Parallel entropic auto-thresholding

M. Fleury*, L. Hayat*, A.F. Clark

Department of E.E., Cass University, Wexham Park, Berkshire SL4 4SF, U.K.

Received 5 April 1995; revised 20 June 1995

Abstract

In this paper, we examine a multi-level thresholding algorithm based on a number of phases including peak-search, fuzzy logic and entropy of the fuzzy membership function. Analysis of the algorithm is presented to show its properties and behavior at the various cascaded stages. The fuzzy entropy function of the image histogram is computed using $S$-function membership and Shannon’s entropy function. To establish a suitable fuzzy region bandwidth, the entropy of the image histogram is used to find a suitable threshold value. Location of the valleys in the entropy function correspond to the certainty within the fuzzy region of the image. Differences are used to indicate an optimal segmentation pattern for multi-level image thresholding. We compare and contrast this method of thresholding with a maximum entropy method. We have implemented the technique in parallel on a transputer-based machine as well as on a cluster of SUN4 workstations, involving ourselves in the PVM communication kernel. A parallel algorithm for the maximum entropy method is given, which significantly reduces computation times. An objective method is used to evaluate the resulting images.

Keywords: Fuzzy-entropy; Thresholding; Parallelism

1. Introduction

Image thresholding is a fundamental process in applications based on image processing and computer vision such as robotic vision, medical diagnosis, and even postal code recognition. Furthermore, a multi-level thresholding technique plays an important role, especially in those imaging problems where the prime objectives are to establish crisp boundaries in order to partition the image into its meaningful regions and extract its features. The ideal situation would be when peaks and valleys in the histogram of the image data are clear, symmetric and balanced. The bottom of each valley would then be the natural choice for segmentation into meaningful homogeneous regions. For example, if there are $k$ clear valleys then the image can be divided into $k+1$ homogeneous segments using the following equation:

$$
g(k; x, y) = \begin{cases} 
1, & f(x, y) < T(k) \\
k - 1, & T(k - 1) < f(x, y) \leq T(k) \\
\cdots & \\
0, & \text{otherwise}
\end{cases}
$$

Here $T(k)$ is the grey-tone threshold, $f$ return the input image grey-tone intensity and $g$ is the output image, with $0 \leq k \leq N, 0 \leq y \leq M$ for image size $N \times M$. Unfortunately, in practical terms, such situations are rare. To overcome this situation, during the past few decades a wide range of thresholding techniques such as histogram modification [1], valley seeking [2], maximum-entropy thresholding [3,4] and fuzzy measures [5-7] have been reported. A comprehensive survey of such techniques can be found in [8-11].

In this paper, we investigate a multi-level thresholding algorithm which utilizes the fuzzy entropy function of the image histogram combined with other techniques to arrive at a threshold pattern. This entropy method is computationally faster, which is important when processing batches of images such as those required for industrial inspection. The algorithm is broadly divided into three different stages: peak-search, fuzzy logic and minimum fuzzy-entropy search. For a given fuzzy bandwidth, image grey levels are selected into fuzzy outputs after applying a non-linear transformation. The entropy of the fuzzy outputs is computed through Shannon’s function [12]. Instead of relying on one measure of entropy, we have constructed the entropies of the image histogram for each given fuzzy bandwidth selected by a
peak detector. The location of valleys in the fuzzy entropy function are taken as the threshold points.

We have also examined a maximum entropy thresholding method as a point of comparison. This method works by finding the point where the maximum information is preserved for a given segmentation. This will also be the point of greatest uncertainty. Two objective measures, uniformity and shape, were used to compare the results for bi-modal and multi-modal images, with a simple averaging method used as a reference point. The trade-off with computational times when the algorithms are parallelized is taken into account.

The rest of the paper is organized as follows. Section 2 reviews the concepts of entropy and fuzzy entropy. Section 3 gives details of the cascaded stages of the algorithm. The following section describes the parallel implementation. An evaluation of the results is given in Section 5. Section 6 looks at an alternative possibility. Finally, Section 7 draws conclusions.

