Signal Processing 75 (1999) 277–301

A novel fuzzy entropy approach to image enhancement and thresholding

H.D. Cheng*, Yen-Hung Chen, Ying Sun

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

Received 27 August 1997; received in revised form 18 November 1998

Abstract

Image processing has to deal with many ambiguous situations. Fuzzy set theory is a useful mathematical tool for handling the ambiguity or uncertainty. In order to apply the fuzzy theory, selecting the fuzzy region of membership function is a fundamental and important task. Most researchers use a predetermined window approach which has inherent problems. There are several formulas for computing the entropy of a fuzzy set. In order to overcome the weakness of the existing entropy formulas, this paper defines a new approach to fuzzy entropy and uses it to automatically select the fuzzy region of membership function so that an image is able to be transformed into fuzzy domain with maximum fuzzy entropy. The procedure for finding the optimal combination of a, b and c is implemented by a genetic algorithm. The proposed method selects the fuzzy region according to the nature of the input image, determines the fuzzy region of membership function automatically, and the post-processes are based on the fuzzy region and membership function. We have employed the newly proposed approach to perform image enhancement and thresholding, and obtained satisfactory results.

Zusammenfassung

Die Bildverarbeitung muß sich mit vielen mehrdeutigen Sachverhalten auseinandersetzen. Die Theorie der unscharfen Mengen ist ein nützliches mathematisches Werkzeug im Umgang mit der Mehrdeutigkeit oder Unsicherheit. Für die Anwendung der Fuzzy-Theorie stellt die Auswahl des Fuzzy-Bereichs der Zugehörigkeitsfunktion eine grundlegende und wichtige Aufgabe dar. Die meisten Forscher verwenden einen Ansatz mit vorausbestimmtem Fenster, der inhärente Probleme aufweist. Es gibt mehrere Formeln zur Berechnung der Entropie einer unscharfen Menge. Um die Schwächen der herkömmlichen Entropieformeln zu überwinden, definiert dieser Aufsatz einen neuen Ansatz der Fuzzy-Entropie und benutzt ihn, um den Fuzzy-Bereich der Zugehörigkeitsfunktion automatisch auszuwählen, so daß ein Bild in ein Fuzzy-Gebiet mit maximaler Fuzzy-Entropie transformiert werden kann. Das Verfahren zum Auffinden der optimalen Kombination von a, b und c wird mit einem genetischen Algorithmus implementiert. Die vorgeschlagene Methode wählt den Fuzzy-Bereich gemäß der Natur des Eingangsbildes aus, bestimmt automatisch den Fuzzy-Bereich der Zugehörigkeitsfunktion, und die Nachverarbeitung baut auf den Fuzzy-Bereich und die Zugehörigkeitsfunktion auf. Wir haben den neu vorgeschlagenen Ansatz zur Bildverbesserung und -begrenzung angewendet und haben dabei zufriedenstellende Ergebnisse erzielt.

* Corresponding author. Tel.: + 1-801-797-2451; fax: + 1-801-797-3265; e-mail: cheng@hengda.cs.usu.edu

PII: S0165-1684(98)00239-4

Résumé

Le traitement des images doit faire face à de nombreuses situations ambiguës. La théorie des ensembles flous est un outil mathématique utile pour manipuler l'ambiguïté et l'incertitude. Afin d'appliquer la théorie floue, la sélection de la région floue de la fonction d'appartenance est une tâche fondamentale et importante. La plupart des chercheurs utilisent une approche par fenêtres prédéterminées, qui présente des problèmes inhérents. Il existe plusieurs formules pour le calcul de l'entropie d'un ensemble flou. Afin de dépasser les faiblesses des formules d'entropie existantes, cet article définit une nouvelle approche de l'entropie floue et l'utilise pour sélectionner automatiquement la région floue de la fonction d'appartenance, de sorte qu'il est possible de transformer une image en un domaine flou avec une entropie floue maximale. La procédure de recherche de la combinaison optimale de a, b et c est mise en œuvre par algorithme génétique. La méthode proposée sélectionne de région floue selon la nature de l'image d'entrée, détermine automatiquement la région floue de la fonction d'appartenance, et les post-traitements reposent sur la région floue et la fonction d'appartenance. Nous avons utilisé cette nouvelle approche pour effectuer du rehaussement et du seuillage d'images, et nous avons obtenu des résultats satisfaisants.

Keywords: Fuzzy logic; Fuzzy entropy; Fuzzy membership; Genetic algorithm; Image enhancement; Thresholding

1. Introduction

In the field of image processing, people have to deal with many ambiguous situations. Ambiguity caused by projecting a 3-D object into a 2-D image or digitizing analog pictures into digital images, and the uncertainty related to boundaries and non-homogeneous regions are very common. Some of definitions, such as edges, contrast, enhancement, etc., are fuzzy as well. Fuzzy set theory is a useful mathematical tool for handling the ambiguity or uncertainty.

However, selecting the fuzzy region of membership function is a fundamental and important task. In order to apply the fuzzy theory, most researchers use a predetermined approach. For instance, multi-level segmentation is determined by many local maxima of fuzzy entropy with a predetermined window (fuzzy region) which is shifted along the histogram [8,9,11,12]. Basically, the predetermined window approach has some inherent problems. First, there is no theoretical evidence to explain why the fuzzy region should be set like that. Second, when the range of intensity is large, it cannot effectively select the fuzzy region. The selection of fuzzy region should depend on the nature of the images. That is, different images with different characteristics need to use fuzzy regions with different sizes.

The purpose of this work is to automatically determine the fuzzy region based on the maximum fuzzy entropy principle and apply it to find the membership function. The search of the best combination of a, b and c of the membership function is conducted using a genetic algorithm. Then, we process the fuzzified images for enhancement and thresholding.

2. Fuzzy theory and maximum entropy principle

In this section, we give brief descriptions about the fuzzy set theory and entropy which is used as a measurement of fuzziness of a fuzzy set. A new fuzzy entropy definition used in our experiments will be introduced as well.

2.1. Classical set versus fuzzy set

The classical *set A* defined in the algebra is a collection of objects. Under this definition, each element x in the universal is either in the set or not. Therefore, the membership $\mu_A(x)$ is 1 for those elements in the set ($x \in A$) and 0 for those out of the set ($x \notin A$).

Generally, a *fuzzy set* is defined as a collection of ordered pairs and can be expressed by the following notation [12]:

$$A = \{(\mu_A(x_i), x_i) \mid i = 1,2,\dots,N\},$$

where $\mu_A(x_i)$ is the membership function that maps x_i to the *fuzzy domain* [0,1] and N is the number of elements in the set. The value indicates the degree of the elements belong to the fuzzy set. Larger values denote higher degrees of the memberships. From the aspect of mathematics, the major difference between the classical set and the fuzzy set is the value of membership function $\mu(\ )$ as shown in Table 1.

For example, a fuzzy set which describes "integers approximately equal to 0" can be defined as

$$\begin{aligned}\text{Near} &= \{(\mu_{\text{Near}}(\text{integer}),\text{integer})\}\\ &= \{(0.1,-3),(0.4,-2),(0.7,-1),(1.0,0),\\ &\quad (0.7,1),(0.4,2),(0.1,3)\}.\end{aligned}$$

The closer to 0 the integer is, the higher membership it has. Because integer 1 is closer to 0 than integer 3, $\mu_{\text{Near}}(1)$ is larger than $\mu_{\text{Near}}(3)$ ($0.7 > 0.1$, in this example). $\mu_{\text{Near}}(0) = 1.0$ means 0 is the integer closest to 0. The elements with zero membership can be ignored in the presentation.

Another fuzzy set example is to describe "the brightness of an image". Assume that an image has 8 $(0,\dots,7)$ intensity scales where 0 means dark and 7 means bright. We can set the membership of pixels from 0 increasingly to 1 according to its gray level from 0 to 7. Such as:

$$\begin{aligned}\text{Bright} &= \{(\mu_{\text{Bright}}(\text{graylevel}),\text{graylevel})|\\ &\qquad \text{graylevel} = 0,\dots,7\}\\ &= \{(0,0),(0.1,1),(0.2,2),(0.4,3),(0.6,4),\\ &\quad (0.8,5),(0.9,6),(1,7)\}.\end{aligned}$$

The selection of membership function is dependent on the applications. The S-function and π-function are most commonly used. Other membership functions can be found in [2,16]. In our experiments, we choose the S-function as the membership function. It is defined as

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

where x is the variable in the intensity domain (usually, it is the gray level between 0 and 255), and a, b and c are the parameters which determine the shape of S-function. The range $[a,c]$ is called the *fuzzy region*, where b is usually set as the mid-point of $[a,c]$, but it is not necessary. If b is defined as the mid-point between a and c, then it is called the *standard S-function*. We use S-function as the membership function to transform an image from the intensity domain into the fuzzy domain for further process, such as thresholding, segmentation, etc.

2.2. Maximum entropy principle

One of the interesting point about transforming an image from the intensity domain into the fuzzy domain is how much information can it keep. According to the *information theory* [3], *entropy* can be defined as

$$H(A) = -\sum_{i=1}^{N} P(x_i)\log P(x_i), \tag{2}$$

where

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

and $x_i, i = 1,\dots,N$, are the possible outputs from source A with the probability $P(x_i)$. The larger entropy $H(A)$ is, the more information A has. Using Lagrange multiplier method, we can get the following equation:

$$H(A) = -\sum_{i=1}^{N} P(x_i)\log P(x_i) + \mu\left(\sum_{i=1}^{N} P(x_i) - 1\right). \tag{3}$$

The maximum of $H(A)$ can be found by letting

$$\frac{\partial H(A)}{\partial P(x_i)} = -\log P(x_i) - 1 + \mu = 0. \tag{4}$$

Table 1
Membership comparison between classical set and fuzzy set

	Classical set A	Fuzzy set A
$\mu_A(x)$ type	integer	real
$\mu_A(x)$ range	$\mu_A(x) \in \{0,1\}$	$\mu_A(x) \in [0,1]$

From Eq. (4), we obtain $P(x_i) = e^{\mu - 1}$ for $i = 1, \dots, N$, i.e., $H(A)$ will reach the maximum when $P(x_1) = P(x_2) = \cdots = P(x_N) = 1/N$. In this case, source A has the maximal capacity of information. In image processing, an image can be viewed as an information source A with the intensities as the possible outputs. The histogram distribution can be viewed as probability $P(x_i)$.

Basically, Eq. (2) only expresses the entropy of a distribution P in the space (intensity) domain. What we need is a measurement of information amount when an image is in the fuzzy domain. Therefore, we can keep the information as much as we can while an image is transformed from the intensity domain into the fuzzy domain by the membership function $\mu(\ )$.

Zadeh [15] suggested a definition about the entropy of a fuzzy set which takes both distribution and membership into consideration. It is defined as follows:

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

where A is a fuzzy set $\{(\mu_A(x_i), x_i) | i = 1, \dots, N\}$ with respect to the distribution $P(x_i)$.

Kaufmann [5] defined the entropy of a fuzzy set A as

$$H(\varphi_A(x_1), \varphi_A(x_2), \dots, \varphi_A(x_N)) = \frac{-1}{\ln(N)} \sum_{i=1}^{N} \varphi_A(x_i) \ln\{\varphi_A(x_i)\}, \tag{6}$$

where $\varphi_A(x_i)$ is defined as

$$\varphi_A(x_i) = \frac{\mu_A(x_i)}{\sum_{i=1}^{N} \mu_A(x_i)}.$$

He also considers both the distribution and membership but takes the distribution in the fuzzy domain $\varphi_A(x_i)$ instead of that in the ordinary domain $P(x_i)$.

