy can be reduced from $O(8N^2)$ to $O(2N)$.

The experiments prove that a filter with window size 3 is enough to collect the local information among pixels. The obtained threshold can be applied to perform image thresholding. The thresholding results are compared with those from 1D fuzzy approach and 2D nonfuzzy approach. The experimental results prove that the spatial information of pixels should be considered in the selection of thresholds and the 2D fuzzy approach outperforms 2D crisp approach and 1D fuzzy approach. The proposed approach can have wide application in computer vision and image processing.

References

[1] J.S. Weszka, A survey of threshold selection techniques, Comput. Vision Graphics Image Process 7 (1978) 259–265.

[2] S.K. Fu, J.K. Mu, A survey on image segmentation, Pattern Recognition, 13 (1981) 3–16.

[3] P.K. Sahoo, S. Soltani, A.K.C. Wong, Y. Chen, A survey of thresholding techniques, Computer Vision Graphics and Image Process. 41 (1988) 233–260.

[4] L.A. Zadeh, Probability Measures of fuzzy events, J. Math. Anal. Appl., 23 (1968) 421–427.

[5] S.K. Pal, D.K.D. Majumder, Fuzzy Mathematical Approach to Pattern Recognition, J Wiley, New York, 1986.

[6] C.A. Murthy, S.K. Pal, Histogram thresholding by minimizing graylevel fuzziness, Inform. Sciences 60 (1992) 107–135.

[7] T.J. Ross, Fuzzy Logic with Engineering Application, McGraw–Hill, New York, 1995.

[8] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addison–Wesley, Reading, MA, 1993.

[9] A.D. Luca, S. Termini, A definition of nonprobabilistic entropy in the setting of fuzzy set theory, Inform. Control 20 (1972) 301–312.

[10] A. Kaufmann, Introduction to the Theory of Fuzzy Subsets—Fundamental Theoretical Elements, I. Academic Press, New York, 1980.

[11] D.E. Goldberg, Genetic Algorithm in Searching, Optimization, and Machine Learning. Addison–Wesley, Reading, MA, 1989.

[12] W.T. Chen, C.H. Wen, C.W. Yang, A fast two–dimensional entropic thresholding algorithm, Pattern Recognition 27 (1994) 885–893.

[13] X. Wu, image coding by adaptive tree–structured segmentation, IEEE Trans. Inform. Theory 38 (1992) 1755–1767.

[14] H.D. Cheng, J.R. Chen, Threshold selection based on fuzzy *c*–partition entropy approach, Pattern Recognition accepted.

[15] A.S. Abutaleb, automatic thresholding of gray–level pictures using two–dimensional entropy, Comput. Vision Gaphics Image Process 47 (1989) 22–32.

Aboust the Author—HENG-DA CHENG received Ph.D. degree in Electrical Engineering from Purdue University, West Lafayette, in 1985. Dr. Cheng was a Visiting Assistant Professor, Electrical and Computer Engineering Department, University of California, Davis, an Assistant Professor, Computer Science Department, Concordia University, Montreal, and an Associate Professor, School of Computer Science, Technical University of Nova Scotia, Halifax, Nova Scotia. Now he is an Associate Professor, Department of Computer Science, and an Adjunct Associate Professor, Department of Electrical Engineering, Utah State University, Logan, Utah.

Dr. Cheng has published over 160 technical papers and is the co-editor of the book, *Pattern Reognition: Alghorithms, Architectures and Applications* (World Scientific Publishing Co., 1991). His research interests include parallel processing, parallel algorithms, artificial intelligence, image processing, pattern recognitions, computer vision, fuzzy logic, genertic algorithms, neural networks and VLSI architectures.

Dr. Cheng is the chairman of the First International Workshop on Computer Vision, Pattern Recognition and Image Processing (CVRIP'98), 1998, was Program Co-Chairman of Vision Interface '90, Program Committe Member of Vision Interface '92 and Vision Interface '90. Session Chairs and Member of Best Paper Award Evaluation Committee of the International Joint Conference on Information Scinces, 1994 and 1995, Program Committee Member of the 1995 International Conference on Tools with Artificial Intelligence and Processing of Oriental Languages, 1997. He serves as reviewer for many scientifical journals and conferences.

Dr. Cheng has been listed in *Who's Who in the World. Who's in America*, *Who's Who in Communications and Media*, *Who's Who in Finance and Industry*, *Who's who is Science and Enginering*, *Men of Archievement*, 2000 *Notable American Men*, *International Leaders in Achievement*, *Five Hundred Leadrs of Influence*, *International Dictionary of Distinguished Leadership*, etc.

Dr. Cheng is a Senior Member of the IEEE Society, and a Membeer of the Association of Computing Machinery. Dr. Cheng is also an Associate of Pattern Recognition and an Associate Editor of Information Sciences.

About the Author—YEN-HUNG CHEN was born in Tainan, Taiwan, 1969. He received the B.S. degree in Mathematics from National Chunghua University, Chunghua, Taiwan, in 1990, and the M.S. degree in Computer Science from Utah State University, Logan, Utah, in 1997. Besides the master thesis, he has two journal papers and two conferrence papers published so far. His research interests include Fuzzy Logic, Genetic Algorithm, Computer Vision, and Pattern Recognition.

He has been with Chunghwa Telecom Co., Ltd., Taipei, Taiwan, since 1997.