

Tracking of Moving Object Using Centroid based Prediction and Boundary Tracing Scheme

Jyotsna Singh

Division of Electronics and Communication Netaji Subhas Institute of Technology
Sector 3, Dwarka, New Delhi-110078, India
Email: jsingh.nsit@gmail.com

Received: 28 March 2016; Accepted: 20 April 2017; Published: 08 August 2017

Abstract—Object tracking has always been a hotspot in the field of computer vision and has myriad applications in the real world. A major problem in this field is that of the successful tracking of a moving object undergoing occlusion in its path. This paper presents centroid based tracking scheme of a moving object without any apriori information of its shape or motion. Once the boundary of the object of interest is obtained, the centroid is calculated from its first order moments. This centroid is further utilized to detect the partial occlusion of test object by some other still or moving object in image frame. In case occlusion is detected, the new centroid location of moving object is predicted for subsequent video frames. The proposed algorithm is able to successfully detect moving object undergoing partial or total occlusion. Experimental results of our algorithm are compared with a popular tracking technique based on Mean Shift tracking algorithm.

Index Terms—Contour based tracking, Linear Prediction, Mean Shift, Object location.

I. INTRODUCTION

Object tracking is defined as the process of locating the position of a moving object or multiple objects in successive video frames. It has numerous applications, some of which are: robot vision, video surveillance, video inpainting, traffic monitoring, animation [1]-[3]. Video tracking is a time consuming procedure because of the size of the data that is contained in the video. Moreover, change in orientation of the moving object over time, noise in the images, loss of evidence caused by estimation of 3D realm on 2D images adds to the complexity of the problems encountered in this field.

Object tracking is one of the recent applications of computer vision. Object tracking can be based on three basic approaches, namely, contour based [4], [5], region based [6] and optical flow based [7], [8]. Region-based technique performs background subtraction preceded by the optical flow measurement. Background subtraction helps in detecting moving objects by subtracting the

estimated background from images. However, this method is sensitive to illumination changes. Active contour based approach is used for finding object boundary from an image. In the contour based tracking algorithm, the objects are tracked by considering their outlines as boundary contours. Thereafter these contours are updated dynamically in successive frames. This approach has been widely adopted in object tracking because the boundary-based features provide reliable information independent of the motion or shape of object. One of the most frequently encountered problems is the improper or entire occlusion of the object by an obstruction between the camera and the moving object. Occlusion can be of multiple types, on the basis of degree of occlusion, partial occlusion, in which only a part of the object is invisible at any particular time, and complete occlusion, in which the object is completely invisible at some point in time. Another type of occlusions can be still occlusion and moving occlusion. In case of still occlusion, object is occluded by a still object such as tree or an electric pole, whereas, moving occlusion occurs due to another object moving in same or opposite direction of object being tracked, such as cars or other vehicles. Fakunaga and Hostetter in 1975, proposed Mean-shift (MS) object-tracking Approach [9], [10]. The algorithm creates a confidence map based on the color histogram of the object in the previous image, into the new image. It then uses mean shift to determine the peak of the confidence map near the object's location. However mean shift algorithm fails to track object in case of occlusion by a still or moving object.

In this paper a contour based method is proposed which utilizes linear prediction, in which the centroid of the target object is determined using boundary of that object. The linear prediction method serves its purpose only when occlusion of object occurs. The centroid, thus calculated is then used to track the object in the subsequent frames.

This paper is structured in the following manner: section II presents the pre-processing of video which is divided into two subsections namely background subtraction and determination of object boundary. Section

III consists of the explanation of proposed tracking mechanism under various conditions of occlusions. In section IV experimental results are presented followed by conclusion in section V.

II. PREPROCESSING OF VIDEO

The diagram shown in Fig. 1 represents the flow of the proposed algorithm for tracking a moving object. It is observed that the objects of interest that are to be tracked change position in a frame from time to time. This necessitates the need to have a moving object detection framework.