De Luca and Termini [1] proposed a quite different definition about the entropy of a fuzzy set A. This entropy has a nonprobabilistic feature and is defined as

$$H(A) = \frac{1}{N \ln 2} \sum_{i=1}^{N} S_n(\mu_A(x_i)), \tag{7}$$

where $S_n(\ )$ is the Shannon's function [14] as follows:

$$S_n(\mu_A(x_i)) = -\mu_A(x_i) \ln(\mu_A(x_i)) - (1 - \mu_A(x_i)) \ln(1 - \mu_A(x_i)). \tag{8}$$

Actually, each of the above fuzzy entropy definitions has some problems when using these fuzzy entropies to determine the fuzzy region. A detailed discussion about these problems is given in Section 6.

2.3. A new entropy definition of a fuzzy set

Here we propose a new formula for calculating the entropy of a fuzzy set. Basically, the entropy of a fuzzy set should be not only related to the membership $\mu_A(\ )$ of each element, but also related to the distribution $P(\ )$, therefore, it can respond to the variety of input information.

Definition 1. Let A be the fuzzy event, and N be the number of partitions of A in the fuzzy domain denoted as sub-events $A_1, \dots, A_N$. The fuzzy domain partition method $\mathscr{M}$ is task-dependent which may be equal partition or non-equal partition. $\mu_A(x)$ is the membership function and $P(x)$ is the probability of x in the space domain. The entropy of a fuzzy event is defined as

$$H(A, N, \mathscr{M}, \mu_A) = \frac{-1}{\log N} \sum_{i=1}^{N} P_P(A_i) \log P_P(A_i), \tag{9}$$

where

$$P_P(A_i) = \sum_{\mu_A(x) \in A_i} P(x).$$

$P_P(A_i)$ means the probability summed in the space domain for the x (space domain) mapping into A_i (fuzzy domain) by the membership function $\mu_A(\ )$.

At the first glance, Eq. (9) is similar to Eq. (2) except for the $\log N$ in the denominator which serves as a normalizer. Actually, the probability computation is not exactly the same. The definition in Eq. (9) combines the membership $\mu(\ )$ and probability $P(\ )$ of Eq. (2) into a single term $P_P(\ )$ which stands for partial probability. Notice that $P(\ )$ in Eq. (2) represents the probability over the space domain but $P_P(\ )$ in Eq. (9) represents the probability sum for those x (space domain) mapping into A_i (fuzzy domain) by the membership function $\mu_A(\ )$. Therefore, $P_P(A_i)$ can be viewed as the probability of fuzzy events $A_i, i = 1,\dots,N$, based on the membership function $\mu_A(\ )$. Different membership functions will cause different values of $P_P(A_i)$. After calculating all the values of $P_P(A_i)$, the entropy of a fuzzy set can be calculated simply by Eq. (2).

If we choose the S-function as the membership function, once the number of categories N and the partition method $\mathscr{M}$ have been decided, $P_P(\ )$ will be dependent only on the parameters of S-function (a,b,c). Therefore, Eq. (9) can be rewritten as

$$H(A,a,b,c) = \frac{-1}{\log N} \sum_{i=1}^{N} P_P(A_i,a,b,c) \log P_P(A_i,a,b,c). \tag{10}$$

For example, suppose there is an image with eight gray levels $(0,1,\dots,7)$ and its histogram (probability) is shown in Table 2. If we want to classify this image into three categories $(A_1,A_2,A_3) =$ (dark,middle,bright), i.e. $N = 3$. by using equal partition as the partition method $\mathscr{M}$. We can divide interval $[0,1]$ into three equal sub-intervals $[0,\frac{1}{3}]$, $[\frac{1}{3},\frac{2}{3}]$ and $[\frac{2}{3},1]$ to represent A_1, A_2 and A_3, respectively.

In case one, the membership function based on $(a1,b1,c1)$ classifies gray level 0 and 1 into "dark" because their memberships $\mu_A(0) = 0.15$ and $\mu_A(1) = 0.30$ fall into the sub-interval $A_1 = [0,\frac{1}{3}]$ which means dark. In the same way, gray level 2 and 3 belong to "middle" and gray level 4, 5, 6 and 7 belong to "bright". Therefore,

$$P_P(A_1,a1,b1,c1) = 0.05 + 0.15 = 0.20,$$

$$P_P(A_2,a1,b1,c1) = 0.15 + 0.20 = 0.35,$$

$$P_P(A_3,a1,b1,c1) = 0.25 + 0.15 + 0.05 = 0.45,$$

and the entropy is gained from Eq. (10),

$$\begin{aligned} H(A,a1,b1,c1) &= \frac{-1}{\log 3} \sum_{i=1}^{3} P_P(A_i,a1,b1,c1) \log P_P(A_i,a1,b1,c1) \\ &= \frac{-1}{\log 3}(0.20\log(0.20) + 0.35\log(0.35) \\ &\quad + 0.45\log(0.45)) \\ &= \frac{-1}{\log 3}(-0.1397940008672 - 0.1595761844774 \\ &\quad - 0.1560543688011) \\ &= 0.9545258142243. \end{aligned}$$

In case two, the membership function based on $(a2,b2,c2)$ classifies gray level 0 into "dark", 1, 2, 3, 4 and 5 into "middle", 6 and 7 into "bright" with $P_P(A_1,a2,b2,c2) = 0.05$, $P_P(A_2,a2,b2,c2) = 0.90$ and $P_P(A_3,a2,b2,c2) = 0.05$. Therefore, its entropy is

$$\begin{aligned} H(A,a2,b2,c2) &= \frac{-1}{\log 3} \sum_{i=1}^{3} P_P(A_i,a2,b2,c2) \log P_P(A_i,a2,b2,c2) \\ &= \frac{-1}{\log 3}(0.05\log(0.05) \\ &\quad + 0.90\log(0.90) + 0.05\log(0.05)) \\ &= \frac{-1}{\log 3}(-0.0650514997832 \\ &\quad - 0.04118174150461 - 0.0650514997832) \\ &= 0.3589962496444. \end{aligned}$$

In this example, $H(A,a1,b1,c1) > H(A,a2,b2,c2)$ means that the image can keep more information while using S-function with parameters $(a1,b1,c1)$ as

Table 2
A sample image with eight gray levels and two different membership functions

Gray level x	0	1	2	3	4	5	6	7
Histogram	0.05	0.15	0.15	0.20	0.25	0.15	0.05	0.0
$\mu_A(x) \propto (a1,b1,c1)$	0.15	0.30	0.49	0.64	0.69	0.79	0.88	0.95
$\mu_A(x) \propto (a2,b2,c2)$	0.28	0.35	0.42	0.52	0.59	0.65	0.77	0.89

the membership function to transform the image into fuzzy domain.

In our definition of fuzzy entropy, there are two settings that are task-dependent. One is the number of partitions N, and the other is the partition method $\mathscr{M}$. In image processing, an image can be just classified into two levels (dark and bright) or multiple levels (dark, middle-dark, middle, middle-bright, bright). In other case like measuring the speed, the classification can be as simple as two levels (fast and slow) or as complicated as five levels of clutch shifting in the car driving. About the partition method $\mathscr{M}$, we use equal partition as the example just for convenience, but it is not necessary. It can be task-dependent and user-specified.

The problem is to find a combination of (a,b,c) such that $H(A,a,b,c)$ has the maximum value. There are some heuristic methods [7], neural networks [16], simulated annealing [6], and genetic algorithm [2,10] that can be used to solve such problem.

3. Genetic algorithm

Genetic algorithm (GA) maintains a set of possible solutions which are encoded as chromosomes. Like physiological reproduction, the algorithm generates next generations by crossovers and mutations. The crossover strategy involves the elitist model which significantly improves the performance on the unimodal surface problems.

For a genetic algorithm, several parameters need to be defined [2,10]. They are described below.

1. *Coding method*. Coding method represents the problem parameters as a finite string over some alphabet (usually 0 and 1). These finite strings are called chromosomes. Each chromosome represents a solution to the problem.
2. *Object function*. Object function is a fitness measure on the solution represented by each chromosome. Its value tells how well the chromosome satisfied the final goal.
3. *MaxGen* (Maximal generation number). Maximal generation number is set as a limit on the generations. Generally, GA will not stop until the number of generations reaches the limit of the generation number.
4. *POPs* (Population size). Population size indicates the number of chromosomes in each generation.
5. *Pc* and *Pm*. Probability of crossover and probability of mutation are real numbers between 0 and 1. Just like the probability of human generation, *Pc* represents the probability with which two chromosomes hybridize while *Pm* represents the mutation probability for each chromosome's element (0 changes to 1 and 1 changes to 0).

After all these parameters are defined, GA can start the operations and find the solution as follows:

```
begin /* Main program */
randomly initialize the OLDPOP;
calculate object function for each chromosome
  in OLDPOP;
for (gen = 0; gen < MaxGen; gen++)
begin /* Iterative loop */
  generate NEWPOP from OLDPOP by
      crossover and mutation;
  calculate object function for each
      chromosome in NEWPOP;
  keep the best chromosome so far;
  OLDPOP = NEWPOP;
end /* Iterative loop */
end /* Main program */
```

The generation of new population selects the offspring strings from the old population using a roulette wheel weighted by the fitness values. Therefore, the strings with higher fitness values will be selected more frequently to generate the next generation. For each generation, if the coming random number is below *Pc*, the *crossover* operation should be performed. There are many ways to perform chromosome's (string's) crossover. The most common way is to cut each of the two parent chromosomes into two parts, then connect the head part of one chromosome to the other's tail part [2]. For example, consider strings A_1, A_2:

$$A_1 = 0\ 1\ 1\ 0 | 1\ 1\ 0,$$

$$A_2 = 1\ 1\ 0\ 1 | 0\ 1\ 1.$$

If the crossover point happens at 4 (a randomly generated number less than the string length), the new strings will be

$A'_1 = 0\ 1\ 1\ 0\ 0\ 1\ 1,$

$A'_2 = 1\ 1\ 0\ 1\ 1\ 1\ 0.$

The *mutation* operation will be performed when the coming random number is below Pm. Then the next random number generated less or equal to the string length will indicate the mutation position where 0 mutates to 1 or 1 mutates to 0.

4. Determine the parameters for S-function

In this section, we will explain how to use genetic algorithm to find a combination of (a,b,c) such that $H(A,a,b,c)$ has the maximum value.

4.1. Coding method

The first step is to encode the parameters a, b and c into an alphabet string. Notice that a, b and c have to follow the increasing order $a<b<c$. In our experiments, each image has 256 gray levels, i.e., the maximum value of c is 255. Therefore, the chromosome of the genetic algorithm in our experiment is encoded as three 8-bits strings which represent the value of point b and two multiplier factors, respectively. The factors promise that all chromosomes are legal, i.e., three parameters a, b and c generated by each chromosome have the order $0 \leqslant a < b < c \leqslant 255$.

Suppose that the three 8-bits string is 110110100100111100110110 whereas the relative position of a,b and c is as follows:

$$\underbrace{\underline{11011010}}_{a_m}\ \underbrace{\underline{01001111}}_{b}\ \underbrace{\underline{00110110}}_{c_m}$$

$a_m = 11011010$ (Binary) = 218 (Decimal),

$b = 01001111$ (Binary) = 79 (Decimal),

$c_m = 00110110$ (Binary) = 54 (Decimal).

