A Thresholding Method for Color Image Binarization

Kalavathi P
Department of Computer Science and Applications
Gandhigram Rural Institute – Deemed University, Gandhigram, Tamil Nadu, India

Abstract

In this paper, a color image binarization method based on isodata thresholding technique is developed. A threshold value is computed separately for each of the color component of the given input RGB image. The binary images of each color component image along with the binary image of the converted grayscale image are added together to produce the resultant binary image. The result of this proposed method was compared with the popular Otsu's thresholding method. The proposed method found to produce better result than the Otsu's thresholding technique.

Keywords: Color image binarization, thresholding technique, isodata thresholding, image binarization, color image segmentation.

1. Introduction

Image binarization is an essential step in image segmentation process because the binary images are easier to process than the grayscale or color image for object analysis and identification. Image binarization is typically treated as a simple intensity thresholding operation on an image. There are many binarization methods being proposed [1]-[9]. Thresholding creates a binary image from gray scale/color images by turning all pixels below the threshold to zero representing the background and all pixels above the threshold to one for foreground representation. Because of its efficiency in performance and its simplicity, thresholding techniques have been studied extensively and a large number of thresholding methods were developed [10]-[12]. Most of these methods find a single threshold value. Sometimes, for color images, this single threshold may not produce correct result, because the color image may be dominated by some color features that the single threshold may fail to segment it correctly. Therefore, in this proposed method, the threshold value is calculated for each of its color component image separately. The result of the proposed method is compared with global thresholding method Otsu.

Otsu is one of the well known and widely accepted methods [13] for gray scale images. It is based on discriminant analysis to find the maximum separability of classes and is used to automatically perform histogram shape-based image thresholding [14]. The algorithm assumes that the image is composed of two basic classes, foreground (C_0) and background (C_1) and it constructs a normalized histogram using the discrete probability density function, and is given by:

$$p_r(r_q) = \frac{n_q}{n}, \; q = 0,1,2,...,L-1 \tag{1}$$

where, n is the total number of pixels in the image, n_q is the number pixels that have intensity level r_q and L is the highest intensity level in that image. Initial threshold is the midpoint between the maximum and minimum intensity values in the image. If k is chosen as the initial threshold then C_0 is the set of pixels with levels $[0,1,...,k-1]$ and C_1 is the set of pixels with levels $[k, k+1, \cdots, L-1]$. Otsu's method chooses the threshold value k that maximizes the between-class variance $\sigma^2{}_B$, which is defined as:

$$\sigma^2{}_B = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 \tag{2}$$

where,

$$\omega_0 = \sum_{q=0}^{k-1} p_q(r_q) \tag{3}$$

$$\omega_1 = \sum_{q=k}^{l-1} p_q(r_q) \tag{4}$$

$$\mu_0 = \sum_{q=0}^{k-1} q p_q(r_q) / \omega_0 \tag{5}$$

$$\mu_1 = \sum_{q=k}^{L-1} q p_q(r_q) / \omega_1 \tag{6}$$

$$\mu_T = \sum_{q=0}^{L-1} q p_q(r_q) \tag{7}$$

In this method, a thresholding technique based on isodata method is developed for color image binarization. This proposed method is applied to find the threshold value for each of its color component images of the given input image. The binarized output image is obtained by combining the binary images of all the three color component images along with the binary image of the converted gray scale image.

The remaining part of the paper is organized as follows: In section 2, the methodological details of this proposed method is given. The results and discussion are given in section 3 and the conclusions is given in section 4.

2. Method

In the proposed method, the input RGB color image is separated into three images; red component image R, green component image G and blue component image B respectively. The proposed thresholding technique is then applied on each of these images in order to obtain the binary image. These three binary images along with the binary image of the gray scale converted input RGB image are added together to produce the resultant image. The flowchart of the proposed method is given in Figure 1.

