IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 6, No 1, November 2011
ISSN (Online): 1694-0814
www.IJCSI.org

Automatic Histogram Threshold with Fuzzy Measures using C-means

1. **K.SRINIVAS**

Research Scholor K.L. University Vaddeswaram,AP,India.

2. **Dr. V. SRIKANTH**

Dean-SED & HOD of IST K.L. University,Vaddeswaram,AP,India

Abstract:

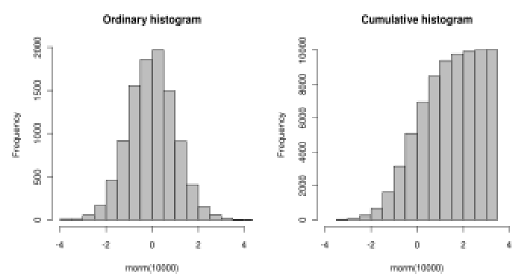
In this paper, an automatic histogram threshold approach based on a fuzziness measure is presented. This work is an improvement of an existing method. Using fuzzy logic concepts, the problems involved in finding the minimum of a criterion function are avoided. Similarity between gray levels is the key to find an optimal threshold. Two initial regions of gray levels, located at the boundaries of the histogram, are defined. Then, using an index of fuzziness, a similarity process is started to find the threshold point. A significant contrast between objects and background is assumed. Previous histogram equalization is used in small contrast images. Fuzzy c-means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters. This method is frequently used in pattern recognition. It is based on minimization of the objective function ! No prior knowledge of the image is required.
Keywords: fcm,threshold,histogram

INTRODUCTION

IMAGE segmentation plays an important role in computer vision and image processing applications. Segmentation of nontrivial images is one of the most difficult tasks in image processing.Segmentation accuracy determines the eventual success or failure of computerized analysis procedures. Segmentation of an image entails the division or separation of the image into regions of similar attribute. For a monochrome image, the most basic attribute for segmentation is image luminance amplitude Segmentation based on gray level histogram thresholding is a method to divide an image containing two regions of interest:object and background. In fact, applying this threshold to the whole image, pixels whose gray level is under this value are assigned to a region and the remainder to the other. Histograms of images with two distinct regions are formed by two peaks separated by a deep valle called bimodal histograms. In such cases, the threshold value must be located on the valley region. When the image histogram does not exhibit a clear separation, ordinary thresholding techniques might perform poorly. Fuzzy set theory provides a new tool to deal with multimodal histograms. It can incorporate human perception and linguistic concepts such as similarity, and has been successfully applied to image thresholding

An **image histogram** is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance. Image histograms are present on many modern digital cameras. Photographers can use them as an aid to show the distribution of tones captured, and whether image detail has been lost to blown-out highlights or blacked-out shadows. The horizontal axis of the graph represents the tonal variations, while the vertical axis represents the number of pixels in that particular tone. The left side of the horizontal axis represents the black and dark areas, the middle represents medium grey and the right hand side represents light and pure white areas. The vertical axis represents the size of the area that is captured in each one of these zones. Thus, the histogram for a very bright image with few dark areas and/or shadows will have most of its data points on the right side and center of the graph. Conversely, the histogram for a very dark image will have the majority of its data points on the left side and center of the graph.

Mathematical definition



An ordinary and a cumulative histogram of the same data. The data shown is a random sample of 10,000 points from a normal distribution with a mean of 0 and a standard deviation of 1. In a more general mathematical sense, a histogram is a function m_i that counts the number of observations that fall into each of the disjoint categories (known as *bins*), whereas the graph of a histogram is merely one way to represent a histogram. Thus, if we let n be the total number of observations and k be the total number of bins, the histogram m_i meets the following conditions:

$$n = \sum_{i=1}^{k} m_i.$$

Cumulative histogram

A cumulative histogram is a mapping that counts the cumulative number of observations in all of the bins up to the specified bin. That is, the cumulative histogram M_i of a histogram m_j is defined as:

$$M_i = \sum_{j=1}^{i} m_j.$$

Number of bins and width: There is no "best" number of bins, and different bin sizes can reveal different features of the data. Some theoreticians have attempted to determine an optimal number of bins, but these methods generally make strong assumptions about the shape of the distribution. Depending on the actual data distribution and the goals of the analysis, different bin widths may be appropriate, so experimentation is usually needed to determine an appropriate width. There are, however, various useful guidelines and rules of thus The number of bins *k* can be assigned directly or can be calculated from a suggested bin width *h* as:

$$k = \left\lceil \frac{\max x - \min x}{h} \right\rceil .$$

Implementation

Consider a discrete grayscale image {*x*} and let n_i be the number of occurrences of gray level *i*. The probability of an occurrence of a pixel of level *i* in the image is

$$p_x(i) = p(x=i) = \frac{n_i}{n}, \quad 0 \le i < L$$

L being the total number of gray levels in the image, *n* being the total number of pixels in the image, and $p_x(i)$ being in fact the image's histogram for pixel value i, normalized to [0,1].

