Efficient Background Subtraction Using Fuzzy C Means

Poornima S
Final Year M.E CSE
Anna University of Technology Thiruchirappalli
Thiruchirappalli, Tamilnadu, India
E-mail: poornimaselvaraj@gmail.com

Suganya Devi K
Assistant Professor/CSE
Anna University of Technology Thiruchirappalli
Banruti Campus, Banruti.
E-mail: its.siva04.aut@gmail.com

Abstract - This paper deals with an efficient background subtraction of image/frames of video by improving the execution speed, accuracy and reduce the usage of memory. Three important techniques are applied to improve the efficiency: superpixel extraction, canny edge detection and fuzzy c means. On applying the above three methods sequentially, the background of image/video can be segmented from foreground object accurately. The first method reduces the processing data more than 75%. Canny edge detection is an optimized method to detect edges. Fuzzy c means works well and good to segment the overlapped objects in an image/video.

Keywords: *superpixel, subpixel, canny edge detection, feature extraction, gradient, segmentation and fuzzy c means.*

I. INTRODUCTION

Background subtraction is the process of identifying moving objects from the portion of a video frame that differs significantly from a background scene [4]. The most commonly faced problem in segmenting the foreground object from background scene of a frame and identifying the motion changes of every frame by comparing it with previous frame is the foreground object may move continuously when background scene may be static whereas in some video both the foreground and background objects may move.

This nature of a video leads to produce noise, undetected edges, loss of smoothness, improper segmentation of foreground object especially in overlapped objects and maintaining robustness due to changes in illumination is difficult in the implementation of background subtraction algorithms[5]. Also some more additional work to be done for object tracking, motion analysis and behavior analysis for segmenting the foreground objects. The main aim to segment the foreground object from background scene is, to reduce the amount of data to be processed during comparison of frames. Thus an efficient background subtraction algorithm must have good foreground detection rate, processing rate and adapt quickly to the changes occurring at the background. This paper gives a solution for the above mentioned problems by sequentially applying the following three methods thereby improving the efficiency: (1) extraction of super pixel to reduce the storage capacity, (2) canny edge detection algorithm to detect the edges of objects in the video, and (3) finally fuzzy c means algorithm to segment the foreground object from background scene accurately. Implementation of the above mentioned techniques will be very much useful for many real time applications such as medical diagnosis, human-computer interaction, video surveillance and traffic monitoring.

II. SUPERPIXEL EXTRACTION

Super pixel is a polygonal part of a digital image/frame which is larger than normal pixel that is rendered in the similar properties such as color, brightness, texture, vector etc[1]. Super pixels are group of pixels with approximately same pixel values. The main goal of extracting the super pixels from an image/frame is to reduce the size of data that undergoes for background subtraction. Other than this the advantages of using super pixels are computational efficiency, reduction in number of primitives and hypothesis, and less error rate [2] [3]. The important point to be noted during super pixel extraction is pixel collisions must not occur such that none of the pixel must belong to more than one super pixel. Consider k is the total number of super pixels to be extracted from an image/frame, the total number of pixels in the image/frame is n, the pixel value of i^{th} pixel is represented by P_i. Fix a minimum threshold T_1=0 and a maximum threshold T_2=$\pm$1 for each super pixel. The pixel value of first pixel in the super pixel is taken as the initial value for that super pixel and the difference between each pixel values are compared with the first pixel value. The equation to determine the difference is,n

$$P(K) = \sum_{i=0} P_i - P_1$$

If the difference of pixel values lies between the threshold T_1(zero) to T_2($\pm$one) then the pixel is considered to be in the same super pixel else the pixel belongs to new super pixel.

$$P_i = \begin{cases} \text{if } T_1 <= P_i <= T_2 \text{ then current super pixel} \\ \text{Else neighboring super pixel} \end{cases}$$

Every pixel in the image/frame is undergone to the above mentioned analysis and hence super pixels are extracted from the image/frame. The direction of the pixel which does not lies between the threshold is noted down with respect to first pixel to form a new super pixel in that direction. For example, a pixel below the first pixel is out of the threshold value hence a new super pixel is created in the south of current super pixel and those pixel coordinates that is greater than this pixel alone are taken for comparison. This idea helps in reducing the computation of super pixel extraction which avoids overlapping of super pixels also. The same procedure is followed till it reaches P_n.