2. Entropy thresholding methods

The information theoretic entropy, which measures the mean value of the uncertainty, is defined as

$$H = - \sum_{i=1}^{n} p_i \log p_i \quad H \in [0,1]$$

where the \( p_i \) form a probability distribution. This has the principal properties that, with uncertainty defined as \(-\log p_i\), events of lower probability are more uncertain, and the uncertainty of two simultaneous events is the sum of their individual uncertainties. In the paper by Kapur et al. [4], an \textit{a posteriori} entropy measure to be used for image segmentation is defined as

$$Enp(s) = -(E_n + E_e)$$

where

$$E_n = \sum_{i=1}^{n} \left( \frac{p_i}{p_e} \right) \log \left( \frac{p_i}{p_e} \right)$$

$$E_e = \sum_{i=1}^{n} \left( \frac{p_e}{p_i} \right) \log \left( \frac{p_e}{p_i} \right)$$

for all \( s \in (0,n-2) \)

\( P_i \) are the grey-level probabilities, with \( a \) being the number of grey levels and \( s \) representing potential threshold points. The aim is to maximize the information content by maximizing the entropy measure, which is brought about by varying \( s \). Now, as \( s \) moves away from its leftmost position (i.e. 0) towards the rightmost position (i.e. \( n \)) the value of \( E_n \) increases while the value of \( E_e \) decreases (Fig. 1). The effect of Eqs. (3–5) is depicted in Fig. 2. A feature discernible in the figure is that, because \( Enp(s) \) is a relative measure, this method responds in the first instance to distributions and not to amplitudes.

The same method can be extended to multi-modal images\(^{2}\) by defining \( Enp(s) = -\sum_{i=1}^{n} E_i \) in the natural way. To find the optimum partition, we have adopted a scheme shown in Fig. 3 for a three-segment partition. According to the figure, \( A \) and \( B \) are fixed at positions near to the leftmost available grey level, \( R_l \). Then \( B \) is slid sequentially and successively towards the rightmost possible position, \( R_1 \) while keeping \( A \) fixed at its previously-defined position. When \( B \) reaches \( R_1 \), then 1 is slid by one position towards the right and \( B \) is reset to the position \( A+1 \). This process is repeated until \( A = R_n \) and \( B = R_1 \). All values of the entropies are recorded. The partition (i.e. the positions of \( A \) and \( B \)) generating maximum value of \( Enp(s) \) is used to set the threshold points for image segmentation. The same procedure could be extended for the \( k \)th partition of the image into homogeneous regions. However, as the number of partitions increases, the time complexity is asymptotically dominated as

\( \text{Table 1} \)

| No. of threshold points | Actual. \( A \) | Predicted. \( P \) | Relative Error, \( |A - P|/A \times 100\% \) |
|-------------------------|----------------|----------------|-----------------------------------|
| 1                       | 254            | 255            | 0.394                             |
| 2                       | 327.8          | 327.6          | 0.983                             |
| 3                       | 329.0004       | 329.002        | 5.001                             |
| 4                       | 40923.5        | 40920.2        | 0.099                             |

\( ^{2} \) Multi-modal images are ones with an underlying multi-modal global histogram. Though it is possible to take a different approach, in this paper a given number of multiple thresholds is imposed for practical reasons.
in the region of interest, where $n$ is the number grey-levels and $r$ is the number of threshold points. In fact, the complexity is almost exactly

$$\binom{n + r}{r} = \frac{(n + r)!}{r!n!}$$

which is the number of possible partitions formed from $r$ threshold points on $n$ grey levels. Table 1 compares the count recorded when the computation was suppressed (so as to avoid delay) with the count predicted using $n! / r!$.

To see this, represent the partition system as $n$ ones and $r$ zeroes. The zeroes can be placed between the ones to form partitions. The number of ways of selecting $r$ threshold points from the $n + r$ partition system gives the combinatorial formula. When $r$ is small compared to $n$ the combinations grow approximately exponentially.

2.1. Fuzzy entropy function

Fuzzy membership is a way of grading membership of a set where there is statistical uncertainty. This is exactly the case when the image histogram does not display a smooth set of valleys between peaks. The standard $S$-function [13] (Fig. 4) is typically chosen for the purpose of grading. We have taken a generalized form of this equation [14]:

$$\mu(x) = S(x, a, b, c) = \begin{cases} 0, & x \leq a \\ \left( \frac{x - a}{b - a} \right)^2, & a \leq x \leq b \\ \frac{c - x}{c - b}, & b \leq x \leq c \\ 1, & x \geq c \end{cases}$$

