

A Triclass Image Segmentation using Adaptive K-means Clustering and Otsu's Method

Parvathy U S, Stephe Sara Philipose
P.G Scholar, Dept of CSE, BMCE, Kollam, Kerala, India
Parvathyus125@gmail.com

Abstract— Image segmentation is used to cluster an image based on features or regions of an image. So each cluster represents each segment. There are different segmentation algorithms, in which thresholding segmentation has an important role. Thresholding is used to obtain binary of an image, in which one state represent the foreground and the complementary represent the background. Among the various thresholding methods otsu method is the best with respect to the time and accuracy. Here we present the modification of otsu method by combining it with adaptive k-means clustering algorithm. Based on the otsu's threshold and two mean values, the algorithm separate it into three regions such as background, foreground and the TBD (to-be-determined) region. The further processing carried out in this TBD region, which is between the two mean values. In this region we apply the adaptive k-means algorithm. The process stops until no cluster left with the image. The new method can achieve good performance in some challenging cases, such as identifying weak objects and revealing fine structures of complex objects while the added computational cost and time is minimal.

Keywords — Binarization, Segmentation, Threshold, Otsu's Method, Triclass Segmentation, Adaptive k-means Clustering.

INTRODUCTION

Image segmentation is useful in any context where an image (such as a photograph) is to be in any way analyzed automatically by computer [5]. image processing and computer vision applications usually require binary images (i.e. black and white) as an introductory step in order to do further processing. The simplest way to use image binarization is to choose a threshold value, and classify all pixels with values above this threshold as white, and all other pixels as black. By choosing a particular intensity value as threshold, the thresholding segmentation can be broadly classified into global and local thresholding. In the global thresholding, partition the image histogram using a single global threshold value. In the local thresholding, divide an image into sub images and threshold these individually. The global thresholding method is simple and time efficient. So our method is coming under the category of global thresholding.

The otsu method is one of the best global binarization technique, and this method is widely used in pattern recognition, document binarization and computer vision [2]. It is trying to find one threshold value for the whole image. The aim of the method is to find the threshold value where the sum of foreground and background spreads is at its minimum. However, since Otsu's threshold is biased towards the class with a large variance, it tends to miss weak objects or fine details in images. For example in biomedical images, nuclei and axons may be imaged with very different intensities due to uneven staining or imperfect lightening conditions. At that situations, algorithm like otsu's method raising difficulty to segment them successfully. It works well with clearly scanned images, but it performs unsatisfactorily for those poor quality images that have low contrast and non-uniform illumination [4]. A quad-tree approach was developed to segment images by combining a centroid clustering and boundary estimation methods but the approach only works under the assumption that the histogram consists of Gaussian distributions only[4]. In [8] the main image segmentation algorithm has been reviewed and gives some valuable characteristics of image segmentation algorithm. The authors have classified Otsu algorithm as thresholding region based segmentation algorithm. Also the complexity rate of Otsu thresholding algorithm is very high and processing rate is very slow. By concluding here, in order to improve the performance of the Otsu algorithm, combine it with other algorithms [9]. In this paper [10], the Otsu method works on global thresholding while the K means method work on the local thresholding. Both methods produce good segmentation result but K means give better results comparatively to Otsu. Otsu method takes comparatively more time and increases the complexity of the algorithm. In this technique, clustering is based on the identification of k elements in the data set that can be used to create an initial representation of clusters. even though the method seems to be straight forward, it suffers from the fact that it may not be easy to clearly identify the initial k elements [3]. In triclass method [4], it is comparatively better performance than all other methods, but fine parts of weak object is missing and initial threshold selection is also difficult.

In this method performs better than the iterative method. It is based on otsu method, but it perform better than the standard application. Initially, Segment the image using otsu's thresholding. Then we get a threshold value from this method. Then calculate two mean values with this threshold. So divide the image into three classes, using this mean. Then separate the image by less than first mean as background, greater than the second mean as the foreground and in-between these two mean values is represented as the TBD (to-be-determined) region. Then apply the adaptive k-means clustering method in the to-be-determined region. Combine the clusters with the foreground by checking the white pixel. This fusion continues until there is no cluster left.

The paper is organized as in the next section includes the related methods of the proposed methods. Then the next includes the proposed work and the flowchart of that. Ultimately this includes the conclusion.

RESEARCH METHODOLOGY

Here there are two methods are used, one is otsu's method and other one is adaptive k-means clustering.