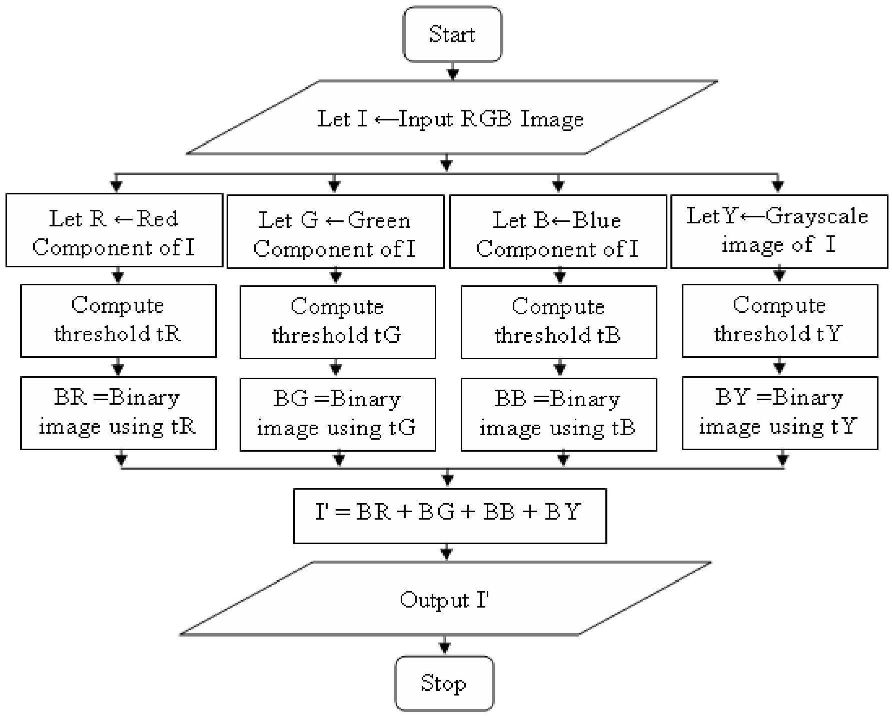


Figure 1: Flowchart of the proposed method

In this method, threshold value is calculated based on isodata thresholding technique [15]. Isodata method is an iterative self-organizing data analysis technique. It is an unsupervised classification method and do not need to know the number of clusters. The initial threshold is periodically updated by splitting and merging the clusters. Cluster centers are randomly placed and pixels are assigned based on the shortest distance to its center. The standard deviation and the distance between the clusters are calculated. Clusters are split, if one or more standard deviation is greater than the user-defined threshold. Clusters are merged, if the distance between them is less than the user-defined threshold. Then it starts the second iteration using the new cluster centers. This process is repeated until the average change in the inter-center distance between iterations is less than the threshold (or) the maximum number of iterations is reached.

The proposed method identifies the initial threshold by calculating the histogram of an image. Then it uses an isodata based method to update the initial threshold. The histogram of digital image with the intensity levels in the range [0, L-1] is defined as a discrete function:

$$h(r_k) = n_k \tag{8}$$

where, r_k is the intensity value, n_k is the number of pixels in the image with intensity r_k and $h(r_k)$ is the histogram of the image with gray level r_k.

Then the initial threshold T_0 is calculated using the following equation:

$$T_0 = \frac{\sum (h(r_k) \times n)}{\sum h(r_k)} \tag{9}$$

where, n is the bin location. The n of the histogram are each half-open intervals of width $A/(n-1)$. In particular, the p^{th} bin is the half-open interval and is given below:

$$\frac{A(p-1.5)}{(n-1)} \leq n \leq \frac{A(p-0.5)}{(n-1)} \tag{10}$$

The scalar factor A depends on the image class. In this method, A is 255 because each color component of the given image is an 8-bit intensity image.

Based on the computed initial threshold value T_0, the image is segmented into two images m_1 (foreground) and m_2 (background) as given below:

$$m_1(x,y) = \begin{cases} 1 & \text{if } f(x,y) \leq T_0 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$$m_2(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T_0 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

where, $f(x,y)$ is the input color component image. Then the new threshold T_{New} is calculated as per the Eqn. (13).

$$T_{New} = \frac{mean(m_1) + mean(m_2)}{2} \tag{13}$$

The original image is partitioned based on the new threshold value T_{New}. This process is repeated until the difference between the old threshold T_0 and the new threshold T_{New} is less than 0.01 or it has reached the maximum number of iterations. The summary of the steps involved in the proposed method is given below:

1. Read the input RGB color image I.
2. Let R←Red_Component_Image(I)

Let G←Green_Component_Image(I)
Let B←Blue_Component_Image(I)
Let Y←Grayscale_Image(I)

3. Repeat the following steps for each of these images R, G, B and Y to calculate the threshold values tR, tG, tB and tY respectively
   i. Compute the image histogram
   ii. Calculate the initial threshold T_0 by the Eqn. (9)
   iii. Divide the image into two images background and foreground by the Eqns.(11) and (12)
   iv. Calculate the new threshold T_{New} using the Eqn. (13)
   v. If $T_0 - T_{New}$ <=0.01 or the maximum number of iteration is reached then stop the threshold computation process, otherwise goto step 3 by assigning T_0 as T_{New}
4. Let BR← Binary_image(R) using tR
   Let BG← Binary_image(G) using tG
   Let BB← Binary_image(B) using tB
   Let BY← Binary_image(Y) using tY
5. Obtain the binary image I' = BR+BG+BB+BY
6. Output the resultant binary image I'