$$s \geq c$$

[7]
This function is symmetrical only when the cross-over point \( b \) lies exactly in the middle of the fuzzy region \((a, c)\) (i.e. \( b = a + c / 2 \), bandwidth \( \Delta b = b - a = c - b \) and \( k = 2 \); see Fig. 5). From the point of view of a two-tone image, a grey-level image is fuzzy. If we had

\[ a \text{ approaches } R_j \]

a method of measuring fuzziness, then it might be possible to find a segmentation point that would minimize the fuzziness. Notice that this assumes a first-order Markovian model for the image field [15], i.e. we expect a smooth transition of grey-levels with the grey-level value of a particular pixel influenced only by the pixels acquired immediately before. It is possible to apply Shannon's function (Eq. 8), which is Eq. (2).
with \( n = 2 \) to the membership function for a particular bandwidth.

\[
S_n(x) = -\mu(x) \log_2 \mu(x) - (1 - \mu(x)) \log_2 (1 - \mu(x)), \quad 0 \leq x \leq n - 1 \tag{8}
\]

A fuzzy entropy measure is now given by

\[
H_{\text{fuzzy}}(4) = \frac{1}{n \ln 2} \sum_{i=1}^{n} S_n(p_i(S_i)) \tag{9}
\]

where \( n \) is the fuzzy set of concern containing \( n \) members. The equation has been normalized because this is not a measure on a probability distribution. Because the set records ‘possibilities’ and not ‘probabilities’, it is no longer viable to use relative entropy in a manner analogous to Eq. (3), because otherwise an equalized histogram would result in maximum entropy. To apply Eq. (9) to an image histogram \( H \), with \( n \) grey-levels within the fuzzy region \( g_i \) and with \( h_i \) pixels in the \( i \)th histogram slot, use:

\[
H_{\text{fuzzy}}(H) = \frac{1}{n \ln 2} \sum_{i=1}^{n} S_n(p_i(g_i))h_i. \tag{10}
\]

The maximum contribution to Eq. (10) is generated by values for \( p_i = 0.5 \), with the contribution falling off symmetrically on either side of this. We require the situation where the contribution from \( p_i = 0.5 \) is at a minimum. The result of applying this measure to a grey-level probability distribution is illustrated in Fig. 6, where the correct identification of a thresholding point is established. Fuzzy entropy is an absolute measure. Thus, bandwidth is varied to bracket the response to differing object sizes in differing images.\(^3\)

---

\(^3\) If an object were to be reduced in area within an image, the amplitudes of the matching grey level distribution would also be reduced.

3. Multi-level thresholding algorithm

This section describes the steps in arriving at a suitable segmentation of the image. A useful preliminary is to perform histogram smoothing with a 3-D Gaussian in order to minimize the possibility of detecting spurious peaks or valleys. The next step of the algorithm is a peak search for humps in the image histogram (see Fig. 7). We have utilized a histogram clipping technique which is a common method when applied to contrast stretching [15,16]. Initially, we clip the histogram data by 5% of the highest peak available in the histogram distribution so as to eliminate small peaks. Then we scan the histogram from 0 to \( n - 1 \) where \( n \) represents the maximum number of available grey levels to locate all valleys at that clipping level, recording the valley start and end locations. This process is repeated by successively increasing the sampling level by 1% up to the maximum sampling level, say 98% of the highest peak. Once a table of peaks at each level is established, the first level at which the number of peaks is below a set maximum is utilized. Extra check scans also be carried out to isolate true and false peaks valleys using figures of merit based on the separation between any two peaks and the strength of the individual peaks, which is similar to the usage in the paper by Sahasrabudhe et al. [2]. This will help to eliminate spurious peaks by merging two peaks with a poor valley into a single clear peak. Experiments were conducted on a variety of histogram data distributions and fairly good results have been achieved. A sample histogram and the number of peaks found by our peak-search method are shown in Figs. 8 and 9.