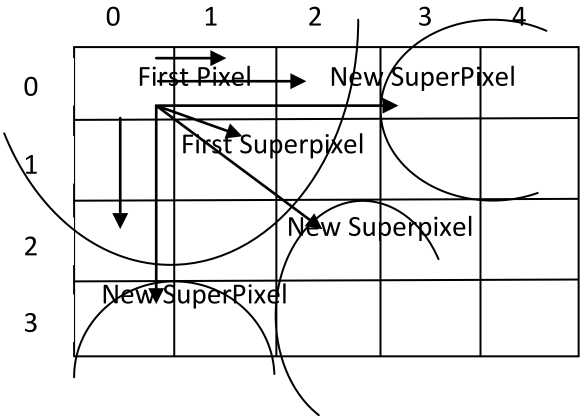

Figure 1: Superpixel Extraction of an Image

This method of super pixel extraction leads to every super pixel approximately with same size and shape. One of the major advantage of this method is every pixels exists within the same super pixel have high correlation and every super pixels have considerably less correlation which will be helpful to easily identify the edges of the object in the image/frame for background subtraction. The space needed to store the pixel values of a image/frame is reduced from thousands to few hundreds. Every super pixel represented in the memory is the representation of group of pixels under the corresponding super pixel.

Compare Pi with P_1 i.e compute P_i-P_1
1. If Pi lies between T_1 and T_2 then current superpixel, increment i then goto 2 else goto 5
2. Create next super pixel along the direction of Pi from current super pixel,
   a) Fix the threshold values
   b) Initialize the current pixel value as initial pixel value of this super pixel
   c) goto 3
3. Stop if P_i=P_n

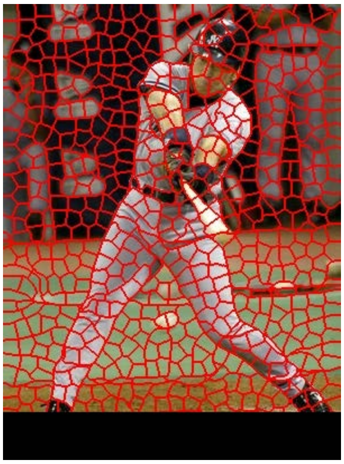

Figure 2: Image after Superpixel Extraction

III. CANNY EDGE DETECTION

Canny edge detection algorithm developed by John F.Canny [7] is the most widely used standard algorithm for detecting the edges of an image/frame for past few years due to its optimal characteristics such as low error rate, well localized edge points, response to single edge etc [6] [8]. It is used for feature detection, feature extraction and to capture the changes in image/every frame of a video in background subtraction. The goal of edge detection is to eliminate the irrelevant data to be processed. This algorithm works in five steps: (1) smoothing, (2) finding gradients, (3) non-maximum suppression, (4) color thresholding, and (5) edge tracking by hyterisis [7]. In this paper, the canny edge detection algorithm is modified accordingly to process the super pixels for background subtraction.

First step is to apply the smoothening step which reduces noise from the image/frame using Gaussian filter [7]. Second step is to find the gradients of the image. The gradients are obtained for every super pixels but not the pixels of the image. The gradient of the image is determined using Sobel-operator as suggested by Canny [7] which uses Euclidean distance to identify the magnitude. Applying gradient for super pixels reduces the computation complexity in a huge manner that is in terms of number of pixels and parameters undergone for gradient measure. After gradient of super pixel is determined the resulting image/frame shows the marked edges along the super pixel boundaries which may not be the actual object edge because a super pixel is a set of sub pixels with similar properties. Thus the obtained image will be of blurred edge (not sharp) due to large super pixel size. The sharp edges are obtained in the next step of Canny edge detection algorithm called non-maximum suppression.

Third step is non-maximum suppression in which the sharp edges are identified from blurred edges [7]. Every sub pixel of every super pixel in blurred edges is undergone for non-maximum suppression such that edge strength of every sub pixel is compared with the edge strength of pixels in the positive and negative gradients. Those pixels that have high local maxima and edge strength in this comparison are taken as edge pixels and others are deleted or suppressed. The output image will be marked with sharp edges but the weak edges are still not yet identified accurately due to noise or some rough surfaces. This objective is achieved in the next step called double thresholding.

