An Adaptive Multi-Thresholding Technique for Binarization of Color Images

LIAQUAT MAJEED SHEIKH, IMTIAZ HASSAN, NAVEED ZAFAR SHEIKH, RANA AMIR BASHIR, SHAHZAD AMJAD KHAN, SALMAN SAEED KHAN
Department of Computer Science
National University of Computer and Emerging Sciences, Lahore, PAKISTAN.

Abstract: - Applying binarization technique on a colored image can yield an image that is quite different from the original one if multi-thresholding is used. The solution proposed is an adaptive multi-thresholding technique (AMTT) that describes an adaptive multi-thresholding technique (AMTT) for binarization of color images depending upon the nature of the image. The technique improves the perception of the binarized images based on color hue and hence the dissimilarity between original and binarized image is reduced. This technique compensates for differences in illumination and shade by including information content in the thresholding calculation. AMTT calculates information loss in the image with respect to human perception on the basis of color hue. Experimental results show that the proposed AMTT reduces the color content by keeping the information loss minimum.

Key-Words: - binarization, thresholding, multi threshold, perception, color histogram, and information loss

1. Introduction

Multi-thresholding of real images is based on their concept of human perception. In this research RGB color scheme was used. The pixels in an RGB color scheme can vary in combinations to produce 256*256*256 colors i.e. more than 16 million colors. However, human perception is the ability to discriminate between different colors. For example there may be thousands of RGB combination, which in fact are different shades of yellow, but humans will generally regard them as "yellow". The inability to discriminate between different shades of color and regard them as a single color is called "color generalization". Color generalization is important in the identification of regions since they may contain different shades of the same color. A human would perceive it as one region but if no color reduction has been performed it would become very difficult to set a threshold based on dissimilarity in color between individual pixels to either include them in the region or not. Color generalization can reduce this effect by allowing the threshold to be the exact match.

2. Color Codebook

Color Codebook [1] is based on the clustering of image pixels of similar colors. Calculating the Euclidian Distance of the RGB vector with the cluster centroid and comparing it with a defined threshold measure similarity. The exact value of the threshold cannot be generalized and depends upon the image data set.

The color codebook poses two arduous problems that affect its efficiency while providing no fruitful counter benefits. These can be explained as follows:

a. Due to the extensive color variation in real life images, a relatively larger threshold will generate clusters, each of which will comprise of member pixels varying vastly from its centroid based on the calculation of Euclidian Distance [2]. As a result, a query regarding some specific color will have, as its output, images containing relatively different color from the one specified in the query. On the contrary, a smaller threshold will not generate those images as its output which have almost similar color to the one specified in the query.
b. Moreover, there are a large number of clusters that contain only a fractional percentage of the pixels of the original real image, for example

0.073%, which is useless to the extent that the user will never be able to find such low percentage color in the image and generate a query on such basis.

3. Binarization Filter or Constant Thresholding

These two problems can be solved to a major extent by using the binarization filter or constant thresholding. For example, a constant thresholding at a threshold value 127 would set the pixel's R, G & B below 127 to 0 and those equal to or above 127 to 255. This produces a binary affect for each of the R, G & B color space producing the total color combinations to only 8. Thus an image of possible more than 16 million colors will be reduced to only 8 colors [3]. It seems to be a remarkable reduction of the color values from 16 million to only 8 but considering the other side of the coin, the information loss associated with this color reduction is unbearable since it is quite possible that the critical image details will be ignored [4]. But it still doesn't completely solve the problem. This can be illustrated by the following primitive examples. Consider the images of this tennis ball:

(a) (b)

Fig. 1: (a) Original Image (b) Binarized Image

In this case, the filtered image has preserved all the regions of the original unfiltered image. Now consider another scenario in which the image contains different shades of the same color:

(a) (b)

Fig. 2: (a) Original Image (b) Binarized Image

Fig. 2 (b) is the binarized image of the above tennis ball. It has failed to retain all the original regions present in that image.

Hence critical image detail has been lost resulting in false region detection. This is because of the fact that the binarization filter does not work properly for shaded regions i.e. the yellowish-green middle region in the unfiltered image.