It is now possible to switch between threshold location methods. For bi-modal images, an averaging technique\(^4\) may be most appropriate, in terms of computation times and subjective results. When it is possible to supply the number of segments desired, then the maximum-entropy

---

\(^4\) The averaging method finds the arithmetic mean of the histogram peaks. Its ‘floor’ becomes the threshold value.
method may be selected. This method will not work in practice for segmentations above five (i.e. where there are four threshold points) because the computation time, on for instance a SUN workstation, is a matter of hours. The algorithm used to arrive at all possible partitions is illustrated in Fig. 10 for four segments. Points A and E in the figure act as boundary markers. Partition point D is repeatedly incremented towards marker E. The other partition points are incremented once, as required, for each occasion that D reaches E. Our first implementation recalculated the entropy (the multi-modal version of Eq. (3)) for each position of D. A second implementation only recalculated the entropy when D reached E, and then only for those segments for which the partition points had moved. When D moves otherwise, it uncovers just one grey-level probability to
its left. The segment to the right must be adjusted for the loss and the segment to the left must be adjusted for a gain. We call this algorithm the running entropy method, which is shown for the case of two partition points in Fig. 11.

The fuzzy entropy method will find a number of peaks depending on the bandwidth (Figs. 12-15). However, we first restricted the bandwidth range by consideration of the known number of peaks as given by the histogram clipping state. It is necessary to restrict the valley search region to the start position of the first peak and the end position of the last peak found at the clipping stage. This measure ensures that the fuzzy valleys match the clipping stage peaks. A heuristic is to use the arithmetic mean of the lowest and highest suitable bandwidths as the selected fuzzy entropy curve. The fuzzy entropy valleys are found simply by finding where $e(k-1) > e(k) < e(k+1)$, for successive discretely-sampled values of fuzzy entropy, $e(k)$, corresponding to each starting position of the bandwidth window. In the case of the search for a maximum (when the search procedure is reversed for the maximum entropy method), a 5% tolerance may be applied after the first maximum has been found. The intention is to avoid finding multiple maxima.

Even if we repeat for a number of bandwidths, the fuzzy-entropy algorithm has calculation complexity $O(n)$. The Shannon function transformation of the $S$-function is performed once for each bandwidth. The maximum entropy method is of exponential order because it does not use the extra grading information. The entropy must be re-computed for each tentative position of the thresholds and for the whole image histogram. Each re-calculation involves costly logarithmic functions. However, it is possible to reduce the computation times by only adjusting the totals by the single value that changes for each partition point.

4. Parallel algorithms and implementation

The proposed algorithm has been implemented in parallel on two different platforms. The software was developed and tested on SUN4 workstations under the Parallel Virtual Machine (PVM) [17] inter-processor communication kernel. It was then implemented on Parrys Supernode machine (SN10000) [18]. The advantage of this procedure is that PVM on the SUNs provides a robust testing environment, while the isolated environment of the transputer machine gives good speed-ups for communication intensive applications. There is little difficulty swapping the communication primitives of PVM for those of 3L's Parallel "C" [19].

The image is divided into equal strips and distributed to $t$ tasks, where $t = 2^z$ for integer $z$. The tasks are connected in a uni-ring topology [20] (Fig. 16). Each task performs a local grey-level histogram. To avoid centralizing the local histograms, local histograms are combined in the manner given in Fig. 17, where the numbers inside the task boxes represent the individual local histograms. Each task is assigned a parity derived from its identity. The parity is used to sequence the order of exchanges. While this method is potentially not as efficient as using a binary-recursive method of globally forming the complete histogram, it has the merit of being scalable on a fixed-valency processor. Where averaging or fuzzy-entropy is used, the short-cuts of the sequential calculation phase made it not worth attempting to parallelize; instead, each task performs an identical calculation to find the thresholds.

Where the maximum entropy method was selected, the overhead involved in exchanging results is small in comparison to the computation time. Additionally, the

---

1 Too small a bandwidth will be influenced by histogram oscillations and too large will ignore relevant changes.

Fig. 8. Sample image histogram distribution.

Fig. 9. Number of peaks in a sample image.
7. Discussion and conclusion

We have examined the idea of using entropies as a method of thresholding, especially in regard to the question of whether fuzzy entropy represents an improvement on the use of information theoretic entropy. We used a maximum entropy method which is claimed as an improvement on previous entropy measures [4]. The theory of entropy, and its variant, fuzzy entropy, have been outlined; fuzzy entropy uses an extra grading step to reduce the calculation involved in forming a thresholding partition. In contrast to the maximum entropy method, which maximizes the information content provided by a suitable partition, fuzzy entropy finds the partition which makes the fuzzy set given by the partition grey-level members as 'crisp' as possible. We have provided details of robust procedures, including smoothing by a linear Gaussian filter, a clipping method for histogram...
topology is already set to exchange global results. Each task finds the maximum entropy of its set of entropies, formed by taking every rth value. The global maximum entropy is now found by a similar exchange of results as used to form the global histogram. In this case, each task compares its local maximum with the new maximum and relays the larger of the two maxima. The final phase for all versions of the thresholder is to apply the segmentation in parallel to each image strip.