Let us also define the cumulative distribution function corresponding to p_x as

$$cdf_x(i) = \sum_{j=0}^{i} p_x(j),$$

which is also the image's accumulated normalized histogram. We would like to create a transformation of the form $y = T(x)$ to produce a new image {*y*}, such that its CDF will be linearized across the value range, i.e.

$$cdf_y(i) = iK$$

for some constant *K*. The properties of the CDF allow us to perform such a transform (see Cumulative distribution function#Inverse it is defined as

$$y = T(x) = cdf_x(x)$$

Notice that the T maps the levels into the range [0,1]. In order to map the values back into their original range, the following simple transformation needs to be applied on the result:

$$y' = y \cdot (\max\{x\} - \min\{x\}) + \min\{x\}$$

Histogram equalization of color images : The above describes histogram equalization on a grayscale image. However it can also be used on color images by applying the same method separately to the Red, Green and Blue components of the RGB color values of the image. However, applying the same method on the Red, Green, and Blue components of an RGB image may yield dramatic changes in the image's color balance since the relative distributions of the color channels change as a result of applying the algorithm. However, if the image is first converted to another color space, Lab color space, or HSL/HSV color space in particular, then the algorithm can be applied to the luminance or value channel without resulting in changes to the hue and saturation of the image. There are several histogram equalization methods in 3D space. Trahanias and Venetsanopoulos applied histogram equalization in 3D color spaceHowever, it results in “whitening” where the probability of bright pixels are higher than that of dark ones. Han et al. proposed to use a cdf defined by the iso-luminance plane, which results in uniform gray distribution

EXISTING METHOD:

This work is an improvement of an existing method based on a fuzziness measure to find the threshold value in a gray image histogram . The method incorporates fuzzy concepts that are more able to deal with object edges and ambiguity and avoids the problems involved in finding the minimum of a function. However, it has some limitations concerning the initialization of the seed subsets. To achieve an automatic process these limitations must be overcome. In order to implement the thresholding algorithm on a basis of the concept of similarity between gray levels, Tobias and Seara made the assumptions that there exists a significant contrast between the objects and background and that the gray level is the universe of discourse, a 1-D set, denoted by . The purpose is to split the image histogram in two crisp subsets, object subset and background subset , using the measure of fuzziness previously defined. The initial fuzzy subsets, denoted by and , are associated with initial histogram intervals located at the beginning and the end regions of the histogram. The gray levels in each of these initial intervals have the intuitive property of belonging with certainty tothe final subsets object or background. For dark objects and , for light objects and . These initial fuzzy subsets, and , are modeled by the and membership functions, respectively. The parameters of the and functions are variable to adjust its shape as a function of the set of elements .These subsets are a seed for starting the similarity measure process. A fuzzy region placed between these initial intervals is defined as depicted in Fig. 2. Then, to obtain the segmented version of the gray level image, we have to classify each gray level of the fuzzy region as being object or background. The classification procedure is done by adding to each of the seed subsets a gray level picked from the fuzzy region. Then, by measuring the index of fuzziness of the subsets and , the gray level is assigned to the subset with lower index of fuzziness (maximum similarity). Applying this procedure for all gray levels of the fuzzy region, we can classify them into object or background subsets. Since the method is based on measures of index of fuzziness, these measures need to be normalized by first computing the index of fuzziness of the seed subsets and calculating a normalization factor according to where and are the IF’s of the subsets and , respectively.This normalization operation ensures that both initial subsets have identical index of fuzziness at the beginning of the process. It is a necessary condition since the method is based in the calculation of similarity between gray levels.illustrate how the normalization works. For dark objects, the method can be described as follows.

1. Compute the normalization factor .
2. For all gray levels in the fuzzy region compute and .
3. If is lower than , then is included in set , otherwise is included in set .

For light objects the method performs similarly except for the set inclusion in step 3. In this case, if is lower than , then is included in set , otherwise is included in set .

