RESEARCH ARTICLE OPEN ACCESS

Image Segmentation Using Local Thresholding And Ycbcr Color Space

Amit Kumar Mandal*, Dilip Kumar Baruah**
*(Dept of Computer Science, Moridhal College, Moridhal, Dhemaji – 787057)
** (Dept of Computer Science, Moridhal College, Moridhal, Dhemaji – 787057)

ABSTRACT
In this experiment a system has been developed that uses threshold values to segment an image. An image is treated as a one dimensional array of pixel values. In this work, the segmentation is performed in YCbCr color space. In this experiment, the segmented image will have 2 different colors, which are black and white, and for this reason the segmentation is done using local thresholding value for Cb component of YCbCr. A mask is used to determine the neighbors of each pixel in the image. The mask also determines an operation to be applied to the neighborhood of every pixel in the image. Now the mask and the operations are used to determine the local threshold for every pixel in the image. For each pixel location the threshold will be different. This value is compared with the color value of the pixel. If the value of the pixel in this location is greater than or equal to the specified threshold for the pixel then it is labeled as 1. Otherwise if it is smaller, then it is labeled as 0.In this way we get an image with two color values.
Keyword: Image, segmentation, local thresholding, YCbCr.

I. INTRODUCTION

Different computer and image processing systems require an image segmentation-pre-processing algorithm as a first procedure. Segmentation refers to the process of partitioning a digital image into multiple segments (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. Each of the pixels in a region is similar with respect to some characteristic or computed property, such as color, intensity, or texture. The regions which are adjacent are significantly different with respect to the same characteristic(s).

Thresholding is one of the simplest and most commonly used techniques for image segmentation. It is useful in discriminating foreground from the background. Thresholding techniques can be categorized into two classes: global thresholding and local (adaptive) thresholding. In the global thresholding, a single threshold value is used in the whole image. In the local thresholding, a threshold value is assigned to each pixel to determine whether it belongs to the foreground or the background pixel using local information around the pixel.
Since the global thresholding is simple and easy to implement, it has been a popular technique in many years. It works well when the intensity distribution of objects and background pixels are sufficiently distinct. But in our real life we have images whose intensity distributions are not distinct. But local thresholding works well in this case since the threshold value at any point depends on the properties of neighborhood of that point.

When a color model is associated with a precise description of how the components are to be interpreted (viewing conditions, etc.), the resulting set of colors is called color space. There are various color spaces such as CIE, YUV, YCbCr, HSV etc. YCbCr is one of the popular color spaces in computing. It represents colors in terms of one luminance component (Y), and two chrominance components (Cb and Cr). YCbCr is used widely in video and image compression schemes such as MPEG and JPEG.

1.1 Local thresholding:

In this method a threshold is computed at every point (x,y) in the image based on one or more specified properties computed in a neighborhood of (x,y).Although this may seem like a laborious process, modern algorithms and hardware allow for fast neighborhood processing, especially for common functions such as logical and arithmetic operations.

Local thresholding can be done using standard deviation and mean of the pixels in a neighborhood of every point in an image. These two quantities are useful for determining the local thresholds because they are descriptors of local contrast and average intensity. Let σ_{xy} and m_{xy} denote the standard deviation and mean value of the set of pixels contained in a neighborhood, S_{xy}, centered at coordinates (x,y) in an image. The following are common forms of variable, local thresholds :

$$Txy = a\,\sigma_{xy} + b\,m_{xy}$$

where a and b are non negative constants, and

$$Txy = a\,\sigma_{xy} + b\,mG$$

where mG is the global image mean.

The Segmented image is computed as

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T_{xy} \\ 0 & \text{if } f(x,y) <= T_{xy} \end{cases}$$

Where f(x,y) is the input image. This equation is evaluated for all pixel locations in the image, and a different threshold is computed at each location (x,y) using the pixels in the neighborhood S_{xy}.

1.2 YCbCr

YCbCr is a color space used as part of the color image pipeline in video and digital photography systems. YCbCr is not an absolute color space. It is a way of encoding RGB information. The actual color displayed depends on the actual RGB primaries used to display the signal.

It represents color in terms of one luminance component (Y) and two chrominance components (Cb and Cr), where Cb is the chrominance-blue component and Cr is the chrominance –red component.