The major steps followed are background subtraction, boundary detection and prediction of centroid under partial and complete occlusion. Each processing step is explained in detail in following subsections.

A. Background Subtraction

Background subtraction, is a technique in the area of image processing and computer vision wherein the foreground object is extracted from an image for further processing in various applications such as object recognition, object tracking etc. Generally the objects of interest in an image frame are humans, cars etc. in different types of background.

In order to differentiate the moving object from the background generally there are three methods: optical flow analysis, Frame differencing, and background subtraction [11]-[13]. Frame differencing technique computes the difference between two or three consecutive frames, and thresholds the differential image for detecting foreground. Optical flow is used to distinguish object of interest from background, different sections of same object or different moving objects,. However, optical flow technique is computationally complex. Background subtraction extracts the object by modeling the background and subtracting frame images from it.

In this paper, frame difference method is followed to detect the moving objects. The current frame is subtracted from the previous frame, and if the difference in pixel values for a given pixel exceeds a predefined threshold value then it is considered as part of the object of interest [14].

The reference background image b_j can be obtained by using a j^{th} frame which is not containing the object of interest. The foreground pixels are obtained by taking the absolute difference d_i of i^{th} frame f_i which contains the object of interest and a reference background image frame b_j .

$$d_i = |f_i - b_j| \quad (1)$$

Here both the frames are of size $N \times M$. The frame thus obtained, is converted into binary by calculating the global threshold level of the gray scale image.

$$d'_k(x, y) = \begin{cases} 1; & \text{if } d_k > T \\ 0; & \text{otherwise} \end{cases} \quad (2)$$

All the pixels above this threshold are taken as '1' and those below threshold are taken as '0'. The binary image is then smoothed by a Gaussian filter.

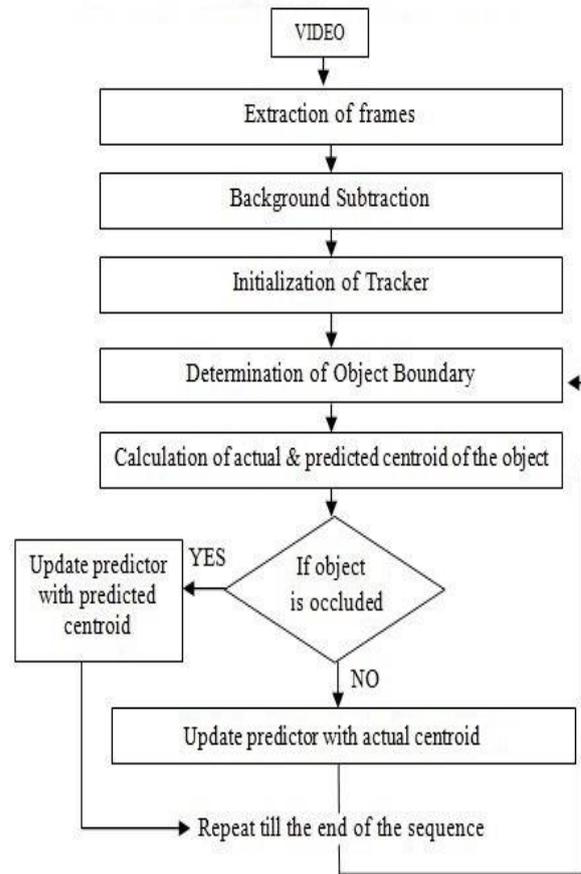


Fig.1. Block diagram of proposed technique for tracking the moving object in a video

B. Boundary tracking

Before initializing the tracker the outline of image is obtained using boundary tracking. An important way of representing a boundary is the so called parametric representation. Normally, the parametric representation of this boundary is obtained by initially defining an arbitrary starting point and traversing the boundary from this point onwards in a clockwise or counterclockwise direction. One of the widely used techniques is 8- neighborhood boundary tracking algorithm or Moore neighborhood tracing method [15].