A suitable decomposition of the partitioning scheme in the maximum entropy algorithm is shown in Fig. 18. For simplicity of exposition, each task is shown with its first partition point, \( t \), where the \( i \) index the tasks, starting at equal intervals through the range of grey levels. While the \( t \) are confined in scope, further partition points must occupy all possible positions beyond \( t \), to the last grey-level position. With this arrangement it is possible to test all possible partition

---

**Fig. 11.** The ‘running’ method of calculation.
points. Other arrangements are possible, such as starting all but one of the partition points at the same positions as for the sequential method. The last partition point samples their range of points in interleaved fashion. The disadvantage of this other method is that it is no longer possible to use the running entropy method described in Section 3 because the partition points uncover more than one grey level when passing over the histograms.

For the method chosen for implementation, if we assign the $t$ at equal intervals then the parallel algorithm is not load-balanced. To correct this a recursive load-balancing assignment method may be possible, rather than an exhaustive search. Suppose that the task with the largest workload, task 0, reduces the size of its interval by $k$, giving an interval length of $n/t-k$ with $t$ the number of tasks (Fig. 19). Each of the other tasks will perform at least the same amount of calculation (in parallel). Confining attention to where two partition points are used and compare the work performed to that of a task in the most favourable position, $t-1$. The extra computation performed by task 0 is $(n/t-k)(n(t-1)/t+k)$. The other tasks, considered in aggregate, will have gained work because of the extension of their region by $k$. The additional work performed, per task, is given by

$$
\frac{1}{t-1}(\lfloor nx+k \rfloor + \cdots + 1 - \lfloor nx/2 \rfloor + \cdots + 1)
$$

where $x = (t-1)/t$. To balance the load, the extra work done by task 0 must equal the extra work given by expression (11). Therefore, the following equation should be solved for $k$:

$$
\frac{1}{t-1}(\lfloor nx+k \rfloor + \cdots + 1 - \lfloor nx/2 \rfloor + \cdots + 1) = 0
$$

which is quadratic in $k$. Once we have the interval length for task 0, then subtracting this length from $n$ allows the equation to be reused for $t-1$ tasks. Though this formula has been extended to the case of more than two partition points, the algebra is less tractable and numerical methods are needed to find $k$, which is a
The performance of the maximum-entropy method for three segments is given in Table 2. When the running total method is used, the timings are reduced considerably. The timings for PVM are slightly worse than for the transputer implementation. In Fig. 20 for four segment timings, though the particular SUN4s used for PVM have a lower nominal CPU speed rating (20 MHz as returned by the Unix utility `fcpu` on the Silicon Graphics Workstation) than the T85 transputers (25 MHz), the timings are better for PVM because in part communication times have marginal effect on the final total. Load-balancing is now seen to have a significant effect on the outcome. It is apparent that for four segments the maximum-entropy method timings are still much in excess of those for the fuzzy-entropy multi-thresholding method.3

Where the fuzzy method is performed in parallel, the histogram formation and the final step of applying the segmentation are implemented in parallel, with the intermediate step being replicated by each task. As mentioned in Section 3, a range of suitable bandwidths is tested in the intermediate step. The extra communication involved in parallelizing the intermediate steps would be counter-productive in view of the overall time for four SUN4 workstations utilized under PVM.

![Diagram](image.png)

Fig. 16. The un-ring topology used for global exchange of results.

5. Evaluation of results

While single-thresholded images can be assessed subjectively, assessment of multi-thresholding is partly dependent on the assignment of grey level to each segment. The visual appearance is also of importance for robotic applications. To provide an objective assessment, we used two measures of the thresholded image: uniformity and shape. These measures are loosely based on those in the papers of Kapur et al. [4] and Levine et al. [21], respectively.

Uniformity = 1 - \( \frac{\sum_{x,y}(f(x,y) - \mu)^2}{B} \)

where \( B \) is the thresholded region of concern, \( f \) returns the grey level, \( \mu \) is the mean grey level within the region.
and $R$ is a normalizing factor based on the region area and its grey-level range. The shape measure was:

$$\text{Shape} = 1 - \frac{\sum (G - I)}{C}$$  \hspace{1cm} (14)

where $R$ is the threshold region of concern, $G$ is a neighbourhood gradient measure, $I$ is the absolute difference between the grey-level of the pixel of concern and its 8-neighbourhood grey-level mean and $C$ is a normalizing factor based on the region area and its grey-level range.