- **Otsu's method**

Otsu's method searches the histogram of an image to find a threshold that binaries the image into two classes, the background with a mean of μ_0 and the foreground with a mean of μ_1 , without loss of generality, here we assume that the foreground is brighter than the background, i.e., $\mu_1 > \mu_0$. The calculation of threshold T is as follows

$$T = \operatorname{argmin}_T \sigma_{\omega}^2(T) \quad (1)$$

Where

$$\sigma_{\omega}^2(T) = q_0(T)\sigma_0^2(T) + q_1(T)\sigma_1^2(T) \quad (2)$$

Where the subscript 0 and 1 denote the two classes, background and foreground, respectively, and q_i and σ_i , $i = [0, 1]$ are the estimated class probabilities and class variances, respectively. These quantities are calculated as

$$q_0 = \sum_{i=1}^T P(i) \quad (3)$$

$$q_1 = \sum_{i=T+1}^k P(i) \quad (4)$$

And the individual class variance are given as

$$\sigma_0^2(T) = \sum_{i=1}^T [i - \mu_0(T)]^2 \frac{P(i)}{q_0(T)} \quad (5)$$

$$\sigma_1^2(T) = \sum_{i=T+1}^k [i - \mu_1(T)]^2 \frac{P(i)}{q_1(T)} \quad (6)$$

Where we assume that the pixel values of the images are from 0 to k. So from the above equations we can see that T is function of the pixel values of both the foreground and the background. If the signal intensity changes, it may affect T in such a way that the segmentation result may become less optimal [7].

- **Adaptive k-means clustering**

The adaptive K-means (AKM) clustering algorithm starts with the selection of K elements from the input data set. The K elements form the seeds of clusters and are randomly selected. This function is also used to compute distance between two elements. An important consideration for this function is that it should be able to account for the distance based on properties that have been normalized so that the distance is not dominated by one property or some property is not ignored in the computation of distance. In most cases, the Euclidean distance may be sufficient. For example, in the case of spectral data given by n-dimensions, the distance between two data elements

$$E_1 = \{E_{11}, E_{12}, \dots, E_{1n}\} \quad (7)$$

And

$$E_2 = \{E_{21}, E_{22}, \dots, E_{2n}\} \quad (8)$$

$$\text{Then } \sqrt{(E_{11} - E_{12})^2 + (E_{12} - E_{22})^2 + \dots + (E_{1n} - E_{2n})^2} \quad (9)$$

It should be pointed out that for performance reasons, the square root function may be dropped. In other cases, we may have to modify the distance function. Such cases can be exemplified by data where one dimension is scaled different compared to other dimensions, or where properties may be required to have different weights during comparison. With the distance function, the algorithm proceeds as follows: Compute the distance of each cluster from every other cluster. This distance is stored in a 2D array as a

triangular matrix. We also note down the minimum distance d_{min} between any two clusters C_{m1} and C_{m2} as well as the identification of these two closest clusters. For each unclustered element E_i , compute the distance of E_i from each cluster. For assignment of this element to a cluster, there can be three cases as follows:

1. If the distance of the element from a cluster is 0, assign the element to that cluster, and start working with the next element.
2. If the distance of the element from a cluster is less than the distance d_{min} , assign this element to its closest cluster. As a result of this assignment, the cluster representation, or centroid, may change. The centroid is recomputed as an average of properties of all elements in the cluster. In addition, we recomputed the distance of the affected cluster from every other cluster, as well as the minimum distance between any two clusters and the two clusters that are closest to each other.
3. If the distance of the element from a cluster is less than the distance d_{min} , assign this element to its closest cluster. As a result of this assignment, the cluster representation, or centroid, may change. The centroid is recomputed as an average of properties of all elements in the cluster. In addition, we recomputed the distance of the affected cluster from every other cluster, as well as the minimum distance between any two clusters and the two clusters that are closest to each other.

The above three steps are repeated until all the elements have been clustered. There is a possibility that the algorithm identifies a number of singletons or single-element clusters, if the distance of some elements is large from other elements. These elements are known as outliers and can be accounted for by looking for clusters with an extremely small number of elements and removing those elements from clustering consideration, or handled as exceptions [3].

PROPOSED METHOD

From different literatures it is found that the otsu threshold segmentation is found to be accurate, the idea of dividing an image's histogram iteratively into 3 classes. The idea of dividing an image's histogram iteratively into three classes is illustrated at the bottom of Fig. 1.