3. Result and Discussion

The proposed color image binarization method is evaluated using the color RGB images obtained from the internet. The performance of the proposed method is compared with the well known binarization technique Otsu. The process involved in this proposed method is given in Figure 2. In this figure, the original color RGB image is shown in Figure 2(a) and the corresponding gray-scale image Y, red component image R, green component image G and blue component image B are shown in column 2 of Figure 2(b), 2(c), 2(d) and 2(e) respectively. The binarized image using the proposed thresholding method for the images of column 2 are given in the next column as Figure 2(b') 2(c'), 2(d') and 2(e') respectively. These binary images are added together to produce the resultant binary image and is shown column 4 of the same figure. It can be observed from this figure that by adding the binary images of R, G and B component alone will not produce the correct binary image. Similarly the binary image of the converted gray scale image (Figure 2(b)) has missed some of the foreground image features that are present in the original image. Therefore, in this proposed method, it is devised to add the binary images of all the color component images along with the binary image of gray scale image to produce the final result. It is necessary to add the binary image of each color component image because some features in the color images are dominated by its color component, a common threshold value may not be sufficient to segment all the objects present in the color image. The result of this proposed method for the color image shown in Figure 2(a) is compared with the existing Otsu's method and is represented in Figure 3. For this image the proposed binarization method produced better result than the Otsu's method.

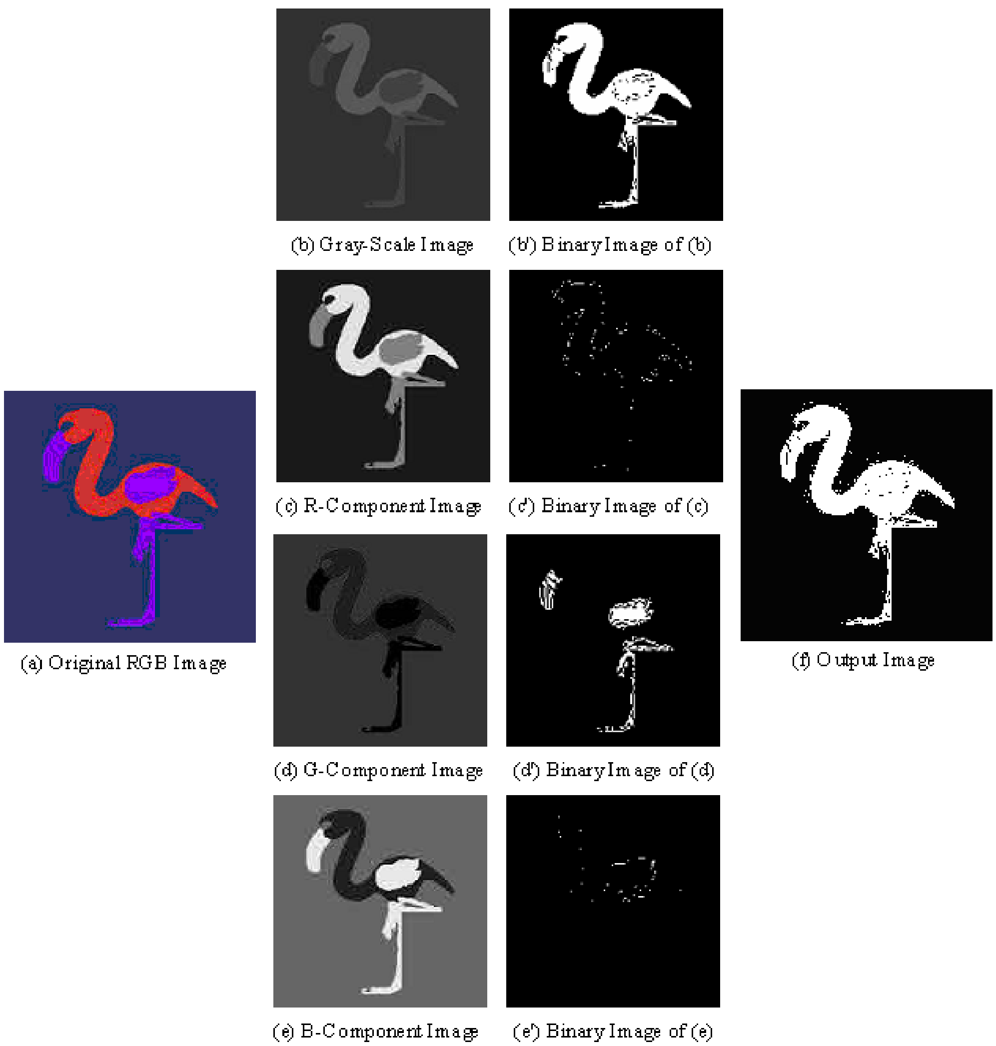
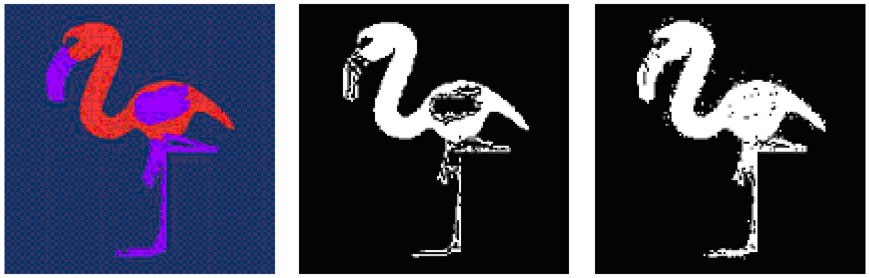