From this we can conclude that binzarization filter works only if there is no shading in the original image.

4. Multi-Thresholding

Binarization filter or constant thresholding, though considerably reduces the total number of colors, loses critical image information when applied to images having different shades of the same color. The reason behind this information loss is the use of a single threshold, which converts the lighter shades of the same color to white and the darker ones to black. To overcome this discrepancy, we propose a Multi-Thresholding Technique (MTT) which is capable of reducing the possible 256*256*256 (16 million) colors to $\mathbf{(2^{(\log_2 n)+1})^3}$ where n is the degree of multi-thresholding. Each component of RGB vector space is divided into 'n' different ranges, each range having its own threshold. In case n is 1, MTT acts as a binarization filter or constant thresholder having only a single range. For n equal to 128, it generates the same true color real image with 128 different ranges. Each component of the RGB vector of the pixel space is given a new value based on the degree of multi-thresholding according Eq. 1:

n = defines the number of ranges for threshold
X = component values of RGB
R = end point of binarized range (based on'n')
Y = (256/n) * R.

$$X = \begin{cases} Y - (256/n) & \text{if } X < Y/2 \\ Y & \text{if } X >= Y/2 \end{cases} \quad (1)$$

5. Adaptive Thresholding

Defining the Degree of Multi-Thresholding (DMT) is a critical issue. A fixed DMT might work well for a certain set of images but it will be subjected to the same problems of information loss as suffered by constant thresholding. A smaller DMT will lose information due to different shades of the same color. Contrary to that, a larger DMT value will not eliminate colors with negligible presence in the image and will result in the production of useless color regions. This leads us to a situation where the adaptive nature of the degree of multi-thresholding becomes inevitable. Hence, the DMT is adapted according to the nature of the image being binarized. Adapted Multi-Thresholding Technique (AMTT) starts binarization with the lowest degree of DMT and heads towards a value where the information loss (IL) becomes ***"acceptable"***. The acceptability of information loss is determined by calculating the difference between the information loss of the current DMT and that of the previous DMT. When this difference becomes equal to or smaller than the maximum possible error (ε), the information loss is considered acceptable and the corresponding DMT is the adapted DMT, as shown in Eqn. 2. If information loss is greater than (ε), value of DMT increases until information loss becomes less than (ε). The calculation of error defined by (ε) depends upon the constraints of the system in which AMTT is being used. For example, for images where information loss should be minimum then value of (ε) is close to zero and vice versa. The adaptive multi-thresholding algorithm is given in Fig. 3.

$$\left| IL_{new} - IL_{old} \right| \leq \varepsilon \qquad (2)$$

do
{
  for (*each image pixel : **P*** **)**
  {
    for (*each R,G,B of P:* **Pi)**
    {
      Threshold
$$= \left(\left\lfloor \frac{P_i}{RangeSize} \right\rfloor * RangeSize \right) + \frac{RangeSize}{2}$$
      if(*Pi is less than threshold* **)**
$$P_i = threshold + \frac{RangeSize}{2}$$
      else
$$P_i = threshold - \frac{RangeSize}{2} \quad \}$$
    }
  }
if (*Information Loss is not Acceptable* **)**
$$RangeSize = \frac{RangeSize}{2}$$
}while (*Information Loss is not Acceptable* **)**

Fig. 3: Adaptive Multi-Thresholding Algorithm (AMTT)

6. Information Loss Calculation

The major consideration in implementing this algorithm is the calculation of information loss. The perception of colors by the human eye, though makes it fairly simple to observe the information loss in the image of Fig 2, but implementing human perception is a very difficult task if not impossible. AMTT uses an attribute, ***hue***, which defines the image in a way closest to the human perception. Hue represents different shades of the same color as a single dominant color in the same way as human eye perceives them as a dominant wavelength. Generally, hue can be defined as a 360^0 circle, each degree representing a unique hue value [5]. There exist 12 distinct regions of 30^0, each of which contains varying shades of a single color, as shown in Fig. 4.