The boundary of the image is omitted in this measure as are pixels not in the interior of the thresholded region. $G$ is given by

$$\sum_{i=1}^{4} D_i^2$$  \hspace{1cm} (15)

with

$$D_1 = f(x + 1, y) - f(x - 1, y)$$

$$D_2 = f(x, y + 1) - f(x, y - 1)$$

$$D_3 = f(x - 1, y - 1) - f(x + 1, y + 1)$$

$$D_4 = f(x + 1, y - 1) - f(x - 1, y + 1)$$

A sample of the results of the tests are given in Table 3 for two images taken from a number of others: a multi-modal image (Fig. 21) and a bi-modal image (Fig. 22), where the latter image was captured using 8-bit sampling. As a point of comparison, in the paper by Sezan [22] thresholding points for Fig. 21 were reported as 47, 96 and 152. If the fuzzy-entropic search area is extended to include the higher grey-levels, it finds a threshold point at 155 as well as those in Table 3. Sample thresholded images are given in Figs. 23, 25. Though the absolute figures are in part a result of the normalizing factor used, it might be observed that there is little to gain from using other than an averaging method on a bi-modal image (Figs. 26 and 27). Additional images, for the reader’s perusal, are included in Figs. 28, 37, when the histogram for Fig. 33 is already included as Fig. 8. The fuzzy-entropic method is not as good as the purely entropic method when used on the multi-modal image, but this distinction is naturally only good for the two images surveyed here. In general, shape appeared a better discriminator in separating out the better performance of the maximum-entropy method for multi-modal images. Further tests revealed a weaker response on the part of the entropic methods when Gaussian noise was added to the images. This may be because the noise has the effect of confusing the underlying distribution of grey-level
Table 3
Comparison between various thresholding methods (bracketed numbers indicate the rank of each method, found by averaging the ratings)

<table>
<thead>
<tr>
<th>Image</th>
<th>Threshold Range</th>
<th>Unimall</th>
<th>Shape</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>Averaging</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0-94</td>
<td>0.408</td>
<td>0.707</td>
</tr>
<tr>
<td></td>
<td>94-255</td>
<td>0.790</td>
<td>0.812</td>
</tr>
<tr>
<td></td>
<td>2-segment entropic</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0-141</td>
<td>0.932</td>
<td>0.836</td>
</tr>
<tr>
<td></td>
<td>141-255</td>
<td>0.812</td>
<td>0.844</td>
</tr>
<tr>
<td></td>
<td>3-segment entropic</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>0-80</td>
<td>0.705</td>
<td>0.728</td>
</tr>
<tr>
<td></td>
<td>80-149</td>
<td>0.681</td>
<td>0.705</td>
</tr>
<tr>
<td></td>
<td>149-255</td>
<td>0.659</td>
<td>0.809</td>
</tr>
<tr>
<td>Bandwidth 31</td>
<td>Fuzzy-entropic</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>0-30</td>
<td>0.326</td>
<td>0.595</td>
</tr>
<tr>
<td></td>
<td>30-63</td>
<td>0.294</td>
<td>0.488</td>
</tr>
<tr>
<td></td>
<td>63-96</td>
<td>0.705</td>
<td>0.646</td>
</tr>
<tr>
<td>Lena</td>
<td>Averaging</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0-109</td>
<td>0.940</td>
<td>0.930</td>
</tr>
<tr>
<td></td>
<td>109-255</td>
<td>0.132</td>
<td>0.934</td>
</tr>
<tr>
<td></td>
<td>2-segment entropic</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>0-72</td>
<td>0.932</td>
<td>0.830</td>
</tr>
<tr>
<td></td>
<td>72-255</td>
<td>0.796</td>
<td>0.926</td>
</tr>
<tr>
<td></td>
<td>3-segment entropic</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>0-76</td>
<td>0.933</td>
<td>0.823</td>
</tr>
<tr>
<td></td>
<td>76-183</td>
<td>0.639</td>
<td>0.667</td>
</tr>
<tr>
<td></td>
<td>183-255</td>
<td>0.359</td>
<td>0.953</td>
</tr>
<tr>
<td>Bandwidth 40</td>
<td>Fuzzy-entropic</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>0-120</td>
<td>0.931</td>
<td>0.980</td>
</tr>
<tr>
<td></td>
<td>120-154</td>
<td>0.560</td>
<td>0.464</td>
</tr>
<tr>
<td></td>
<td>154-255</td>
<td>0.585</td>
<td>0.948</td>
</tr>
</tbody>
</table>