Figure 2: The process involved in the proposed method



(a) Original Image (b) Binary Image by Otsu (c) Binary Image by the Proposed Method

Figure 3: Comparison of proposed with the Otsu's method

For visual comparison some of the selected sample images along with the result obtained by the proposed and the Otsu's method are given in Figure 4. For all these selected images the proposed method produced better result than the Otsu's method. From these results it is observed that the proposed thresholding method may be used as an efficient technique for color image binarization.

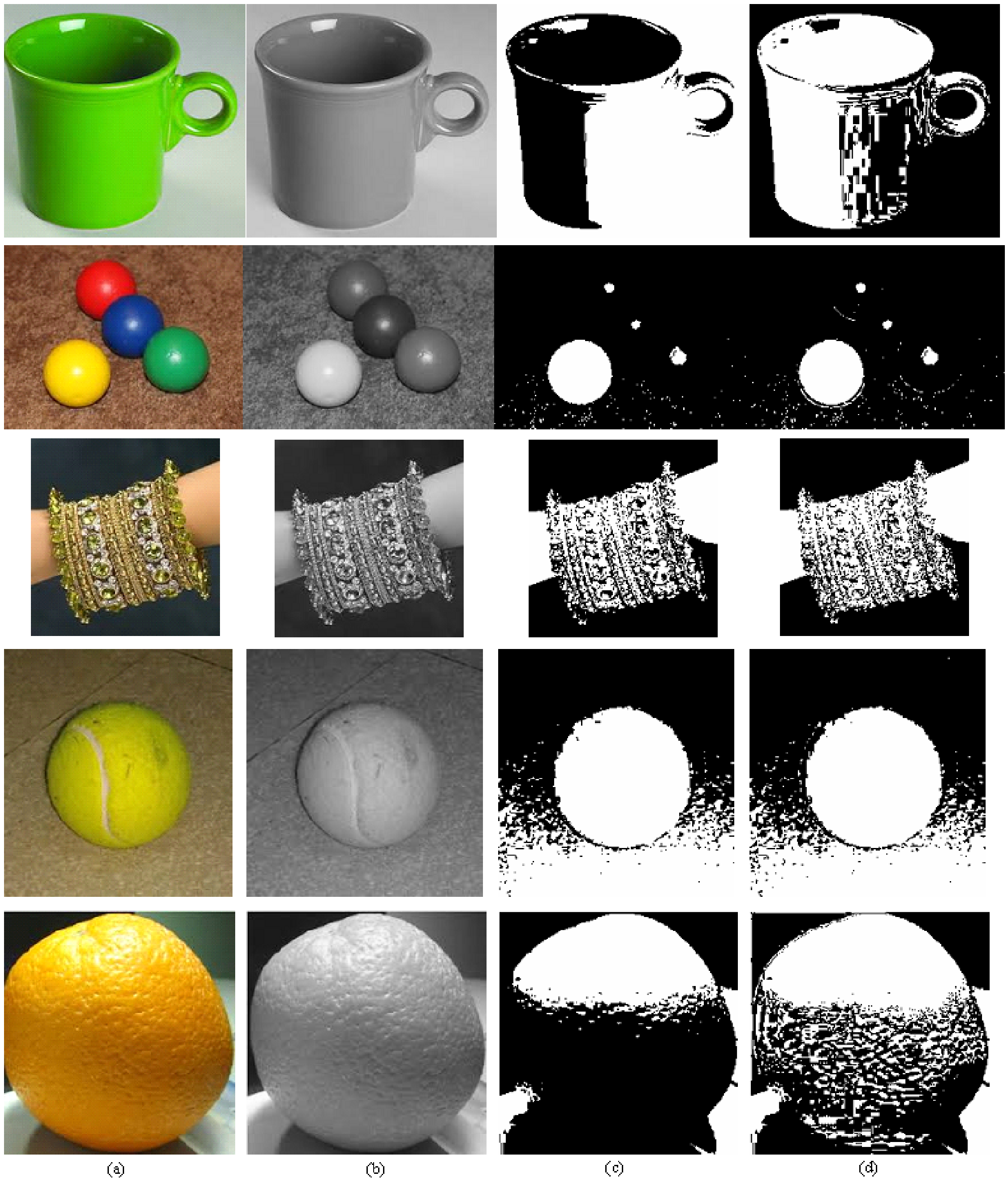


Figure 4: Binarization output of the proposed and the Otsu's method; (a) Original image (b) Gray-scale image, (c) Binary image by Otsu and (d) Binary image by the proposed method

4. Conclusions

In this article, a simple and efficient color image binrization method based on isodata thresholding technique is developed. The results obtained by this method are compared with the popular thresholding technique Otsu. It is found that the proposed method produced better result than the existing Otsu's method. This method may be further standardized to obtain binary image on color images of different color models.

References:

1. Chin-Hsing, C., J. Lee, J. Wang and C.W. Mao, Color Image Segmentation for Bladder Cancer Diagnosis, *Math. Comput. Modeling*, 27, 103-120, 1998.
2. Cheng, Y., Mean Shift, Mode Seeking, and Clustering, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17, 790-799, 1995.
3. Cheriet, M., J.N. Said and C.Y. Suen, A Recursive Thresholding Technique for Image Segmentation, *IEEE Transactions on Image Processing*, **7**, 918-921, 1998.
4. Rodríguez, R., T.E. Alarcón, R. Wong and L. Cuello, Color Segmentation Applied to Study of the Angiogenesis, *Journal of Intelligent and Robotic System*, 34, 83-97, 2002.