Threshold: It is the simplest method of image segmentation. From a grayscale image, thresholding can be used to create binary images Method. During the thresholding process, individual pixels in an image are marked as "object" pixels if their value is greater than some threshold value (assuming an object to be brighter than the background) and as "background" pixels otherwise. This convention is known as threshold above. Variants include threshold below, which is opposite of threshold above; threshold inside, where a pixel is labeled "object" if its value is between two thresholds; and threshold outside, which is the opposite of threshold inside .Typically, an object pixel is given a value of "1" while a background pixel is given a value of "0." Finally, a binary image is created by coloring each pixel white or black, depending on a pixel's labels.

Threshold selection: The key parameter in the thresholding process is the choice of the threshold value (or values, as mentioned earlier). Several different methods for choosing a threshold exist; users can manually choose a threshold value, or a thresholding algorithm can compute a value automatically, which is known as automatic thresholding A simple method would be to choose the mean or median value, the rationale being that if the object pixels are brighter than the background, they should also be brighter than the average. In a noiseless image with uniform background and object values, the mean or median will work well as the threshold, however, this will generally not be the case. A more sophisticated approach might be to create a histogram of the image pixel intensities and use the valley point as the threshold. The histogram approach assumes that there is some average value for the background and object pixels, but that the actual pixel values have some variation around these average values. However, this may be computationally expensive, and image histograms may not have clearly defined valley points, often making the selection of an accurate threshold difficult. One method that is relatively simple, does not require much specific knowledge of the image, and is robust against image noise, is the following iterative method:

1. An initial threshold (T) is chosen, this can be done randomly or according to any other method desired.
2. The image is segmented into object and background pixels as described above, creating two sets:
    1. $G_1 = \{f(m,n):f(m,n)>T\}$ (object pixels)
    2. $G_2 = \{f(m,n):f(m,n)\leq T\}$ (background pixels) (note, $f(m,n)$ is the value of the pixel located in the m^{th} column, n^{th} row)
3. The average of each set is computed.
    1. m_1 = average value of G_1
    2. m_2 = average value of G_2
4. A new threshold is created that is the average of m_1 and m_2
    1. $T' = (m_1 + m_2)/2$
5. Go back to step two, now using the new threshold computed in step four, keep repeating until the new threshold matches the one before it (i.e. until convergence has been reached).

This iterative algorithm is a special one-dimensional case of the k-means clustering algorithm, which has been proven to converge at a local minimum—meaning that a different initial threshold may give a different final result.

Method selects the algorithm to be applied :

The Ignore black and Ignore white options set the image histogram bins for [0] and [255] greylevels to 0 respectively. This may be useful if the digitised image has under- or over- exposed pixels. White object on black background sets to white the pixels with values above the threshold value (otherwise, it sets to white the values less or equal to the threshold). Set Threshold instead of Threshold (single images) sets the thresholding LUT, without changing the pixel data. This works only for single images. It you are processing a stack, two additional options are available: Stack can be used to process all the slices (the threshold of each slice will be computed separately). If this option is left unchecked, only the current slice will be processed. Use stack histogram first computes the histogram of the whole stack, then computes the threshold based on that histogram and finally binarises all the slices with that single value. Selecting this option also selects the Stack option above automatically.

Important notes:

1. This plugin is accessed through the Image>Auto Threshold menu entry, however the thresholding methods were also partially implemented in ImageJ's thresholder applet accessible through the Image>Adjust>Threshold... menu entry. While the Auto Threshold plugin can use or ignore the extremes of the image histogram (Ignore black, Ignore white) the applet cannot: the 'default' method ignores the histogram extremes but the others methods do not. This means that applying the two commands to the same image can produce apparently different results. In essence, the Auto Threshold plugin, with the correct settings, can reproduce the results of the applet, but not the way round.

2. From version 1.12 the plugin supports thresholding of 16-bit images. Since the Auto Threshold plugin processes the full greyscale space, it can be slow when dealing with 16-bit images. Note that the ImageJ thresholder applet also processes 16-bit images, but in reality ImageJ first computes a histogram with 256 bins. Therefore, there might be differences in the results obtained on 16-bit images when using the applet and the true 16-bit results obtained with this plugin. Note that for speeding up, the histogram is bracketed to include only the range of bins that contain data (and avoid processing empty histogram bins at both extremes).