The proposed boundary tracking method is slightly different from the traditional 8-neighbour tracing method. The detail of technique is explained as follows:

1. First, an initial point belonging to the boundary of the object must be selected. A simple approach to implement this initial step is to start from the frame from where we wish to start tracking and which has the object of interest completely visible.
2. The image frame is traversed row wise and the first black pixel is assigned as the starting pixel of the boundary, as shown in Fig. 2.

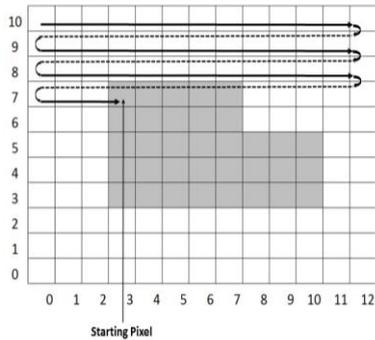


Fig.2. Image frame traversing row wise

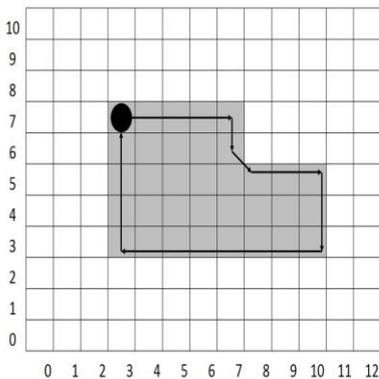


Fig.3. 8-neighbour chain code method

3. After Identifying the black pixel, the algorithm circumnavigates the object until the starting pixel is revisited, indicating the completion of the task. In the proposed technique the 8-neighbour chain code method is followed, as can be seen in Fig. 3.
4. Once the starting pixel has been found, a search for the next neighboring pixel is carried out by placing the mask of 8-neighbour chain code in such a way that its central position lies over the current boundary pixel. The next pixel will be one of the neighbors labeled 0, 1 ...7.
5. In the initial case, where the current boundary pixel is the starting pixel, the algorithm only tests the neighbors 0, 1, 2, and 3. The other neighbors should not be tested since they have already been verified during previous line search (see Fig. 4). The scanner chooses the first black pixel as candidate 0 and carries forward till 3.

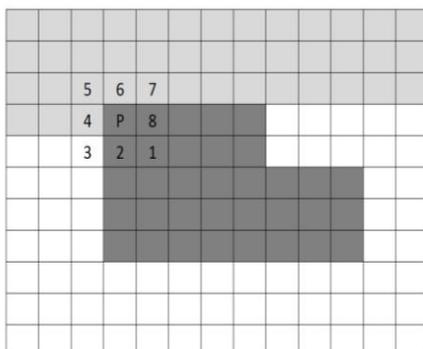


Fig.4. Line search technique for tracking of moving object in a video

6. After the second boundary pixel is found, the algorithm proceeds to track along the boundary in an analogous, but slightly different manner. The algorithm needs two important variables at each step: the second boundary pixel, denoted by $E[n]$ and the direction followed from the previous neighbor to the current pixel, denoted by pcd . The direction from the current pixel to the previous one, denoted as cpd , can be calculated from pcd , using function $invert(d)$ given as

$$invert(d) = mod(d + 4, 8) \quad (3)$$

The modulus function $mod(x, y)$, returns the remainder after the division of x by y . It is now desired to find the next pixel and its direction from the current pixel (denoted by cpd). Starting from cpd , the algorithm tests the neighbors (in clockwise direction) in order to find the boundary pixel candidates, which is done similar to initial step.

7. The method chooses the first candidate from the 8-neighbor chain code direction as the next pixel. Nevertheless, differently from the case, the last candidate is taken as the next boundary pixel. In the existing case, the direction cpd from the previous to the current pixel is 6, implying $cpd = 2$. The algorithm tests the neighbors from $cpd = 3, 4 \dots 7$ to $cpd=1$ following the 8-neighbour chain code direction. The first pixel is looked at as the candidate of the next pixel. After the completion of the given procedure, the vector $E[n]$ will store the information of boundary pixel.
8. The above algorithm stops when it reaches the starting point again, indicating that the object has been circumvented. Finally, the object boundary can be represented by a set of serialized points in parametric form

$$B = \{x(t), y(t)\} \quad (4)$$

where t is the path length of the curve, $x(t)$ and $y(t)$ are x- and y- coordinates of the curve at path length t . The above two sequences of values can be thought of as two discrete signals.