5. Somasundaram, K and Kalavathi, P, Medical Image Binarization using Square Wave Representation, CCIS, Springer, 140, 152-158, 2011.
6. Dong, L, Yu, G, Ogeinbona, P and Li, W, An Efficient Iterative Algorithm for Image Thresholding, Pattern Recognition Letter, 29(9), 1311-1316, 2008.
7. Pal, N.R. and Pal, S, A Review on Image Segmentation Techniques. Pattern Recognition, 26, 1277-1294, 1993.
8. Sahoo, A.K.C., Soltam, S. and Wong, A.K.C, SURVEY: Survey of Threshold Techniques, Computer Vision Graphics and Image Processing, 41, 233-260, 1988.
9. Trier, O.D. and Jain, A.K, Goal-Directed Evaluation of Binarization Method, IEEE Transaction on Pattern. Analysis and Machine Intelligence. 17,1191-1201, 1995.
10. Trier, O.D. and Taxt, T, Evaluation of Binarization Method for Document Images, IEEE Transaction on Pattern. Analysis and Machine Intelligence. 17, 312-315, 1995.
11. Kaur J and Mahajan R, A review of Degraded Document Image Binarizaton Techniques, International Journal of Advanced Research in Computer and Communication Engineering, 3(5),6581-6586, 2014.
12. Milyaev S, Barinova O, Novikova T, Lempitsky V, and Kohli P, Image Binarization for End-to-End Text Understanding in Natural Images, Proceeding of ICDAR, 128 – 132, 2013.
13. Otsu, N, A Threshold Selection Method from Gray-level Histogram, IEEE Transaction on System, Man. and Cybernetics, 9(1), 62-66, 1979.
14. Kalavathi, P, Brain Tissue Segmentation in MR Brain Images using Otsu's Multiple Thresholding Technique, IEEE Xplore Digital Library, 638-642, 2013.
15. Memarsadeghi, N. and Netanyahu, N.S., LeMoigne, J., A Fast Implementation of the ISODATA Clustering Algorithm, International Journal of Computational Geometry and Applications, 7(1), 71–103 , 2006.