3. The result of 16 bit images and stacks (when processing all slices) is an 8 bit container showing the result in white [255] to comply with the concept of "binary image" (i.e. 8 bits with 0 and 255 values). However, for stacks where only 1 slice is thresholded, the result is still a 16 bit container with the thresholded phase shown as white [65535]. This is to keep the data untouched in the remaining slices. The "Try all" option retains the 16 bit format to still show the images with methods that might fail to obtain a threshold. Images and stacks that are impossible to threshold remain unchanged.

4. The same image in 8 and 16 bits (without scaling) returns the same threshold value, however Li's method originally would return different values when the image data was offset (e.g. when adding a fixed value to all pixels). The current implementation avoids this offset-dependent problem.

5. The same image scaled by a fixed value (e.g. when multiplying all pixels by a fixed value) returns a similar threshold result (within 2 greyscale levels of the original unscaled image) for all methods except Huang, Li and Triangle due to the way these algorithms work.

Proposed System: : The concept presented above sounds attractive but has some limitations concerning the initialization of the seed subsets. In these subsets should contain enough information about the regions and its boundaries are defined manually. The proposedmethod in this paper aims to overcome some of the limitations of the existing method. In fact, the initial subsets are defined automatically and they are large enough to accommodate a minimum number of pixels defined at the beginning of the process. This minimum depends on the image histogram shape and it is a function of the number of pixels in the gray level intervals and . It is calculated as follows:

Fuzzy c-means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters. This method is frequently used in pattern recognition. It is based on minimization of the objective function !

1. Initialize U=[u ij] matrix, U (0)1
2. At k-step: calculate the centers vectors C (k) =[c j] with U (k)

The Algorithm (Contd…)

3. Update U (k) , U (k+1)

4. If $\| U(k+1) - U(k) \| < \pounds$ then STOP; otherwise return to step 2.

Where,

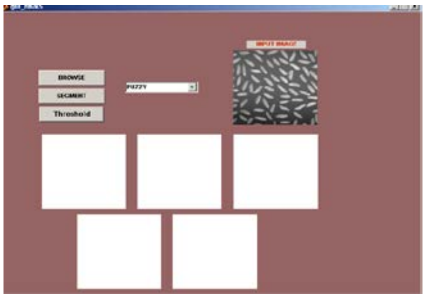
U = Membership Matrix £ = Termination Criteria

C = Centroids

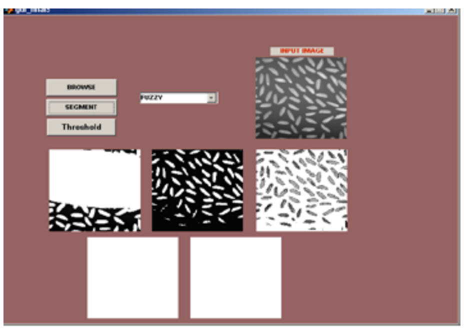
X = Pixel Intensity

Results:

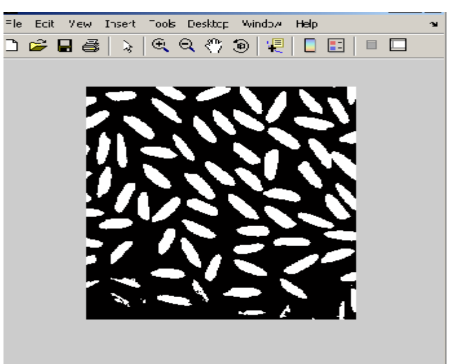
Input Image



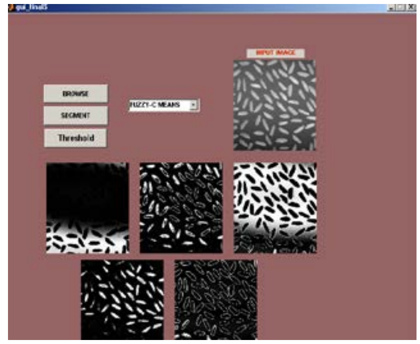
Segmentation with fuzzy



Thresholding



Segmentation with fuzzy c means



Conclusion & Future work:

In this paper

In this paper, an automatic histogram threshold approach based on index of fuzziness measure is presented. This work overcome some limitations of an existing method concerning the definition of the initial seed intervals. Method convergence depends on the correct initialization of these initial intervals. After calculating the initial seeds a similarity process is started to find the threshold point. This property of similarity is obtained calculating an index of fuzziness. To measure the performance of the proposed method the misclassification error parameter is calculated. For performance evaluation purposes, results are compared with two well established methods: the Otsu's technique and the Fuzzy C-means clustering algorithm. After results analysis we can conclude that the proposed approach presents a higher performance for a large number of tested images. Standard Fuzzy C-Mean is not suitable for the lip and skin region . The resulting regions are not spatially continuous, due to the fact that only gray level uniformity is checked. FW,In future we will implement the nuero fuzzy based histogram threshold approach. That will be over come the distortion level occur based on fuzzy.