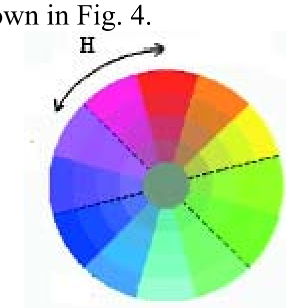

Fig. 4: Varying Shades of a Single Color and Hue

The information loss is determined by calculating the difference between the hue values of the same pixel before and after applying the multi-thresholding algorithm. If the hue value of a pixel does not lie in the same hue region as was before, the information in that pixel is considered lost. The number of lost pixels in relation to the total image pixels gives the overall information loss associated with the binarized image. The choice whether the information loss is acceptable or not depends upon the system using AMTT.

7. Results and Discussion

We applied AMTT on the images in Fig. 1 (a) and Fig. 2 (a). The AMTT starts with DMT equal to 1 and calculates the information loss. It then proceeds for DMT equal to 2 and calculates the information loss. If the information loss has increased, it takes the previous DMT as acceptable. If, however, the information loss has decreased then it calculates the difference between the previous and current information losses and compares it with a threshold. If the difference is smaller than or equal to the threshold, the previous DMT is taken as acceptable. The results are shown in Fig. 5 and Fig.6 respectively.

(a) (b)

Fig. 5: (a) Original Image (b) Binarized Image Through AMTT

(a) **(b)**

Fig. 6: (a) Original Image (b) Binarized Image Through AMTT

It was found that the information loss after applying AMTT to Fig. 1 (a) at DMT equal to 1 was acceptable. Comparing Fig. 1 (b) and Fig. 5 (b) it appears that there is no difference in the results between constant thresholding and multi-thresholding for this particular example. The histogram of the image before and after applying AMTT to Fig. 1 (b) is shown in Fig. 7. The binarized image histogram shows that in contrast to the original image histogram, the R, G & B components of the pixels have accumulated at the extreme ends i.e. at 0 or 255.

Applying AMTT to Fig. 2 (a) it was found that at DMT equal to 6 the information loss was acceptable. Comparing Fig. 2 (b) and Fig. 6 (b) it appears that there is a great difference in the results between constant thresholding and multi-thresholding for this particular example. The histogram of the image before and after applying AMTT to Fig. 2 (b) is shown in Fig. 8. The binarized image histogram shows that in contrast to the original image histogram, the R, G & B components of the pixels have accumulated at the extreme ends of the six thresholds.

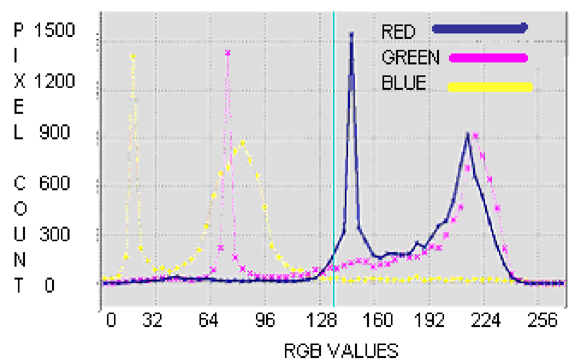

Fig. 7 (a): Histogram of the Original Image given in Fig. 1 (a)

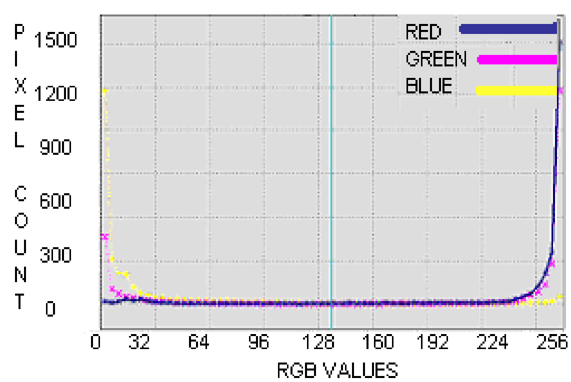

Fig. 7 (b): Histogram of the Image given in Fig. 1 (a) after Binarizing Through AMTT