Obviously, letting $(a,b,c) = (a_m,b,c_m)$ does not satisfy the criteria $0 \leqslant a < b < c \leqslant 255$. In such a case, we can assign zero to the value of object function for illegal chromosomes. It will not participate the reproduction of next generation. The drawback of this method is that there are too many useless chromosomes in the searching space. Here we use another method to make all chromosomes legal. That is, every chromosome will satisfy the criteria $0 \leqslant a < b < c \leqslant 255$. Notice that only 79 ($=b$) is the final value. The values 218 ($=a_m$) and 54 ($=c_m$) serve as multiplier factors, they need to be transferred to the range $[0,b-1]$ and $[b+1,255]$, respectively. Therefore,

$$a = (b-1)\times(a_m/255) = 78\times 218/255 \approx 67,$$

$$c = (b+1) + (254-b)\times(c_m/255)$$

$$= 80 + (254-79)\times 54/255 \approx 117,$$

i.e. $(a,b,c) = (67,79,117)$.

4.2. Object function and other GA parameters

The object function is the entropy function in Eq. (10). We have employed $N=3$ (*dark*, *middle* and *bright* three levels) and $N=5$ (*dark*, *middle-dark*, *middle*, *middle-bright* and *bright* five levels) for experiments. The partition method $\mathscr{M}$ is equal partition and this is only for the explanation convenience.

We have conducted experiments on many images. For all the tested images, the parameters of the genetic algorithm are set as follows:

MaxGen (Maximal generation number) = 100,

POPs (Population size) = 100,

Pc (Probability of crossover) = 0.5,

Pm (Probability of mutation) = 0.01.

After determining the coding method, object function and all GA parameters, GA can operate and find the solution (a,b,c) with the maximal entropy $H(A,a,b,c)$. Finally, the fuzzy region is determined by the interval $[a,c]$ from the final result, and the gray levels mapped to the division points of fuzzy domain by membership function can serve as the thresholding values.

5. Experimental results

In this section, we test the performance of the proposed approach among different images. The original images are shown in Figs. 1(a)–7(a). The corresponding histograms are shown in Figs. 1(b)–7(b). Each image has 256 gray levels from 0 (the darkest) to 255 (the brightest). The first subsection describes the experimental results of $N=3$ and $N=5$. The second subsection describes the effect on an image by different N values. In the third subsection, we use the obtained fuzzy regions to perform image enhancement and thresholding.

5.1. Three partitions versus five partitions

Here, we try to classify an image into three subimages (dark, middle and bright) and five subimages (dark, middle-dark, middle, middle-bright and bright, respectively), by dividing the fuzzy domain [0,1] into three and five equal parts. Due to the non-deterministic property of GA, the results will not be the same when we run the same GA program several times. Therefore, for each image we run the GA 20 times and choose the best one from these 20 results. The results based on three partitions are listed in Table 3. For five partitions, the results are listed in Table 4.

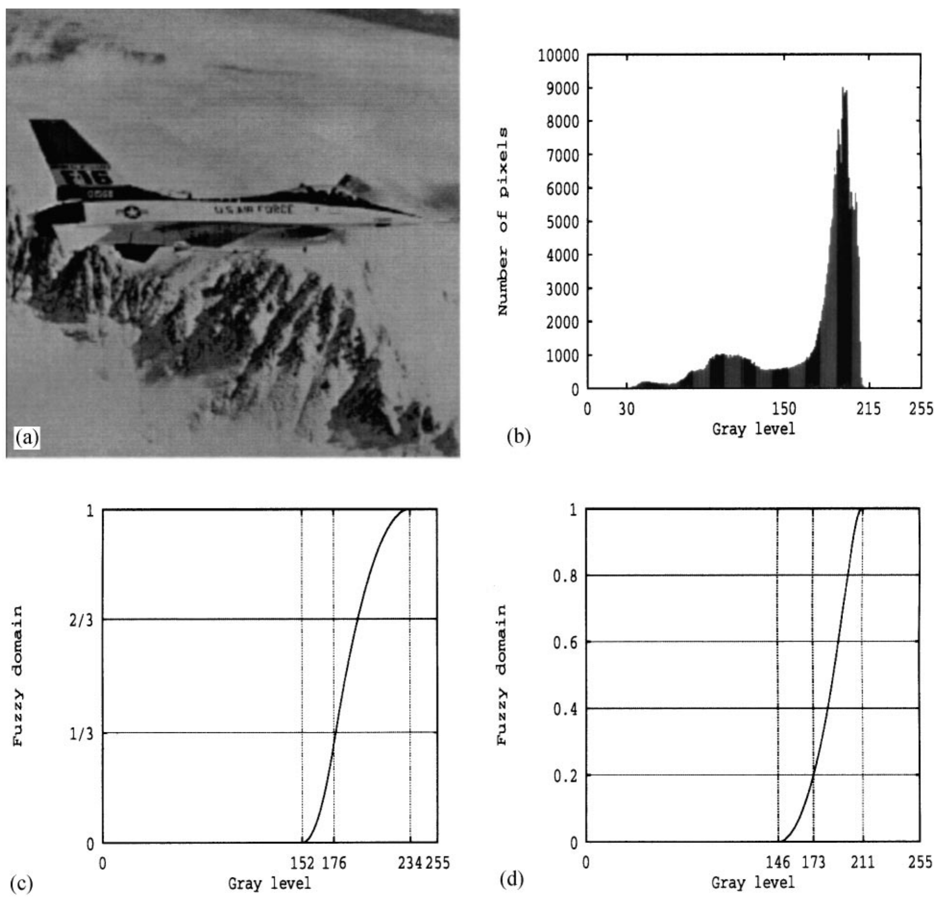

Fig. 1. “Airplane”: (a) original image, (b) histogram, (c) S-function (152, 176, 234) based on three partitions, (d) S-function (146, 202, 211) based on five partitions.

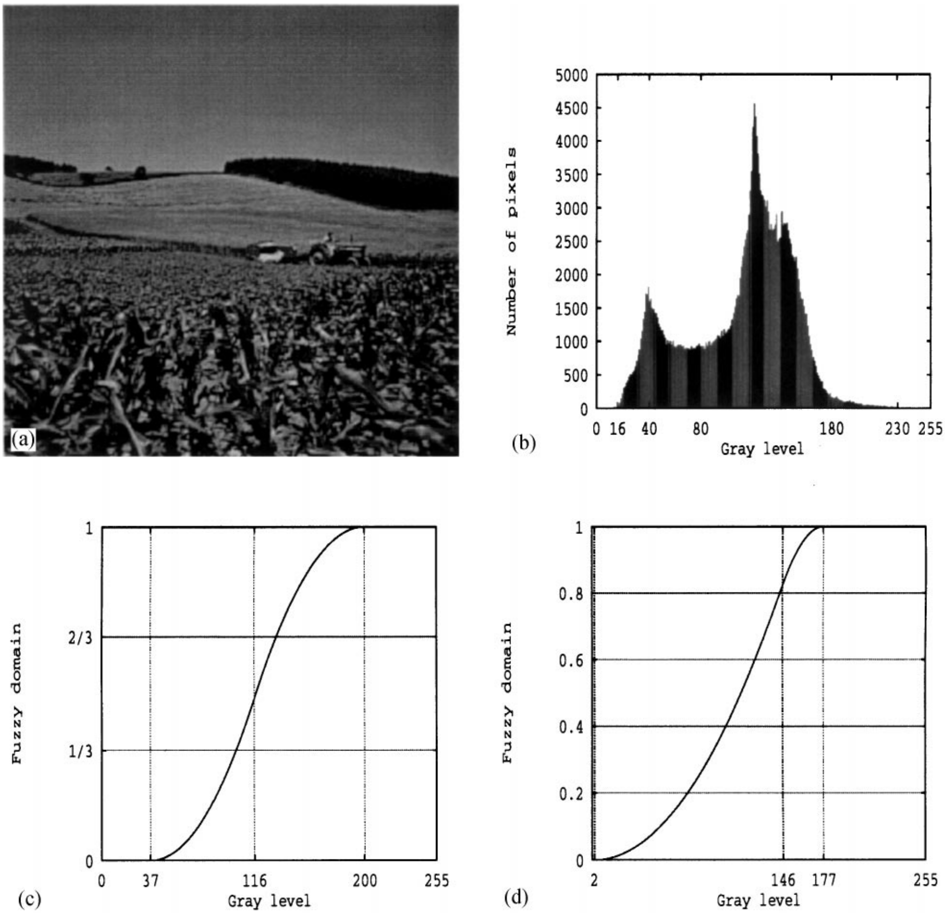

Fig. 2. "Cornfield": (a) original image, (b) histogram, (c) S-function (37, 116, 200) based on three partitions, (d) S-function (2, 146, 177) based on five partitions.

Their corresponding S-functions are shown in Figs. 1(c)–7(c) for three partitions and in Figs. 1(d)–7(d) for five partitions. From the results, almost all the S-functions are not symmetrical. For the bright image in Fig. 5(a), the fuzzy region [126,249] is close to the brightest (255). For the dark image in Fig. 6(a), the fuzzy region [12,122] is close to the darkest (0). It shows that the result from the proposed approach can reflect the accordance with the nature of input image. It is more flexible than the symmetrical methods due to the global maximum fuzzy entropy.

5.2. *Multiple partitions* ($N > 5$)

In this experiment, we take an image and try to classify its intensity into N parts where the values of N under consideration are 3, 5, 10, 20, 30, 50 and 100. As used in the previous experiment for convenience sake, the fuzzy domain is divided into N equal sub-intervals. The results are listed in Table 5.

In Fig. 4(b), the gray level range of "yacht" image is from 16 to 216. That is, the most dark pixel in this image has gray level 16 and the most bright pixel

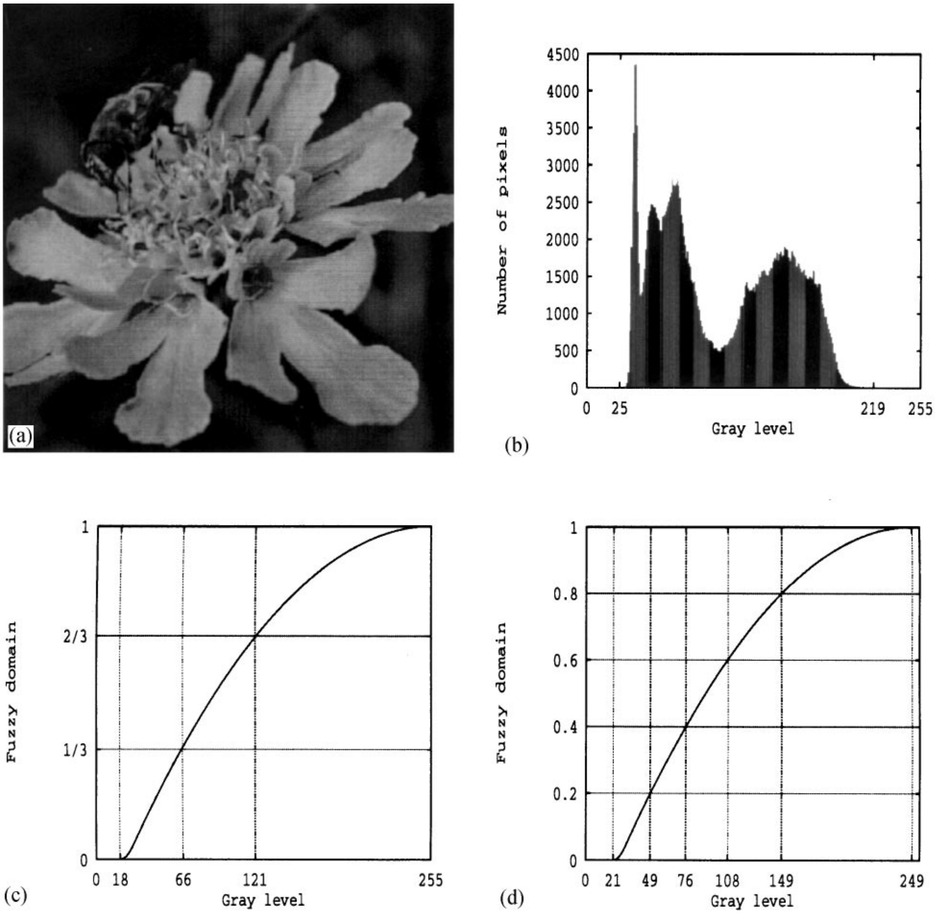

Fig. 3. “Flower”: (a) original image, (b) histogram, (c) S-function (18, 28, 255) based on three partitions, (d) S-function (21, 30, 249) based on five partitions.