References

[1] W. K. Pratt, *Digital Image Processing*, third ed. New York: Wiley, 2001.

[2] A. S. Pednekar and I. A. Kakadiaris, "Image segmentation based on fuzzy connectedness using dynamic weights," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1555–1562, Jun. 2006.

[3] S. Sahaphong and N. Hiransakolwong, "Unsupervised image segmentation using automated fuzzy c-means," in *Proc. IEEE Int. Conf. Computer and Information Technology*, Oct. 2007, pp. 690–694.

[4] O. J. Tobias, R. Seara, and F. A. P. Soares, "Automatic image segmentation using fuzzy sets," in *Proc. 38th Midwest Symp. Circuits and Systems*, 1996, vol. 2, pp. 921–924.

[5] O. J. Tobias and R. Seara, "Image segmentation by histogram thresholding using fuzzy sets," *IEEE Trans. Image Process.*, vol. 11, 2002.

[6] C. V. Jawahar, P. K. Biswas, and A. K. Ray, "Investigations on fuzzy thresholding based on fuzzy clustering," *Pattern Recognit.*, vol. 30, no. 10, pp. 1605–1613, 1997.

[7] K. S. Chuang, H. L. Tzeng, S. Chen, J. Wu, and T. J. Chen, "Fuzzy c-means clustering with spatial information for image segmentation," *Comput. Med. Imag. Graph.*, vol. 30, no. 1, pp. 9–15, 2006.

[8] L. K. Huang and M. J. J. Wang, "Image thresholding by minimizing the measures of fuzziness," *Pattern Recognit.*, vol. 28, no. 1, pp. 41–51, 1995

[9] H. R. Tizhoosh, "Image thresholding using type II fuzzy sets," *Pattern*

Recognit., vol 38, pp. 2363–2372, 2005.
[10] A. Rosenfeld and P. de la Torre, "Histogram concavity analysis as an aid in threshold selection," *SMC*, vol. 13, no. 3, pp. 231–235, Mar. 1983.
[11] J. S. Wezka and A. Rosenfeld, "Histogram modification for threshold selection," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, pp. 38–52, 1979.
[12] T. Ridler and S. Calvard, "Picture thresholding using an iterative selection method," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 630–632, Aug. 1978.
[13] N. Otsu, "A threshold selection method from gray level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, pp. 62–66, 1979.
[14] J. Kittler and J. Illingworth, "Minimum error thresholding," *Pattern Recognit.*, vol. 19, no. 1, 1986.
[15] J. N. Kapur, P. K. Sahoo, and A. K. C.Wong, "A new method for graylevel picture thresholding using the entropy of the histogram," *Graph. Models Image Process.*, vol. 29, pp. 273–285, 1985.
[16] T. Pun, "A new method for gray-level picture thresholding using the entropy of the histogram," *Signal Process.*, vol. 2, no. 3, pp. 223–237, 1980.
[17] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *J. Electron. Imag.*, vol. 13, no. 1, pp. 146–165, Jan. 2004.
[18] C. Murthy and S. Pal, "Fuzzy thresholding: Mathematical framework, bound functions and weighted moving average technique," *Pattern Recognit. Lett.*, vol. 11, pp. 197–206, 1990.
[19] N. R. Pal and J. C. Bezdek, "Measuring fuzzy uncertainty," *IEEE Trans. Fuzzy Syst.*, vol. 2, 1994.
[20] A. Kaufmann, *Introduction to the Theory of Fuzzy Subsets*. New York: Academic, 1975, vol. I.
[21] R. C. Gonzalez and R. E.Woods, *Digital Image Processing*. Reading, MA: Addison-Wesley, 1993.

Babilography:

Author 1: K. SRINIVAS obtained his M.Tech. from Nagarjuna University and pursuing his PhD from KL University under the guidance of Dr. V. Srikanth. His interested research area are image processing.

Author 2: Dr. V.Srikanth obtained his B.Tech and M.E degrees from Madras University and PhD from Nagarjuna university. Presently working as Dean and HOD of IST, K.L. University,Vaddeswaram.AP.