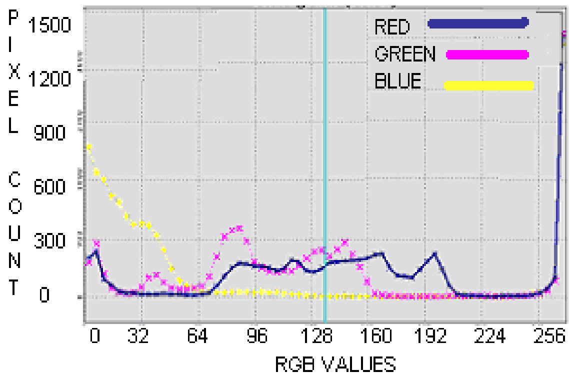

Fig. 8 (a): Histogram of the Original Image given in Fig. 2 (a)

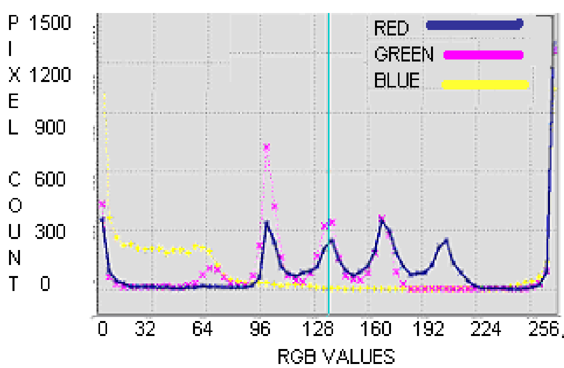

Fig. 8 (b): Histogram of the Image given in Fig. 2 (a) after Binarizing Through AMTT

Tables 1 and 2 show that the total information loss after applying AMTT to Fig. 1 (a) is 35% and that of Fig. 2 (a) is 24%.

8. Conclusion

Adaptive multi thresholding, which is an alternative to constant thresholding, reduces the color content of the image that remains perceivable for human eyes. Experimental studies show that the proposed AMTT reduces the color contents to the acceptable range by minimizing the information loss. The ignorable error difference (ε) varies according to the nature of the image and hence leaves the prospect of further research on determining the exact value of ignorable error.

Table 1:Information Loss for Fig. 1 (a) after applying AMTT

CR	%age loss	PCOI	PCBI
1	0	49	3480
2	100	2441	0
3	11	7340	6575
4	100	15	0
5	0	0	0
6	100	2	0
7	0	0	0
8	0	0	0
9	100	23	0
10	100	120	0
11	100	58	0
12	100	52	0
Total Loss	**35 %**		

CR = the current hue region
PCOI = pixel count of the original image in CR
PCBI = pixel count of binary image in CR

Table 2:Information Loss for Fig. 2 (a) after applying AMTT

CR	%age loss	PCOI	PCBI
1	17	4674	7026
2	24	4397	5196
3	17	11428	9763
4	100	215	9
5	98	104	17
6	100	151	0
7	100	316	0
8	100	126	0
9	94	284	34
10	100	116	0
11	100	175	7
12	100	66	0
Total Loss	**24 %**		

CR = the current hue region
PCOI = pixel count of the original image in CR
PCBI = pixel count of binary image in CR

References:
[1] Wei-Ying Ma, B.S. Manjunath, NeTra: A toolbox for navigating large image databases, Multimedia Systems, Springer-Verlag 1999, 7: pp 184–198,
[2] Pedro F. Felzenszwalb, Daniel P. Huttenlocher, Image Segmentation Using Local Variation, Conference on Computer Vision and Pattern Recognition (CVPR '98), June 23-25, 1998, Santa Barbara, CA, USA, pp 98-104
[3] Image tester version 1.1, www.swiftgear.com
[4] Ji Zhang, Wynne Hsu, Mong Li Lee, An Information-driven Framework for Image Mining, Proceedings of 12th International Conference on Database and Expert Systems Applications (DEXA), Munich, Germany, September 2001.
[5] Timothy K. Shih, Jiung-Yao Huang, Ching-Sheng Wang, Jason C. Hung and Chuan-Ho Kao, An Intelligent Content-based Image Retrieval System Based on Color, Shape, and Spatial Relations, Proceedings of the National Science Council, ROC, Part A: Physical Science and Engineering, Vol. 25. No. 4, pp. 232 -- 243, May 2001.