Table 3
Fuzzy region (a,b,c), entropy and thresholds based on three partitions

Image	Gray level range	(a,b,c)	$H(A,a,b,c)$	Threshold
Airplane	[30,215]	(152,176,234)	0.999864	178,194
Cornfield	[16,230]	(37,116,200)	0.999935	103,132
Flower	[25,219]	(18,28,255)	0.985755	66,121
Yacht	[16,216]	(16,81,233)	0.999992	85,128
Lena	[24,245]	(1,125,252)	0.999973	103,148
Lena-bri	[139,249]	(126,191,249)	0.999890	178,200
Lena-drk	[12,122]	(12,50,128)	0.999890	50,73

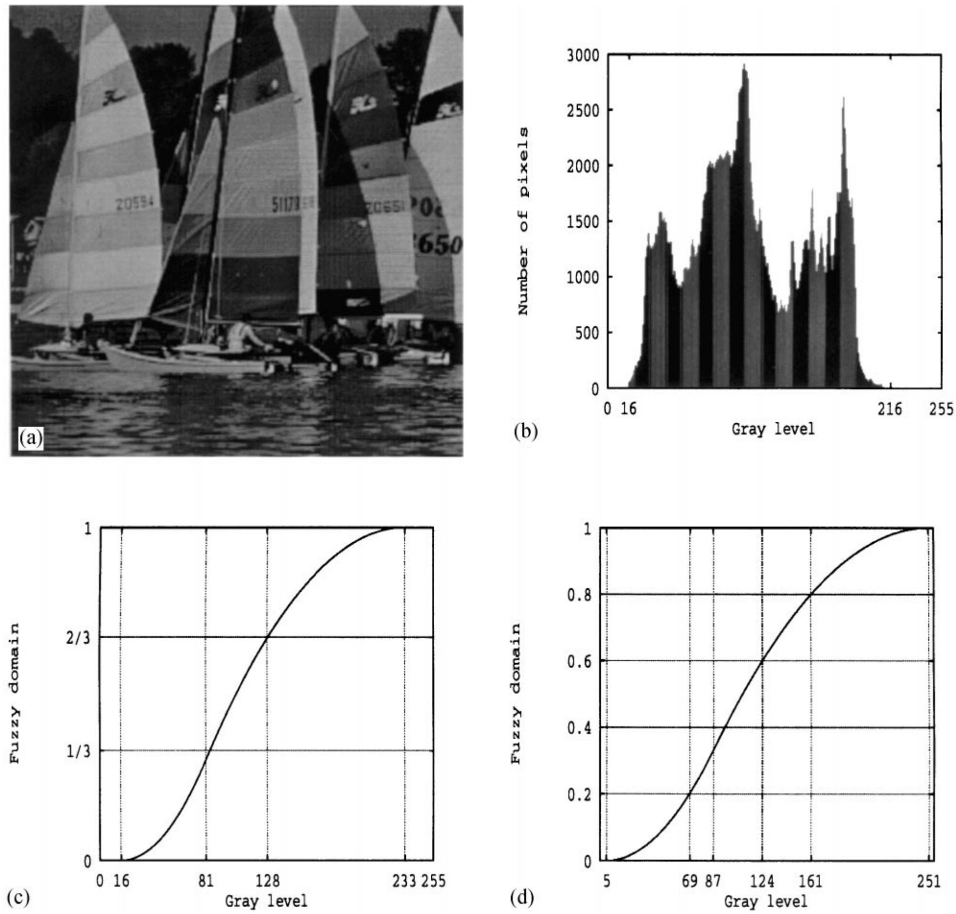

Fig. 4. "yacht": (a) original image, (b) histogram, (c) S-function (16, 81, 233) based on three partitions, (d) S-function (5, 87, 251) based on five partitions.

Table 4
Fuzzy region (*a*,*b*,*c*), entropy and thresholds based on five partitions

Image	Gray level range	(*a*,*b*,*c*)	$H(A,a,b,c)$	Threshold
Airplane	[30,215]	(146,202,211)	0.968965	173,184,193,200
Cornfield	[16,230]	(2,146,177)	0.986079	73,102,125,144
Flower	[25,219]	(21,30,249)	0.977177	49,76,108,149
Yacht	[16,216]	(5,87,251)	0.995919	69,95,124,161
Lena	[24,245]	(0,146,224)	0.998872	81,114,140,165
Lena-bri	[139,249]	(121,213,226)	0.999739	165,183,197,209
Lena-drk	[12,122]	(1,73,110)	0.998251	41,57,70,82

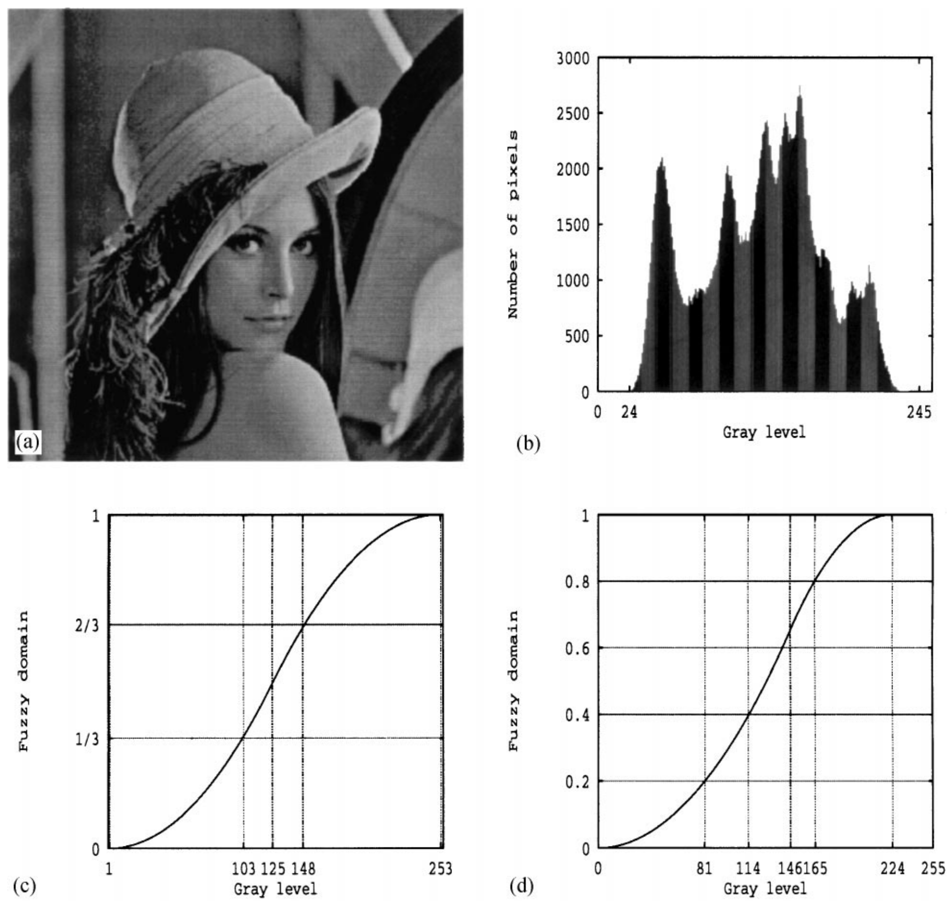

Fig. 5. "Lena": (a) original image, (b) histogram, (c) S-function (1, 125, 252) based on three partitions, (d) S-function (0, 146, 224) based on five partitions.

Table 5
Fuzzy region (*a*,*b*,*c*) based on different partition number *N*

N	Airplane [30, 215]	Cornfield [16, 230]	Yacht [16, 216]
3	(152,176,234)	(37,116,200)	(16,81,233)
5	(146,202,211)	(2,146,177)	(5,87,251)
10	(147,201,212)	(1,137,183)	(0,81,238)
20	(61,207,210)	(4,150,176)	(6,98,216)
30	(57,206,213)	(11,139,179)	(7,99,214)
50	(52,207,210)	(3,142,184)	(13,96,215)
100	(26,205,211)	(6,143,187)	(12,86,215)

has intensity 216. As you can see from the results of image "yacht" in Table 5, when *N* value changes from 20 to 100, the fuzzy region [*a*,*c*] changes from [6,216] through [7,214], [13,215] to [12,215]. These fuzzy regions almost match the gray level range [16,216].