Fourth step in edge detection is double thresholding to identify the weak edges of the image/frame using three threshold ranges. . Remaining subpixels in every super pixels after non-maximum suppression is undergone for color thresholding. Canny used double thresholding for grayscale images whereas for color images multiband thresholding should be used. For grayscale images two thresholds has to be set, high threshold and low threshold [7]. For color image/frame, it must be converted from RGB image to HSV and three threshold values are used as discussed by Ming-Hsuan Yang and Narendra Ahuja [10]. The important issue to be considered in color thresholding is selection of threshold ranges because the accuracy of output image/frame is also based on these threshold values. Those subpixels that exits within the threshold value are taken as weak pixels, those lie beyond the threshold are strong pixels and those below the threshold value are suppressed. After identifying the weak edges its time to analyze whether the selected weak edges belongs to the edges of the object or noisy pixels. This job is done by the next step of the Canny edge detection called as edge tracking by hyteresis.

Finally edge tracking decides whether a weak subpixel must be included in the edge of the object or not because weak pixels may exist due to noise or color variations in the image/frame [7]. Every weak subpixel is taken as edge pixel only if there is atleast one strong subpixel exists at the 8 connected neighborhood of this subpixel. After edge tracking the complete edges of the object in image/frame is determined effectively using Canny edge detection algorithm with some specified changes that leads to less computation and storage.

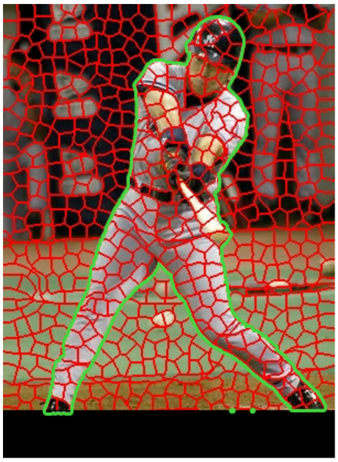

Figure 3: Image after Edge Detection

IV. FUZZY C MEANS

Fuzzy c means is most efficient clustering algorithm developed by Dunn [12] and it was improved by Bezdek [13] for segmenting the image which allows a particular pixel to exist in more than one cluster and its degree of membership in those clusters can be identified [10]. This helps to segment the objects of an image efficiently and accurately [11]. Fuzzy c means algorithm is used in this paper to segment the background and foreground object to perform background subtraction of consecutive frames of a video. This algorithm helps to segment the pixels whether it belongs to background of the frame or not. The main advantage of this method is, identifying the pixel membership in each cluster helps to segment the overlapped objects accurately.

The number of clusters is decided based on the objects present in the frames of video used for background subtraction. One of the most important criteria to be considered in fuzzy c means is choosing the centroid. Initially the centroid is chosen based on the mean of the data points or randomly and the exact centroid can to calculated finally after several iterations of finding the membership degree of pixels. In this paper a new idea is proposed for choosing the centroid based on the superpixels and edge detected using the Canny edge detection. Consider all the superpixels at the left vertical axis of the image/frame, scan every superpixel along horizontal direction till it intersects with superpixel that belongs to the edge of the object marked during the edge detection and let us name it as edge superpixels. The point to be noted is, the strong edge superpixels alone must be taken into consideration for centroid because weak edges may be inner part of the object so during segmentation the entire foreground object is segmented on the whole. Store this edge superpixel and continue scanning horizontally till another edge superpixel is found which means the first edge superpixel and second edge superpixel gives us the boundary of first object in the image/frame. Similarly continue scanning horizontally and note down the third and fourth edge superpixels that represents the boundary of second object in the image/frame.this process continues till the last superpixel in the right vertical axis of the image/frame. If no edge superpixel is found then those superpixels are considered as background superpixels. At the corners of some objects there may not be pair of edge superpixels found hence those edge superpixels without pair belongs to the superpixel of foreground objects.

Now calculate the midpoint between first and second edge superpixel pair. Select any one pixel inside first edge superpixel and second edge superpixel along same row and find the midpoint between them by the formula,

$$Midpoint_{(1,2)} = \frac{(x_1,y_1) + (x_2,y_2)}{2}$$

where (x_1,y_1) is the coordinate of a pixel in first edge superpixel and (x_2,y_2) is the coordinate of a pixel in second edge superpixel. Similarly apply the formula for all edge superpixel pairs to identify their midpoints and mark all those midpoints. Join all those midpoints column wise and find the column which contains more number of midpoints in each object. Mark the coordinates of first and last midpoint pixel in that column and find the midpoint for this column using the above formula. The midpoint obtained from this column acts as the centroid for the object.