9. In addition, both the starting point and the tracking direction (clockwise) should be defined by convention and used in a consistent fashion. This allows the set of points that compose the parametric contour to be indexed in terms of increasing parametric values. It is worth noting that such signal representation of boundary depends on the starting point in the sense that changing such point implies a shift in signal along the parameter (i.e. t- axis).

C. Edge Detection

Once the boundary and its centroid is obtained, morphological operation [16] is applied to the binary image for edge detection. The morphological operators

implemented are dilation followed by erosion. Erosion changes each object pixel that is touching a background pixel into a background pixel. It removes isolated foreground pixels. In case of erosion, $f(x, y) = 0$, if there is one or more pixels of the 8 neighbors are 0 and is equal to 1 otherwise. The basic effect of the operator on a binary image is to erode away the boundaries of regions of foreground pixels.

Dilation changes each background pixel that is touching an object pixel into an object pixel. Dilation adds pixels to the boundary of the object and closes isolated background pixel. In case of dilation $f(x, y) = 1$ if there is one or more pixels of the 8 neighbors are 1 and is equal to 0 otherwise.

III. TRACKING FRAMEWORK

The contours of all objects present in the frame immediately after the chosen model frame are obtained. Location of center of the object of interest obtained from the model frame is passed on to the aforementioned frame. The entire contour of the object of interest is tracked which gives the perimeter of the contour P in the present frame.

Once the boundary of the object is found, the centroid of the object is obtained using the image moments [17], given by equation

$$\bar{x} = \frac{M_{10}}{M_{00}}; \hat{y} = \frac{M_{01}}{M_{00}} \quad (5)$$

For a gray scale image with pixel intensities $I(x, y)$, raw image moments M_{ij} are calculated by

$$M_{ij} = \sum_i \sum_j x^i y^j I(x, y) \quad (6)$$

This centroid is the location of the object of interest in the current frame.

A. Detection of Occlusion

As can be inferred from previous explanation, that boundary detector starts from a black pixel of the moving object and catches the boundary of object by searching black pixels line by line. As soon as another object hinders the object of interest, the boundary detector adds up the boundary of occluding object with the boundary of object being occluded. As a result, the perimeter of boundary increases. If the perimeter P obtained from the current frame exceeds by a predefined threshold value, it indicates that the object of interest has been occluded by another moving object.

B. Linear Prediction of Centroid

Linear prediction [18], [19] is a mathematical operation in which future values of a discrete time signal are gauged as a linear function of samples from the past. Linear prediction method is incorporated in the proposed algorithm to calculate the predicted value of centroid location \hat{C}_j in next coming frame of video based on the

finite past measurements of the target's centroid namely $\{C_{j-i}; i = 1, 2, \dots, g\}$. The linear predictor of order g can be defined as in:

$$\hat{C}_j = - \sum_{i=2}^g a_i C_{j-i} = -[a_2 C_{j-1} + a_3 C_{j-2} + \dots + a_{g+1} C_{j-g}] \quad (7)$$

The g prediction coefficients $a_2, a_3 \dots a_{g+1}$ are chosen to minimize the mean-squared prediction error, i.e.

$$\epsilon = E[e_j^2] \quad (8)$$

where e_j is the prediction error:

$$e_j = C_j - \hat{C}_j = C_j + a_1 C_{j-1} + a_2 C_{j-2} + \dots + a_g C_{j-g} \quad (9)$$