In the case of image "cornfield", its gray level range is [16,230]. From the results, when *N* is between 5 and 100, the fuzzy region [*a*,*c*] of "cornfield" is [5,180]. It seems not matching the gray level range [16,230]. This can be explained by looking at the histogram of "cornfield". From Fig. 2(b), the histogram is relatively small between gray level 180 and 230 compared with the entire histogram. Therefore, this relatively small part at the end of histogram will be ignored. Reversely, the gray level near 16 (especially between 20 and 80) has relatively large portion of histogram (there is a peak at 40), this will cause *a* going down to 0 in order to reduce the probabilities of fuzzy event A_1. That is why *a*

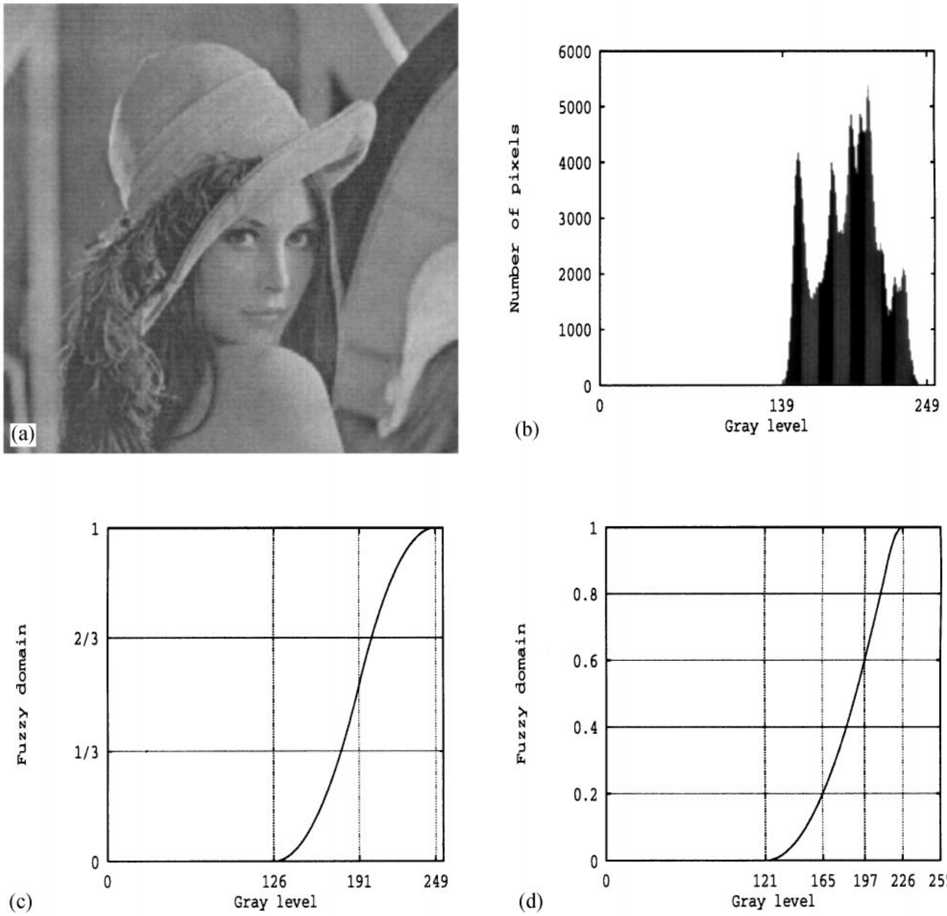

Fig. 6. "Lena-bri": (a) original image, (b) histogram, (c) S-function (126, 191, 249) based on three partitions, (d) S-function (121, 213, 226) based on five partitions.

is stable around 5, not the left end of histogram range 16.

A more explicit example is the image "airplane" whose histogram has gray level range between 30 and 215. From Fig. 1(b), it has a relatively low histogram between gray level 30 and 150. Therefore, when N is between 3 and 50, most of them are ignored and a is found larger than 30. When N becomes larger, a becomes smaller.

These examples show that fuzzy region found by the proposed method does not simply exactly match the gray level range of the histogram. Actually, it is dependent on the distribution of histogram, the partitions method $\mathscr{M}$, the number of partitions N, and the maximum entropy principle. Therefore, it is more useful and rational.

5.3. Image enhancement and thresholding

Now, we apply the results from the proposed approach to image enhancement and image thresholding. Image enhancement can be achieved by applying the *contrast intensification* (INT) operation on the image in fuzzy domain [12]. Assume that A is an image (a fuzzy set) in fuzzy domain with membership function $\mu_A(x)$. The contrast

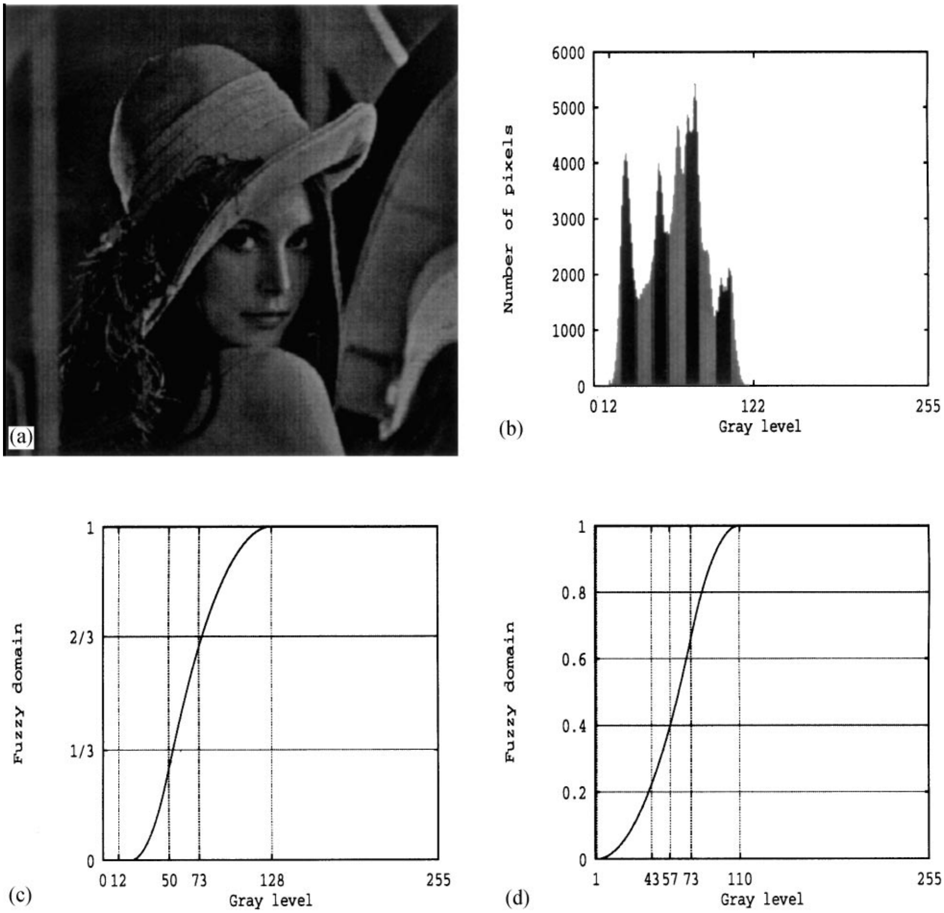

Fig. 7. "Lena-drk": (a) original image, (b) histogram, (c) S-function (12, 50, 128) based on three partitions, (d) S-function (1, 73, 110) based on five partitions.

intensification of A denoted by INT(A) is defined as

$$\mu_{\mathrm{INT}(A)}(x) = \begin{cases} 2(\mu_A(x))^2, & 0 \leqslant \mu_A(x) \leqslant 0.5, \\ 1 - 2(1 - \mu_A(x))^2, & 0.5 \leqslant \mu_A(x) \leqslant 1. \end{cases}$$

INT is one of the basic operations on fuzzy sets and has the effect of altering the fuzziness of the fuzzy set. It reduces the fuzziness of A by increasing those of $\mu_A(x)$ which are above 0.5, and decreasing those which are below 0.5. After mapping the images into fuzzy domain, we perform INT operation to get new membership function, then use inverse membership function μ_A^{-1} to map the new membership back to intensity domain, and an enhanced image is generated. The image enhancement operation can be expressed by

$$x' = \mu_A^{-1}[\mu^{+}_{\mathrm{INT}(A)}(x)],$$

where x' is the enhanced intensity and the plus sign means that INT operation should be performed at least one time. Notice that when the number of INT operations increases to infinity, the effect on resultant image will be binary thresholding like.

The images under experiment are "flower" in Fig. 3(a), "yacht" in Fig. 4(a), and "lena-drk" in Fig. 7(a). We select (a,b,c) from Table 3 as the parameters of S-function to map images into fuzzy domain and perform INT operation three times for each image.

The comparison of original and enhanced images are shown in Fig. 8. The enhanced images have high-contrast between bright and dark.

Image thresholding is an important approach to image segmentation. It can be accomplished in our approach by using the inverse S-function to map the partition point of fuzzy domain back to gray level range. In the case of three equal partitions, we map $\frac{1}{3}$ and $\frac{2}{3}$ from fuzzy domain back to intensity domain by the inverse S-function which is dependent on the parameters (a,b,c) found from our approach. The results are shown in the last column of Tables 3 and 4.

Several researchers had used the concept of entropy to perform image thresholding [4,13], we compared the threshold values from Kapur's method [13], Pun's method [4] and our approach for binary thresholding. Some of them are listed in

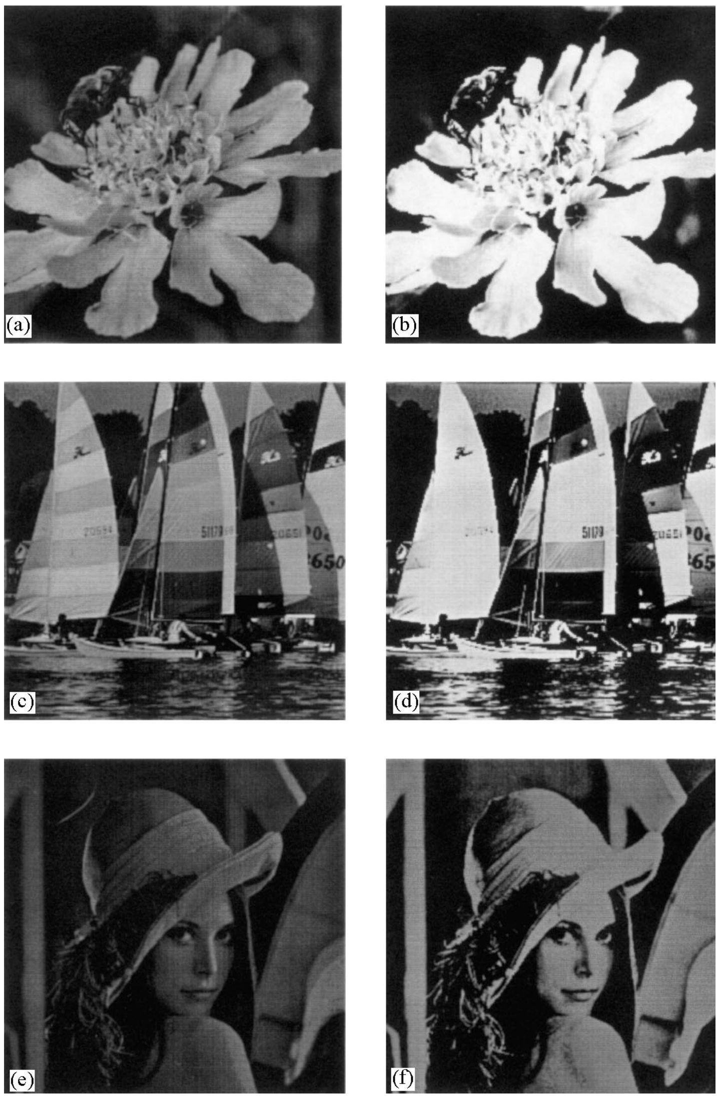

Fig. 8. Enhanced images: (a) original "flower", (b) enhanced "flower", (c) original "yacht", (d) enhanced "yacht", (e) original "lena-drk", (f) enhanced "lena-drk".