The YCbCr image can be converted to/from RGB image. To convert from RGB to YCbCr, one variant of this color space (according to ITU-R BT.709):

Y = 0.2126 *red+0.7152*green+0.0722*blue
Cb = 0.5389*(blue-Y)
Cr = 0.6350*(red-Y)

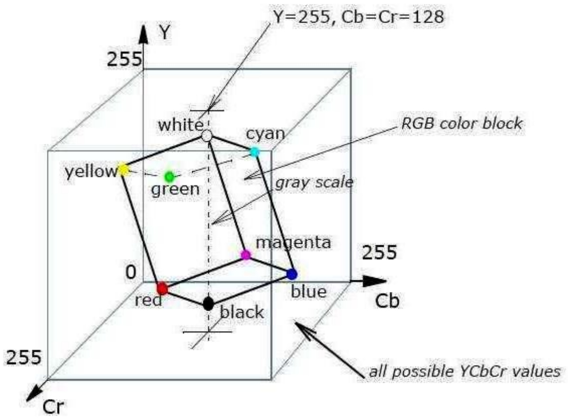

Fig1: RGB colors cube in the YCbCr space

1.2.1 Reasons for using YCbCr:

1. Human eye has different sensitivity to colour and brightness. Thus there is the need to transform RGB to YCbCr.
2. In contrast to RGB, the YCbCr color space is lumma-independent, resulting in better performance.
3. YCbCr is broadly utilized in video compression standards such as MPEG and JPEG.
4. Y can be stored with high resolution or transmitted at high bandwidth, and two chrominance components (C_B and C_R) that can be bandwidth-reduced, sub sampled, compressed or otherwise treated separately for improved system efficiency.

II. ALGORITHM

2.1 Introduction:

In this section we discuss about the algorithm that has been used for image segmentation. We have used the variable thresholding approach based on local image properties as the base which is also known as the local thresholding method to implement the system. Color value of the pixels has been used as the property to implement the algorithm.

2.2 Algorithm analysis:

The algorithm considers the image pixels as a one-dimensional array. The algorithm then computes the RGB value of each pixel in the image. The RGB values are then converted to YCbCr color space. These were done by using the following transformation:

Y = 0.2126 *red + 0.7152*green +0.0722*blue
Cb = 0.5389*(blue-Y)
Cr = 0.6350*(red-Y)

The value of Cb is used for further calculations. For the computation a 3x3 mask has been selected. It should be kept small. If a large mask is selected, the result will be poorer as it is more influenced by the illumination gradient. This mask slides all over the image and place the centre of the mask on the pixel for which we need to calculate the threshold. The algorithm then computes the mean and standard deviation of the pixels in the neighborhood of the specified pixel.

The general form for calculating the Mean and Standard Deviation value are as shown below:

Mean:

$$m = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)$$

Standard Deviation:

$$\sigma = \sqrt{\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}[f(x,y)-m]^2}$$

Where M = Number of rows of the image
N = Number of columns of the image.

The calculation of Standard Deviation and Mean is taken into account varying number of pixels. This is because for some of the pixels in the extreme boundary of the image, some portion of the mask will go outside the image region. So we need to take only those regions of the mask which are inside the image. The calculated standard deviation and mean are then used to calculate the threshold value for that pixel. The following expression is used for calculating the threshold:

$$T= a*\sigma_{xy}+b*m_{xy} ,$$

where a and b are non negative constants.

The Cb value of the pixel is then compared with the threshold value. If the Cb value is greater than the threshold the pixel is labeled as 1 .Otherwise if the Cb value is less than or equal to the threshold than the pixel is labeled as 0.

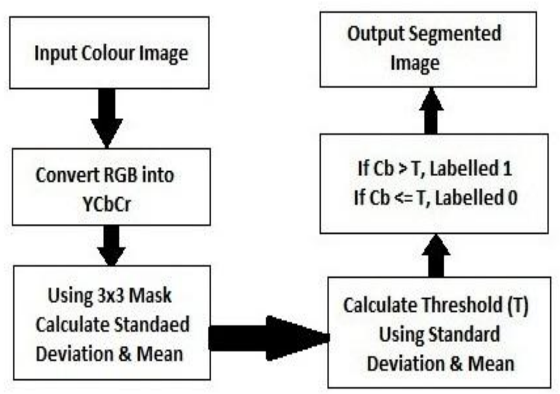

Fig2: Block diagram of the algorithm

III. EXPERIMENTAL RESULTS

The algorithm has been implemented using the Java 2.0 programming language in a machine having Intel® Pentium Dual-Core Processor and having the processor speed of 2.6 GHz and 0.99 GB of RAM. The machine operates using Ubuntu 11.04 operating system.