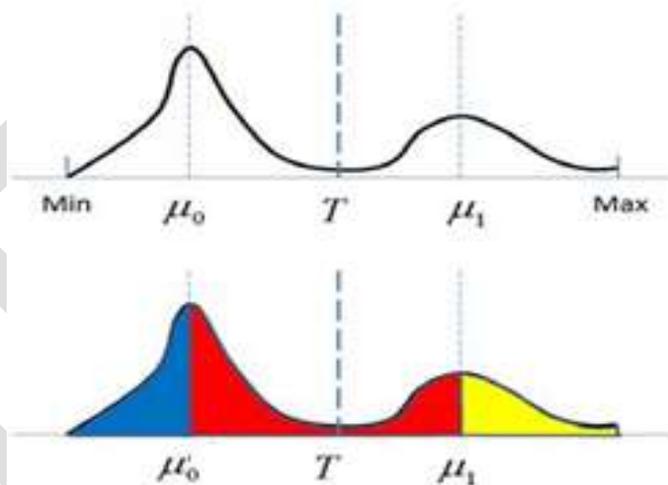


Fig.1. Histogram divided into three classes using mean

In top of Fig.1 shows that the Otsu's method binarizes an image to two classes based on threshold T . Bottom shows that our method classify the histogram into three classes, namely the foreground region with pixel values greater than μ_1 (shown in yellow), the background region with pixel values less than μ_0 in blue, and the third region, called TBD, in red.

For an image u , at the first iteration, Otsu's method is applied to find a threshold T , and denote the means of the two classes separated by T as μ_0 and μ_1 for the background and foreground, respectively. Then we classify regions whose pixel values are greater than μ_1 as foreground F and regions whose pixel values are less than μ_0 as background B . For the remaining pixels $u(x, y)$ such that $\mu_0 \leq u(x, y) \leq \mu_1$ we denote them as the TBD class Ω . So our process assumes that the pixels that are greater than the mean of the "tentatively" determined foreground are the true foreground. Similarly, pixels with values less than μ_0 are for certain the background. But the pixels in the TBD class, which are the ones that typically cause misclassifications in the standard Otsu's method, are not decided at once and will be further processed [4].

The second procedure is start with the TBD region. The adaptive k-means clustering is applied in this region. The AKM estimates the correct number of clusters and obtains the initial centers by the segmentation of the norm histogram in the linear normed space consisting of the data set, and then performs the local improvement heuristic algorithm for K-means clustering in order to avoid

the local optima. So we obtain clusters according to the images [6]. Take the first cluster and the foreground, which is greater than the μ_1 . Fuse this two by comparing the amount of white pixels, and take the output and fuse it with the second cluster, this continue until there is no cluster remaining. And then combine with the background. These steps are described in the flowchart shown in the Fig.2. This method can detect the fine details of weak objects clearly, because the TBD region has the confusion about background and foreground, in that region we applied the clustering technique, so we will get the clear picture of background and foreground. And also this method is parameter free and cost effective method.