Apply the fuzzy c means algorithm and segment the objects in the image. The edge pixels already found using Canny edge detection is also used during segmentation to accurately segment the pixels along the boundaries of clusters.

Figure 4: Image after Segmentation

To perform the background subtraction, apply the above mentioned procedure to every frame of a video for segmentation, then find the difference between the previous frame (PF) and current frame(CF) by subtracting the changes in the pixel positions among both. The change detection of a pixel coordinate (x,y) is obtained from the formula,

$$CD(x,y) = PF - CF$$

V. CONCLUSION

The superpixel extraction method reduces the processing of thousands of pixels into few hundreds of pixels during the edge detection, segmentation and background subtraction techniques. For background subtraction also superpixels may be taken for finding the difference between current frame and previous frame. After finding the change detection, only those superpixels which are found to be changed can be further reduced to identify the changes in its subpixels. So the processing on superpixels improves the computational complexity in a large extent nearly 75% of running time. Also the space needed to store the superpixel values alone is very less such that 25% of storage only when compared to the total size of the image/frame. Those superpixels which needs further processing can only be reduced to its subpixels during processing.

The Canny edge detection algorithm is the standard algorithm proved to be most optimal compare to the other edge detection algorithms. By using this edge detection with fuzzy c means segmentation, both edge detection and segmentation of background object and foreground object can be accurately performed so that background subtraction is done efficiently with elimination of noise and sharp edges. Thus the output frames will have accurate marked edges in it which is very useful for real time applications like video surveillance, medical imaging etc.

VI. ACKNOWLEDGMENT

The author is thankful to her mother Mrs.S.Rajammal for her constant encouragement and motivation. She also wishes to thank Mrs. K.Suganya devi, AP/HOD, Department of Computer Science, Anna University Thiruchirappalli (Panruti Campus) for her valuable guidance and her friends Mr.S.Manigandan, Mr.S.Vignesh and Mr.A.Raja for their timely help and support .

REFERENCES

1. X. Ren and J. Malik. Learning a classification model for segmentation. In Proc. 9th Int. Conf. Computer Vision, volume 1, pages 10-17, 2003. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
2. G. Mori, X. Ren, A. Efros, and J. Malik, Recovering Human Body Configurations: Combining Segmentation and Recognition, IEEE Computer Vision and Pattern Recognition, 2004.
3. G. Mori, Guiding Model Search Using Segmentation, IEEE International Conference on Computer Vision, 2005.
4. Cheung, S.-C. and C. Kamath, "Robust Background Subtraction with Foreground Validation for Urban Traffic Video," EURASIP Journal on Applied Signal Processing, Volume 14, pp 1-11, 2005. UCRL-JRNL-201916.
5. Li Cheng, Minglun Gong, Dale Schuurmans and Terry Caelli, “Real Time Discriminative Background Subtraction”, in IEEE Transactions on Image Processing, Vol 20, No.5, pages 1401-1414, May 2011.
6. Sergei Azernikov. Sweeping solids on manifolds. In Symposium on Solid and Physical Modeling, pages 249–255, 2008.

7. John Canny. A computational approach to edge detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on, PAMI-8(6):679–698, Nov. 1986.
8. F. Mai, Y. Hung, H. Zhong, and W. Sze. A hierarchical approach for fast and robust ellipse extraction. Pattern Recognition, 41(8):2512– 2524, August 2008.
9. Thomas B. Moeslund. Image and Video Processing. August 2008.
10. C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms. New York: Plenum Press, 1981.J.K. Udupa and S. Samarasekera, “Fuzzy connectedness and object definition: Theory, algorithm and applications in image segmentation,” Graphical Models Image Processing, vol.58, no.3, pp.246-261, 1996.
11. J. C. Dunn (1973): "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters", *Journal of Cybernetics* 3: 32-57.
12. J. C. Bezdek (1981): "Pattern Recognition with Fuzzy Objective Function Algoritms", Plenum Press, New York.