The system was tested with different images. We used two sets of images -- the image which has different objects in the image which were not distinct from the background and the images with sufficiently distinct objects from the background.

For our experiment we used only the JPG images. Fig: 3 shows the input image in which the objects are not sufficiently distinct from the background. Fig: 4 shows the output images for respective Fig: 3 images. Fig: 5 shows the input image in which the objects are sufficiently distinct from the background of the images. Fig: 6 shows the output images for respective Fig: 5 images.

From our experiments on different images, we observed that the system works well on both the cases -- when the objects in the image were indistinct and distinct from the background. We experimented on blurred images. But the performances were not as good as these results.

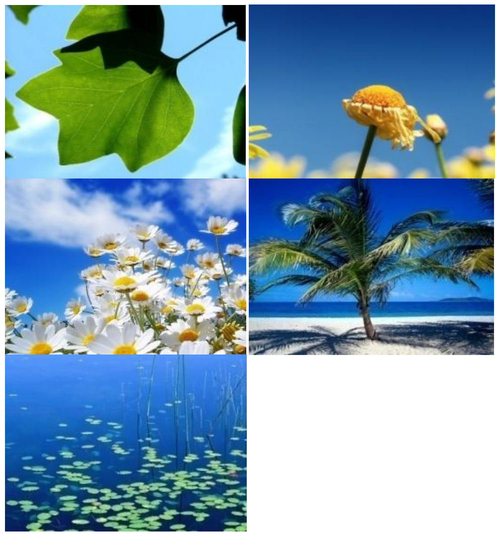

Fig 3: Input images with objects indistinct from background

Fig 4: Output for images with objects indistinct from background

Fig 5: Input images with objects distinct from background

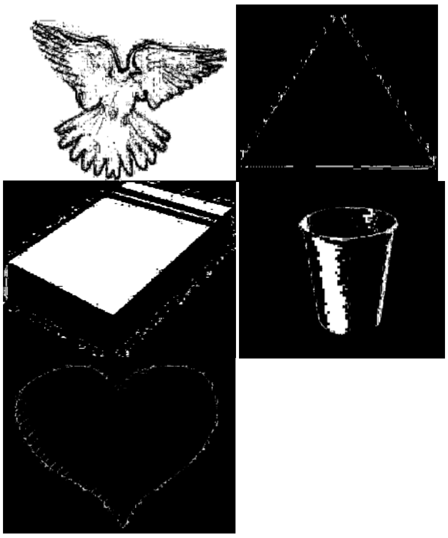

Fig 6: Output for images with objects distinct from background

IV. CONCLUSION AND FUTURE WORKS:

In this paper a method for image segmentation has been discussed which is based on local threshoding and YCbCr color space. Experimental results show that our method can produce good results for the process of color image segmentation. There is a scope for research on blurred images (motion and still) using local thresholding and YCbCr.

REFERENCES

Journal Paper:

[1] S. S. Al-amri, N.V. Kalyankar and S.D. Khamitkar, Image Segmentation by Using Threshod Techniques, *Journal of Computing*, Vol. 2, Issue 5, 2010, 83 – 86.

[2] P. Bhosale and A. Gokhale, Segmentation of Color Images Based on Different Segmentation Techniques, *International Journal of Electronics and Computer Science Engineerin, Vol. 2, No. 2,* 640 – 644.

[3] J. A. M. Saif , A. A. Md. Al-Kubati, A. S. Hazaa, and Md. Al-Moraish, Image Segmentation Using Edge Detection and Thresholding, *The 13th International Arab Conference on Information Technology,* 2012, 473 – 476.

[4] V. Power, A. Jahagirdar and S. Sirsikar, Skin Detection in YCbCr Color Space, *International Journal of Computer Applications*, 2011, 1 – 4.

[5] D. Cahi and K. N. Ngan, Face SegmentationUsing Skin-Color Map in Videophone Applications, *IEEE Transaction on Circuit and Systems for Video Technology*, *Vol. 9*, 1999, 551-564.

[6] G. E. Suji, Dr. Y. V. S. Lakshmi, and Dr. G. W. Jiji, MRI Brain Image Segmentation based on Thresholding, *International Journal of Advanced Computer Research, Vol. 3, No. 1*, 2013, 97 – 101.

[7] P. Thakare, A Study of Image Segmentation and Edge Detection Techniques, *International Journal on Computer Science and Engineering*, *Vol. 3, No. 2*, 2011, 899 – 904.

Books:

[1] Rafael C. Gonzalez, Richard E. Woods, *Digital Image Processing* (Pearson (2008