Table 6 and the binary images are shown in Figs. 9–12. In these four figures, (a) represents the original image, (c) is the histogram, (b), (d) and (e) are the binary images for Kapur's algorithm, Pun's algorithm and the proposed approach, respectively. In Fig. 9, the girl's hair, eyes, and the outline of the face are best represented by the proposed approach. In Fig. 10, we can observe that the proposed approach best describes the shape of the face and the eyes of the girl. From the binary images in Figs. 11(b), 11(d) and 11(e), we can observe much more details of the life jackets and the stones on the bank of the revier by using the proposed approach. Fig. 12 also shows that the background, the face, the eyes, etc. are better described using the proposed approach. Three-level and five-level thresholdings are shown in Figs. 13 and 14, respectively. In general, higher-level thresholding can produce more

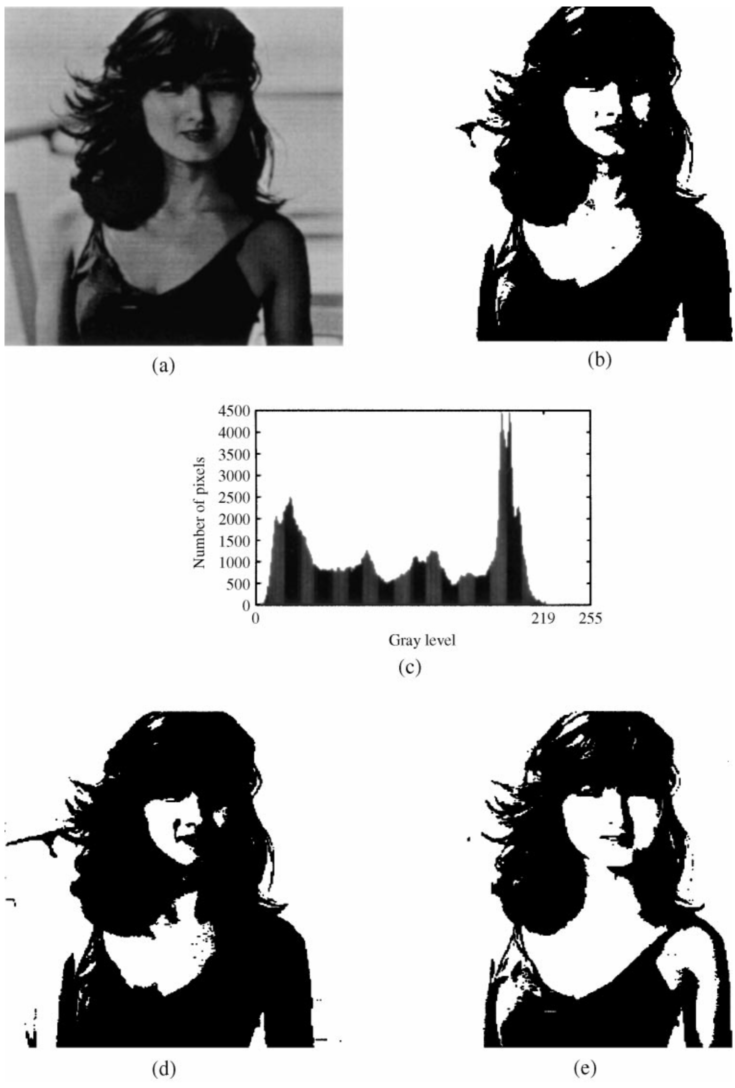

Fig. 9. Two-level thresholding: (a) original image "bluegirl", (b) binary "bluegirl" with $T = 107$ by Kapur's algorithm, (c) histogram of "bluegirl", (d) binary "bluegirl" with $T = 116$ by Pun's algorithm, (e) binary "bluegirl" with $T = 77$ by proposed approach.

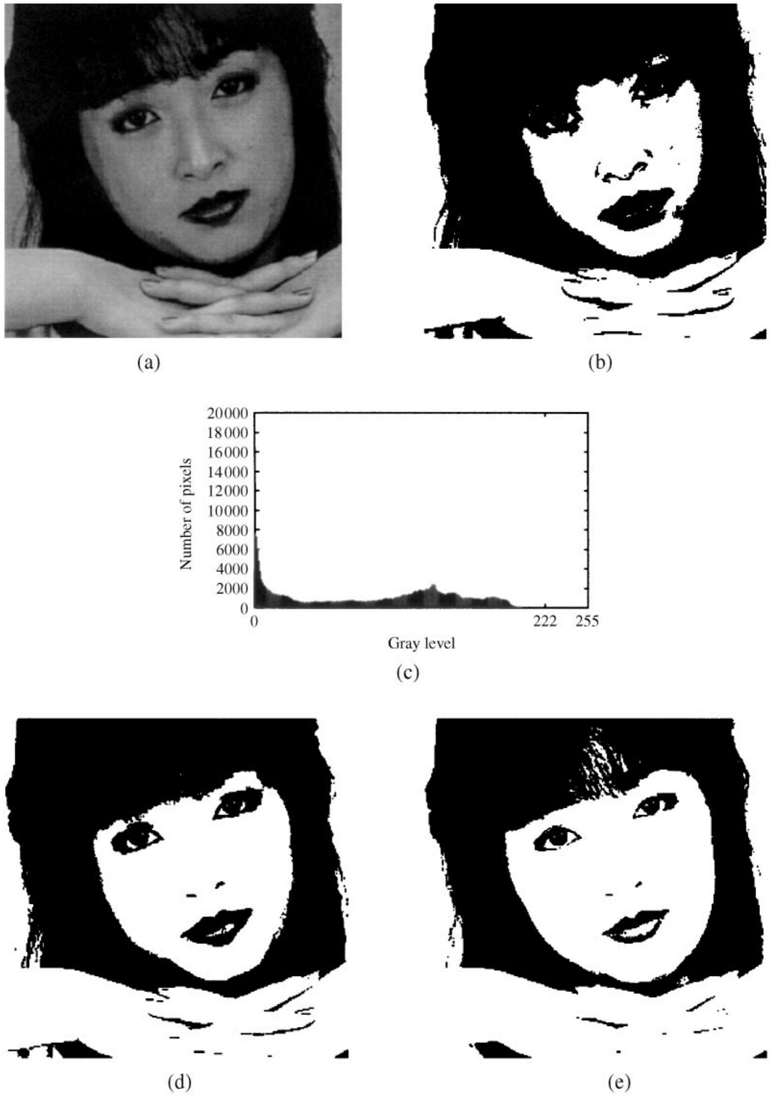

Fig. 10. Two-level thresholding: (a) original image “masuda”, (b) binary “masuda” with $T = 115$ by Kapur’s algorithm, (c) histogram of “masuda”, (d) binary “masuda” with $T = 97$ by Pun’s algorithm, (e) binary “masuda” with $T = 75$ by proposed approach.

Table 6
Binary threshold values

	Sport	Girl	Masuda	Bluegirl
Kapur’s algorithm	116	122	115	107
Pun’s algorithm	96	52	97	116
Proposed approach	86	75	75	77

uniform results close to the original ones, but, it requires more computational resources. Certainly, how many levels should be employed that depends on the applications and could be user-defined. Figs. 15–17 demonstrate the results using non-equal partitions. We run a series of experiments on non-equal partitions with $N = 3$ and $N = 5$. $N = 3$ refers to partition the pixels into dark, middle and

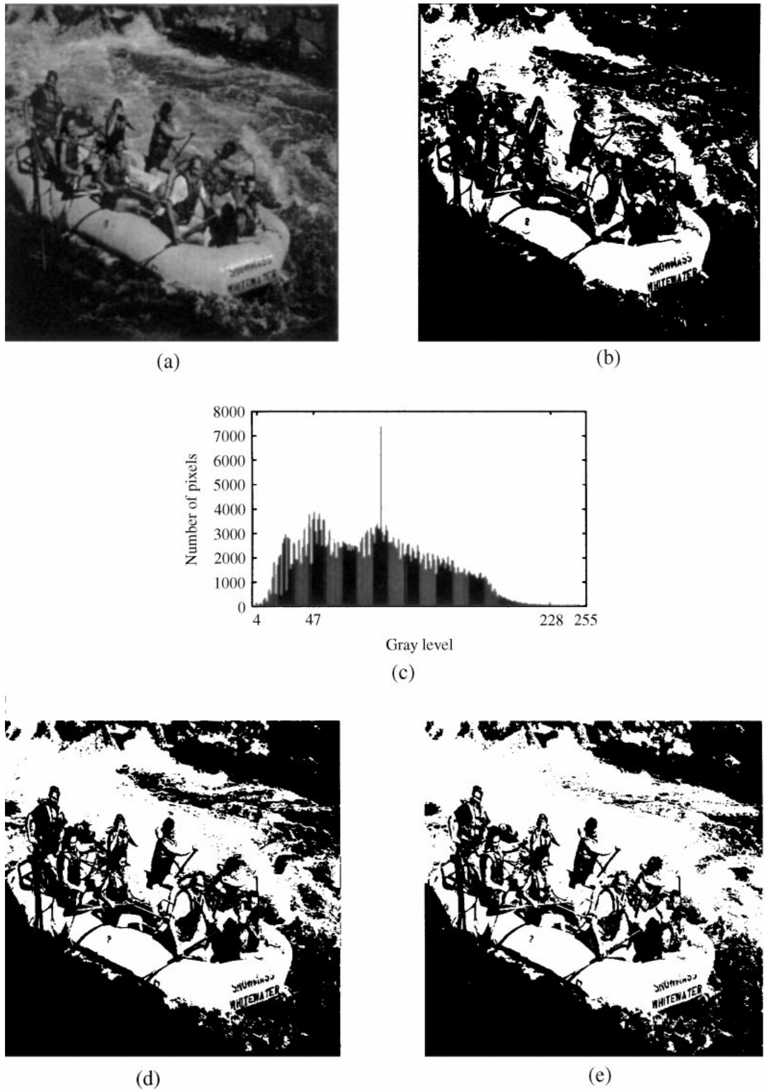

Fig. 11. Two-level thresholding: (a) original image "sport", (b) binary "sport" with $T = 116$ by Kapur's algorithm, (c) histogram of "sport", (d) binary "sport" with $T = 96$ by Pun's algorithm, (e) binary "sport" with $T = 86$ by proposed approach.

bright three categories: A_1, A_2 and A_3. $N = 5$ refers to dark, middle-dark, middle, middle-bright and bright five categories: A_1, A_2, A_3, A_4 and A_5. For instance, as shown in Fig. 17, we apply parameters (0.5,0.4,0.1) and $N = 3$ on the image "flower", which means that the sub-intervals [0,0.5], [0.5,0.9] and [0.9,1] represent A_1, A_2 and A_3, respectively. Whether equal partition or non-equal partition should be employed that is image-dependent/task-dependent or user-defined.