6. Extension to the fuzzy-entropic method

Examining Eq. (7) shows that $\lambda$, which is dependent on $h$, governs the symmetry of the function. Figs. 38 and 39 show the effect of varying $h$. When different choices of $h$ were made and applied to actual images, the choice of $h$ was found to make little difference to the thresholding points chosen, as Table 4 shows. This is because altering the position of $h$ changes the effective bandwidth, by projecting more gray-levels either towards zero or one as a fuzzy grade, where they contribute little to the entropy measure. Another technique involving localized fuzzy regions was used on the histogram shown in Fig. 8. Clipping points were obtained for either side of each valley, so as to define a set of fuzzy regions for each valley. Values of fuzzy entropy were obtained by varying the crossover point across the fuzzy region. Fig. 40 illustrates the resulting local entropy curves, where the valley of each curve is close to the observed floor of each histogram valley. For Fig. 21, thresholding points were found at 46, 80 and 112. Further investigation will reveal whether the threshold points that were obtained represent a more meaningful segmentation.

Fig. 20: Timing as the number of processors used is varied for the maximum-entropy algorithm, four segments.

Fig. 21: Original Lena image, size 512 x 512.
Fig. 23: Faces entropy thresholded Lena image with bands
Fig. 24: Maximum entropy method threshold of Lena image using three segments
Fig. 25: Fuzzy-entropy thresholded objects image with bands
Fig. 26: Average method threshold of objects image
Fig. 27: Maximum entropy method threshold of objects image using two segments
Fig. 28: DNA image, size 512 x 512 [24]
7. Discussion and conclusion

We have examined the idea of using entropies as a method of thresholding, especially in regard to the question of whether fuzzy entropy represents an improvement on the use of information theoretic entropy. We used a maximum entropy method which is claimed as an improvement on previous entropy measures [4]. The theory of entropy, and its variants, fuzzy entropy, have been outlined; fuzzy entropy uses an extra grading step to reduce the calculation involved in forming a thresholding partition. In contrast to the maximum entropy method, which maximizes the information content provided by a suitable partition, fuzzy entropy finds the partition of which makes the fuzzy set given by the partition grey-level members as 'crisp' as possible. We have provided details of robust procedures, including smoothing by a linear Gaussian filter, a clipping method for histogram
peak search and the method of selecting a suitable fuzzy bandwidth. The multi-level maximum entropy method will normally only be used for up to four segments because of the computational burden. We have devised a running total method for computing the maximum-entropy function, resulting in substantial computational savings. It is possible to successfully parallelize this algorithm, which is a species of combinatorial search, by means of a novel breakdown of a partitioning problem. Other multi-thresholding techniques, which use a similar partitioning strategy [23], should also benefit from this parallelization method. However, parallelization will not enable it to compete in computational time with the fuzzy method. Both of these entropic methods should be applied to images with the number of modes greater than two. Moreover, confidence in the fuzzy-logic method will only be general if the question of choice of membership function and bandwidth can be systematically resolved. In practice, so long as the choice is sensibly restricted, the precise choice is not significant. We have restricted attention to global methods because they are most suitable for automatic use, but it is possible to use local discrimination when applying to global threshold, perhaps by a fuzzy measure such as compactness, as outlined in work by Pal and Rosenfeld [6].

![Thresholds obtained for Fig. 8 by varying crossover point A.](image)

**Table 4**

<table>
<thead>
<tr>
<th>Threshold point (Luma)</th>
<th>Value of A relative to unit bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.25</td>
</tr>
<tr>
<td>1</td>
<td>65</td>
</tr>
<tr>
<td>2</td>
<td>95</td>
</tr>
<tr>
<td>3</td>
<td>135</td>
</tr>
</tbody>
</table>

Acknowledgement

This work was carried out as part of project HED-3-12711 (Parallel Reconfigurable Image Processing Systems).

**References**