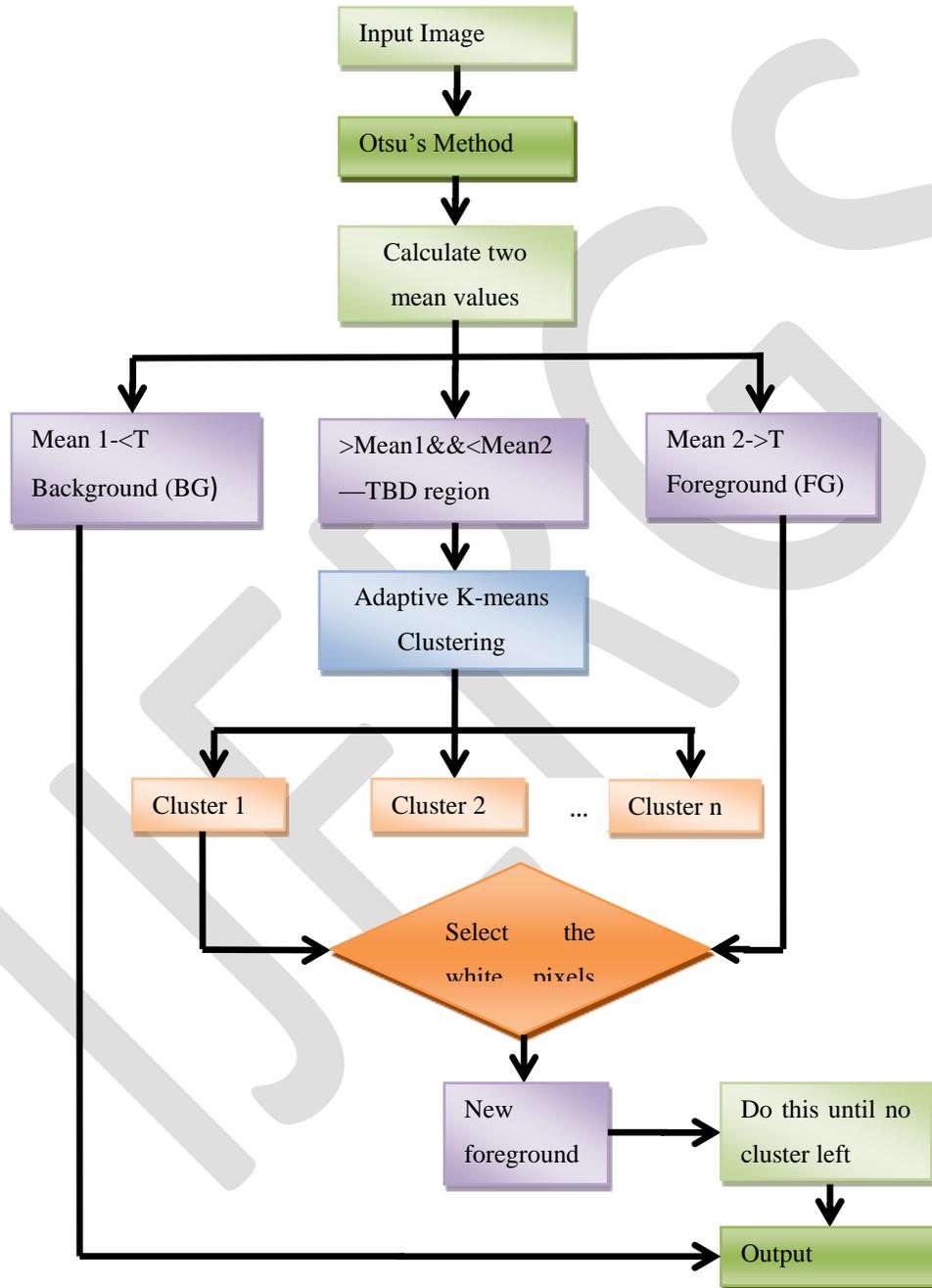


Fig.2. Flowchart of the proposed system

CONCLUSION

The otsu method is to divide the image into foreground and background, but it fails in some times to detect the weak objects clearly. Otsu can be used as a preprocessing technique in document binarization and pattern recognition.

So in this paper represent an enhancement of otsu's method. Instead of dividing in to classes, here represent 3 classes. One is foreground one is background and a newly determined region. Apply adaptive cluster in this region. Because this algorithm is a fully automatic way to cluster an input color or gray image using k-means principle, and here you do not need to specify number of clusters or any initial seed value to start iteration. This algorithm automatically finds number of cluster and cluster center iteratively. It is very fast implementation of clustering without knowing number of clusters. Then fuse the clusters with the foreground region by comparing the intensity. Finally combines with the background.

This method is parameter free and there is no complexity coding required. Also it perfectly determines the weak objects with minimal time and cost.

REFERENCES:

- [1] M. Huang, W. Yu, and D. Zhu, "An improved image segmentation algorithm based on the Otsu method," in Proc. 13th ACIS Int. Conf. Softw. Eng., Artif. Intell., Netw. Parallel Distrib. Comput., Aug. 2012, pp. 135–139
- [2] Ann Josy John, Merlin Mary James, "A New Tri Class Otsu Segmentation With K-means clustering in brain image segmentation," International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Volume 4 issue 3, march 2015
- [3] Sajiv K. Bhatia, "Adaptive k-means clustering"
- [4] Hongmin Cai, Zhong Yang, Xinhua Cao, Weiming Xia, and Xiaoyin Xu, "A New Iterative Triclass Thresholding Technique in Image Segmentation," IEEE Transactions on Image Processing, VOL. 23, NO. 3, March 2014.
- [5] Matthew Kerwin, "Comparison of Traditional Image Segmentation Techniques and Geostatistical Threshold"
- [6] [Chen, Hailin](#); [Wu, Xiuqing](#); [Hu, Junhua](#), "Adaptive K-means clustering algorithm"
- [7] Nobuyuki Otsu, "A Threshold Selection Method from Gray-Level Histograms", IEEE Transactions on Systems, Man, and Cybernetics, VOL. SMC-9, NO. 1, JANUARY 1979.
- [8] W. X. Kang, Q. Q. Yang, R. R. Liang, "The Comparative Research on Image Segmentation Algorithms", IEEE Conference on ETCS, pp. 703-707, 2009.
- [9] Miss Hetal J. Vala, Prof. Astha Baxi, "A Review on Otsu Image Segmentation Algorithm", ISSN: 2278 – 1323 International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 2, February 2013 387.
- [10] L. Dongju and Y. Jian, "Otsu method and k-means," in Hybrid Intelligent Systems, 2009. HIS '09. Ninth International Conference on, vol. 1, 2009, pp. 344–349.
- [11] Y. Zhang and L. Wu, "Fast document image binarization based on an improved adaptive Otsu's method and destination word accumulation," J. Comput. Inf. Syst., vol. 7, no. 6, pp. 1886–1892, 2011.
- [12] M. Cheriet, J. N. Said, and C. Y. Suen, "A recursive threshold technique for image segmentation," IEEE Trans. Image Process., vol. 7, no. 6, pp. 918–921, Jun. 1998