6. Discussions

There are many formulas for computing the entropy of a fuzzy set as described in Section 2.2 such

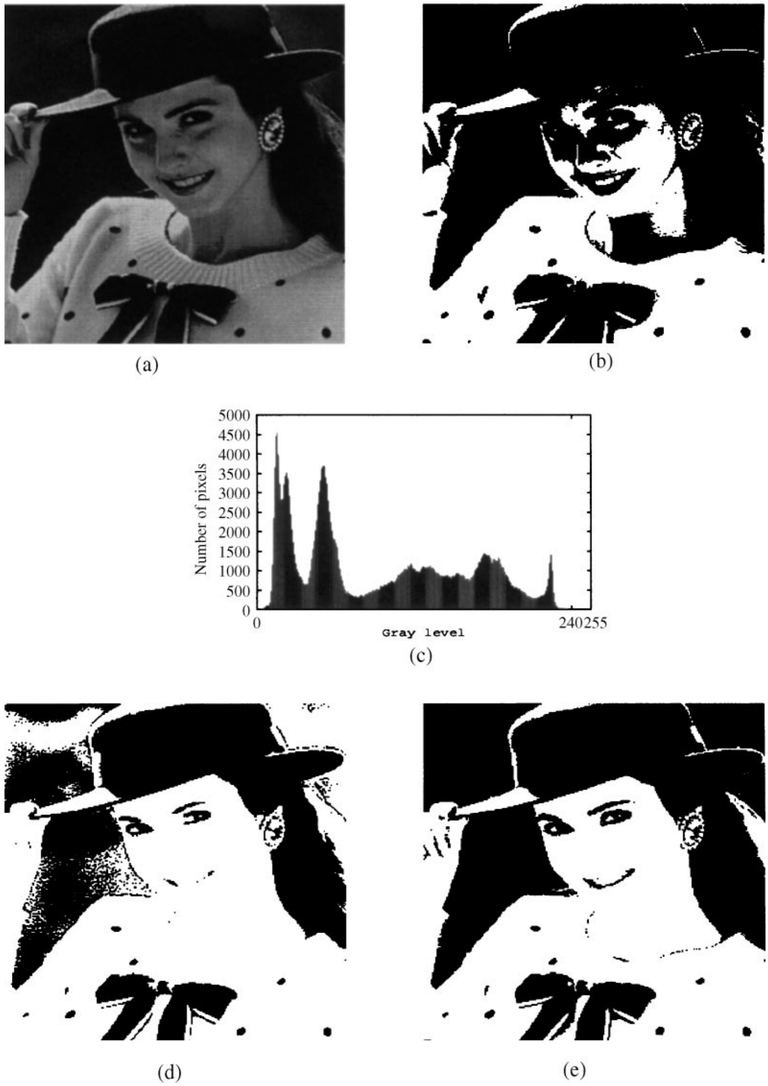

Fig. 12. Two-level thresholding: (a) original image "girl", (b) binary "girl" with $T = 122$ by Kapur's algorithm, (c) histogram of "girl", (d) binary "girl" with $T = 52$ by Pun's algorithm, (e) binary "girl" with $T = 75$ by proposed approach.

as Eqs. (5)–(7). Here we want to explain why we need to propose a new method. Notice that the membership function used in our experiments is an asymmetric S-function with three parameters (a,b,c) and our goal is to keep the entropy as large as possible. Besides, the solution of (a,b,c) should be related to the histogram distribution.

In Zadeh's formula Eq. (5), $P(x_i)$ is from the intensity distribution of an image. When GA generates a new combination of (a,b,c), $\mu_A(x_i)$ will change but the production $P(x_i)\log P(x_i)$ will keep unchanged. In order to make $H(A)$ as large as possible, $\mu_A(x_i)$ should be as large as possible. Because the maximal value of $\mu_A()$ is 1, then c will go below the minimal gray level to make $\mu_A(x_i) = 1$ for

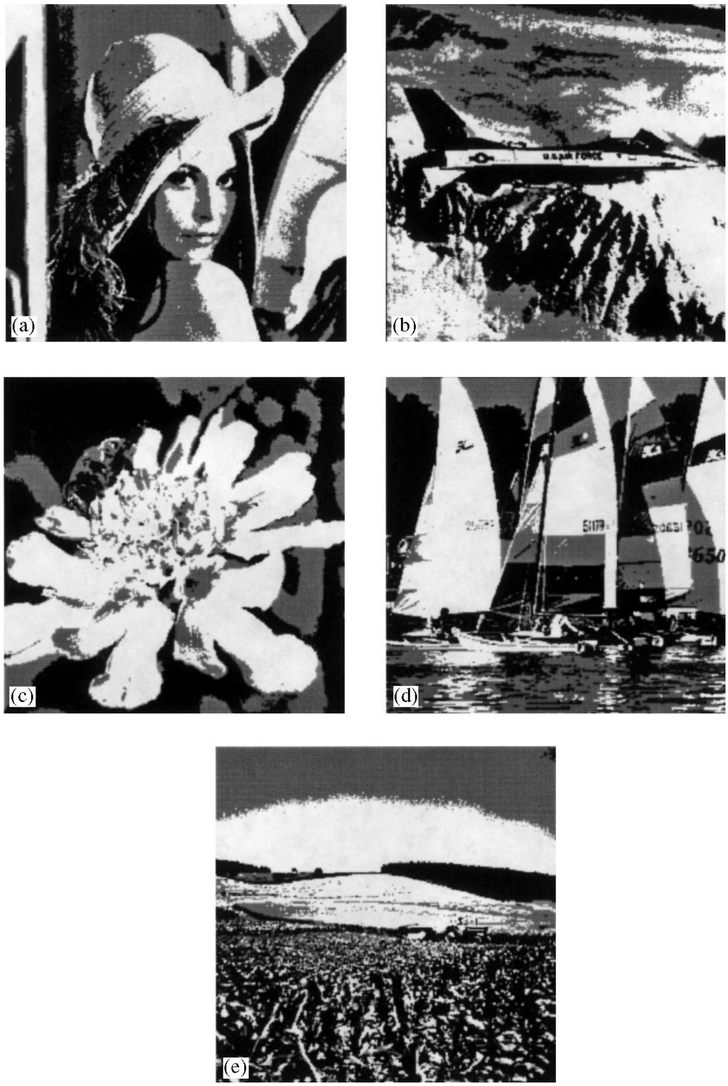

Fig. 13. Three-level thresholding: (a) "lena" with $T = 103$ and 148, (b) "airplane" with $T = 178$ and 194, (c) "flower" with $T = 66$ and 121, (d) "yacht" with $T = 85$ and 128, (e) "cornfield" with $T = 103$ and 132.

$i = 1,2,\ldots,N$. Therefore, any combination of (a,b,c) with the restriction of c below the minimal gray level of image will make all the memberships be 1 in order to get the maximal entropy value. However, this kind of (a,b,c) is not what we desired.

In Kaufmann's formula Eq. (6), the entropy will reach the maximum when $\varphi_A(x_i) = 1/N$ for $i = 1,2,\ldots,N$. This is the same concept as described in Eq. (2) from information theory. Since $\varphi_A(x_i)$ is defined as

$$\varphi_A(x_i) = \frac{\mu_A(x_i)}{\sum_{i=1}^{N} \mu_A(x_i)},$$

its value cannot be $1/N$ unless $\mu_A(x_i) = 1$ for $i = 1,2,\ldots,N$. This will get the same result as

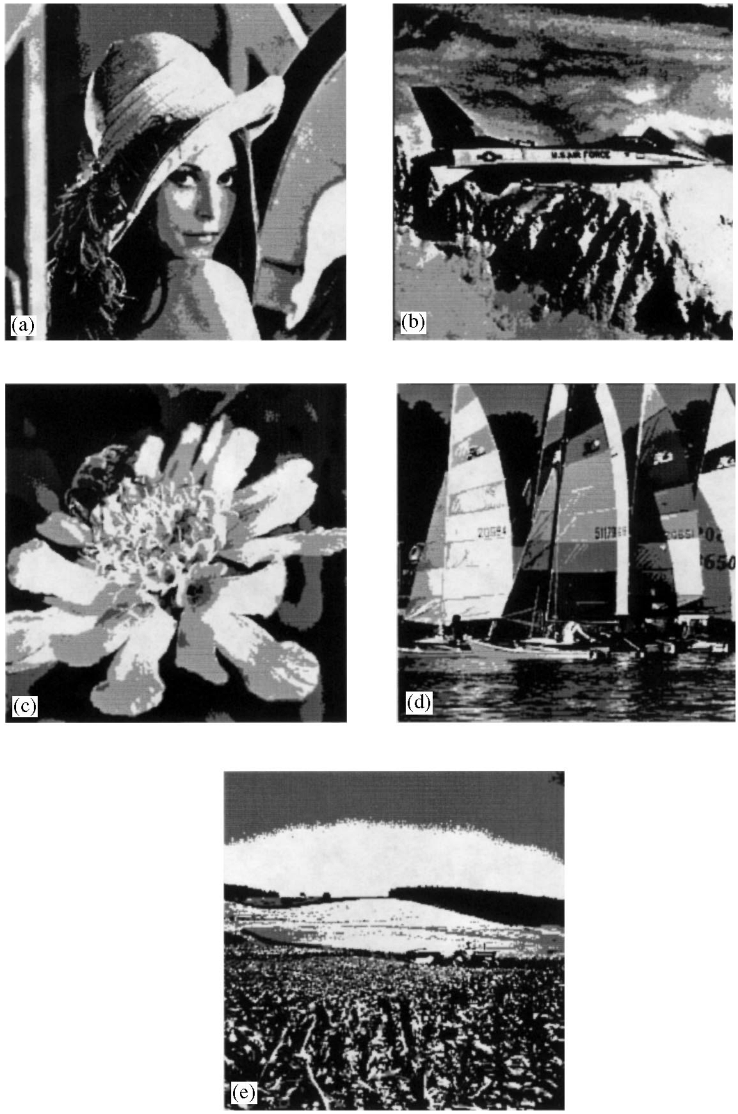

Fig. 14. Five-level thresholding: (a) "lena" with $T = 81, 114, 140$ and 165, (b) "airplane" with $T = 173, 184, 193$ and 200, (c) "flower" with $T = 49, 76, 108$ and 149, (d) "yacht" with $T = 69, 95, 124$ and 161, (e) "cornfield" with $T = 73, 102, 125$ and 144.

Zadeh's entropy. Consequently, GA will try to make all the memberships equal to 1 as the solution.

In De Luca and Termini's entropy function Eq. (7), $H(A)$ is a summation on N Shannon's functions. According to Eq. (8), $S_n(\mu_A(x_i))$ will reach the maximum 1 when $\mu_A(x_i) = 0.5$. Therefore, making most of the gray levels to have membership close to 0.5 will get a larger entropy. In order to achieve this goal, a and c will move to the two ends of gray levels, i.e., a is close to 0 while c is close to 255, and b will go close to either a or c. This makes all the fuzzy regions almost the same to be [0,255]. This result does not show the capability of reflecting the natures of different input images.

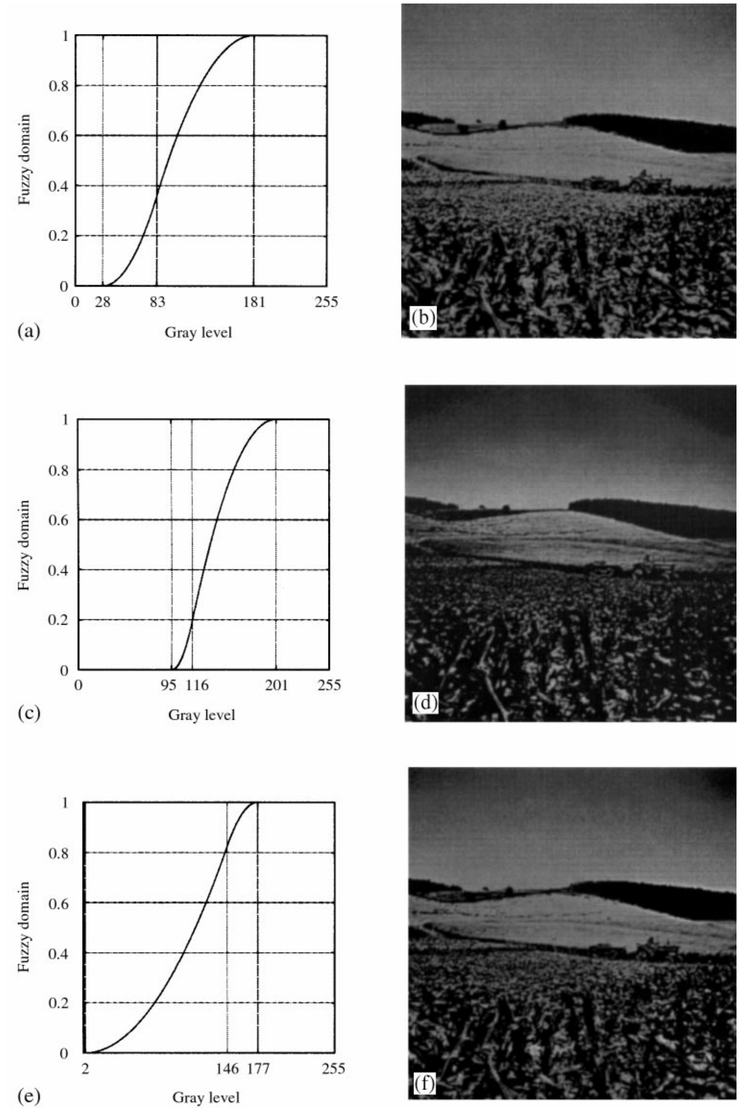

Fig. 15. S-functions and fuzzy domain images based on non-equal partition and equal partition. (a) S-function (28, 83, 181), (b) "cornfield" in fuzzy domain using non-equal partition $N = 5$ (0.1, 0.6, 0.1, 0.1, 0.1), dark portion enhanced, (c) S-function (95, 116, 201), (d) "cornfield" in fuzzy domain using non-equal partition $N = 5$ (0.2, 0.2, 0.2, 0.1, 0.3), bright portion enhanced, (e) S-function (2, 146, 177), (f) "cornfield" using equal partition $N = 5$ (0.2, 0.2, 0.2, 0.2, 0.2).

In order to overcome the weakness of the above entropy formulas, we propose a new definition of fuzzy entropy which combines the concepts of probability and fuzzy membership. By employing the proposed approach, an image can be transformed into fuzzy domain with maximum fuzzy entropy reflecting the nature of the image. The selected fuzzy region and membership function can be used in thresholding and segmentation.

Notice that the number of categories N and the partition method $\mathscr{M}$ in our approach are two task-dependent settings. Actually, it can be used in many

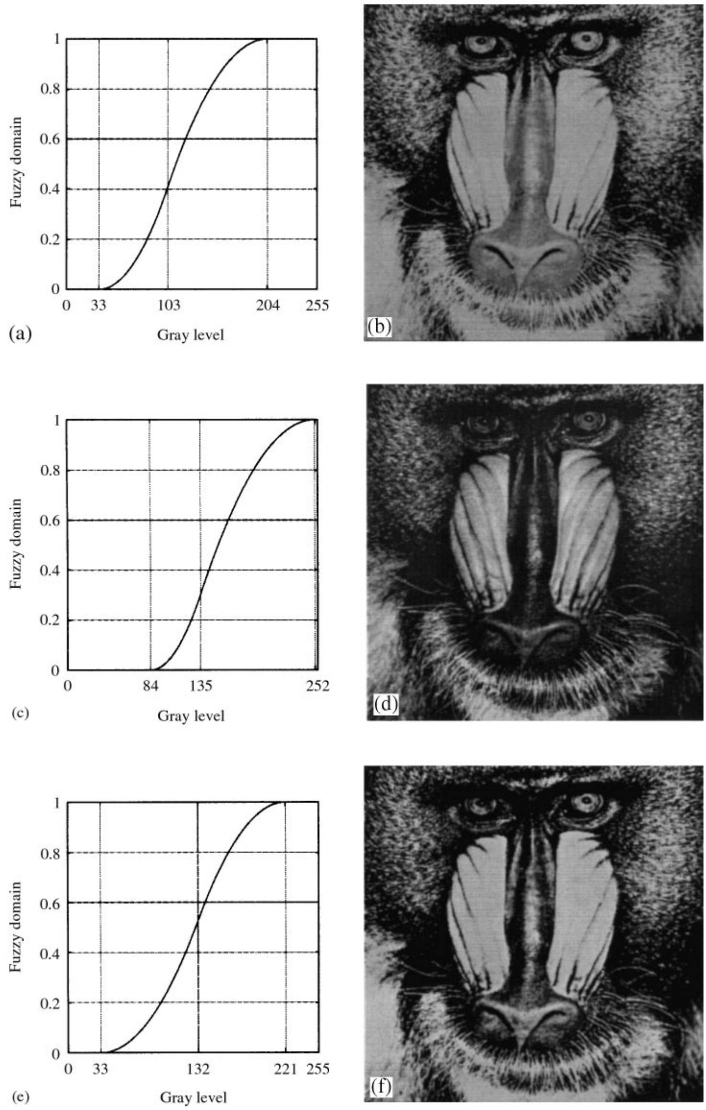

Fig. 16. S-functions and fuzzy domain images based on non-equal partition and equal partition. (a) S-function (33, 103, 204) (b) "baboon" in fuzzy domain using non-equal partition $N = 5$ (0.3, 0.3, 0.1, 0.2, 0.1), dark portion enhanced, (c) S-function (84, 135, 252), (d) "baboon" in fuzzy domain using non-equal partition $N = 5$ (0.1, 0.1, 0.3, 0.1, 0.4), bright portion enhanced, (e) S-function (2, 146, 177), (f) "baboon" in fuzzy domain using equal partition $N = 5$ (0.2, 0.2, 0.2, 0.2, 0.2).

applications to automatically select the fuzzy region when the data distribution is given and the described object can be viewed in the fuzzy domain like "fast", "rich", "old", "high", "cold", etc. Take the speed control of cars as an example, if the speed needs to be classified into five categories (very fast, fast, middle, slow, very slow), we just reasonably divide the fuzzy domain [0,1] into five parts (equally or nonequally, it depends on the tasks).

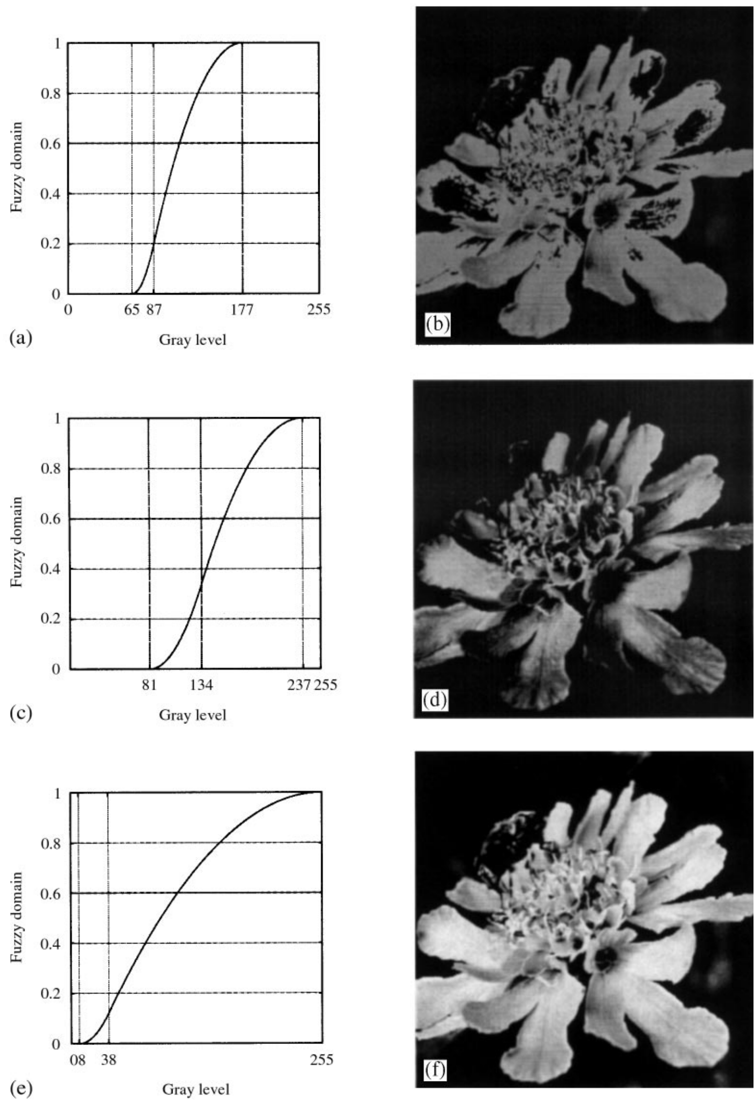

Fig. 17. S-functions and fuzzy domain images based on non-equal partition and equal partition. (a) S-function (65, 87, 177), (b) "flower" in fuzzy domain using non-equal partition $N = 3$ (0.5, 0.4, 0.1), dark portion enhanced, (c) S-function (81, 134, 237), (d) "flower" in fuzzy domain using non-equal partition $N = 3$ (0.3, 0.3, 0.4), bright portion enhanced, (e) S-function (8, 38, 255), (f) "flower" in fuzzy domain using equal partition $N = 3$ (1/3, 1/3, 1/3).

7. Conclusions

An automatically selecting fuzzy region approach with a new definition of fuzzy entropy is developed. The new entropy definition overcomes the drawbacks of the existing entropy definitions. The fuzzy region is found by a genetic algorithm based on the maximum fuzzy entropy principle. The image can keep as much information as possible when the image is transformed from the intensity domain to the fuzzy domain. In most of the existing methods, the cross-over point of S-function is set as the mid-point of the fuzzy region $[a,c]$. That is, $\Delta b = b - a = c - b$. However, the mid-point strategy does not promise that the fuzzy entropy is the global maximum. In our experiments, when searching for the fuzzy region $[a,c]$, parameter b of the asymmetrical S-function is determined at the same time to make the fuzzy entropy higher than that using the symmetrical S-function. We have applied the proposed approach to perform image enhancement and thresholding. The results have shown the satisfactory. The proposed approach will have wide applications in the areas employing fuzzy logic.

References

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

[2] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, 1989.

[3] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addison-Wesley, 1993.

[4] J.N. Kapur, P.K. Sahoo, A.K.C. Wong, A new method for gray-level picture thresholding using the entropy of the histogram, Comp. Graphics, Vision and Image Proc. 29 (1985) 273–285.

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

[6] S. Kirkpatrick, C.D. Gelatt, Jr., M.P. Vecchi, Optimization by simulated annealing, Science 220 (May 1983) 671–680.

[7] D.E. Knuth, The Art of Computer Programming, Vol. 3, Addison-Wesley, 1973.

[8] X. Li, Z. Zhao, H.D. Cheng, Fuzzy entropy threshold approach to breast cancer detection, Information Sciences, Applications, An International Journal 4 (1) (1995) 49–56.

[9] X. Li, Z. Zhao, H.D. Cheng, C. Huang, R.W. Harris, Fuzzy logic approach to image segmentation, in: The 12th International Conference on Pattern Recognition, Jerusalem, Israel, 9–13 October 1994.

[10] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, 3rd edition, Springer, Berlin, 1996.

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

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

[13] T. Pun, A new method for grey-level picture thresholding using the entropy of the histogram, Signal Processing 2 (1980) 223–237.

[14] C.E. Shannon, A mathematical theory of communication, Bell Syst. Tech. J. 27 (July 1949) 379–423.

[15] L.A. Zadeh, Probability measures of fuzzy events, Journal of Mathematical Analysis and Applications 23 (1968) 421–427.

[16] J.M. Zurada, Introduction to Artificial Neural Systems, West Publishing, MN, 1992.