C_j is the actual centroid as determined by the Image moments method. To compute the prediction coefficients linear prediction method computes the least square solutions to

$$Ca = b$$

Where

$$C = \begin{bmatrix} C(1) & 0 & \dots & 0 \\ C(2) & C(1) & \ddots & \vdots \\ \vdots & C(2) & \ddots & 0 \\ C(g+1) & \vdots & \ddots & C(1) \\ 0 & C(g+1) & \ddots & C(2) \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & C(g+1) \end{bmatrix} \quad (11)$$

$$a = \begin{bmatrix} 1 \\ a_2 \\ \vdots \\ a_{g+1} \end{bmatrix}; b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (12)$$

Least squares problem is solved via the normal equations

$$C^H Ca = C^H b \quad (13)$$

where C^H is the complex conjugate of C matrix. This results in Yule-Walker equations:

$$\begin{bmatrix} R(1) & R(2)^* & \dots & R(g)^* \\ R(2) & R(1) & \ddots & \vdots \\ \vdots & \ddots & \ddots & R(2)^* \\ R(g) & \dots & R(2) & R(1) \end{bmatrix} \begin{bmatrix} a_2 \\ a_3 \\ \vdots \\ a_{g+1} \end{bmatrix}$$

$$= \begin{bmatrix} -R(2) \\ -R(3) \\ \vdots \\ -R(g+1) \end{bmatrix} \quad (14)$$

Where $R = [R(1) R(2) \dots R(g+1)]$ is an autocorrelation estimate for C. These equations are solved by using Levinson-Durbin Recursive algorithm.

Since the boundary of object is tracked from centroid obtained by the moment of moving object, the location of centroid will get disturbed in case of occlusion. Under this circumstance, instead of considering the centroid calculated from present frame, the centroid location predicted from previous frame is utilized for the current frame. Otherwise, the calculated location is passed on to the next frame.

IV. EXPERIMENTAL RESULTS

In order to verify the efficacy and accuracy of the proposed algorithm multiple real world videos containing moving objects were shot in the laboratory. The image sequences were acquired using a camera Nikon D5100 DX format RGB CMOS sensor and then the images were converted to gray scale manually. The videos are of different resolutions. In our linear predictor, we have provided data of the previous 65 frames. The size of the captured image is 320x240 pixels. In order to reduce the processing time, the color image is converted into gray scale image.

A. Tracking of object undergoing occlusion

Still Occlusion: In the case of a still occlusion, background subtraction results in the removal of the occluding object from the video.

Fig. 5(a) shows the object in the video with background subtraction before the occlusion begins. In

Fig. 5(b), the occlusion begins, there is one white blob in the image and the algorithm tracks the centroid of moving object derived from its contour. In Fig. 5(c), (d), part of the object has begun to appear from the left end of the occlusion. There are two blobs of occluded object. As the object moves right to left, the left blob increases in size and the right blob decreases in size. The algorithm is still able to track the centroid of the contour (of the blob) in the right half. In Fig. 5(e), the backside of the object has been completely occluded, and the tracker has jumped to the centroid of the left contour. In

B. Moving Occlusion

In the case of moving occlusion, background subtraction does not remove the occluding object. In Fig. 6(a), the object of interest is not occluded and the tracking is performed by using the actual centroid of the object. In Fig. 6(b), the occlusion has begun; the objects have coalesced into one blob and the centroid predictor is updated with the value obtained from the linear predictor instead of the calculated value. This happens because the perimeter of the blob thus formed exceeds the threshold value above P. In Fig. 6(c) - 6(e), the occluding object has crossed the object of interest. The tracker is successfully holding on to the object of interest.

Fig. 7 and Fig. 8 shows tracker using proposed and mean shift algorithm respectively (when occluding object is still). In Fig. 9 and Fig. 10, the occlusion is moving in the opposite direction from that of our object. In Fig. 11 and Fig. 12, the occlusion is moving in the same direction as that of the object. In Fig. 7, 9 and 11, we have applied the proposed algorithm and as it can be seen, the tracker has been able to track the object successfully. In Fig. 8, 10 and 12, to compare our results, we have applied the mean shift tracking algorithm, and as it can be seen, the tracker has failed to track the object successfully.

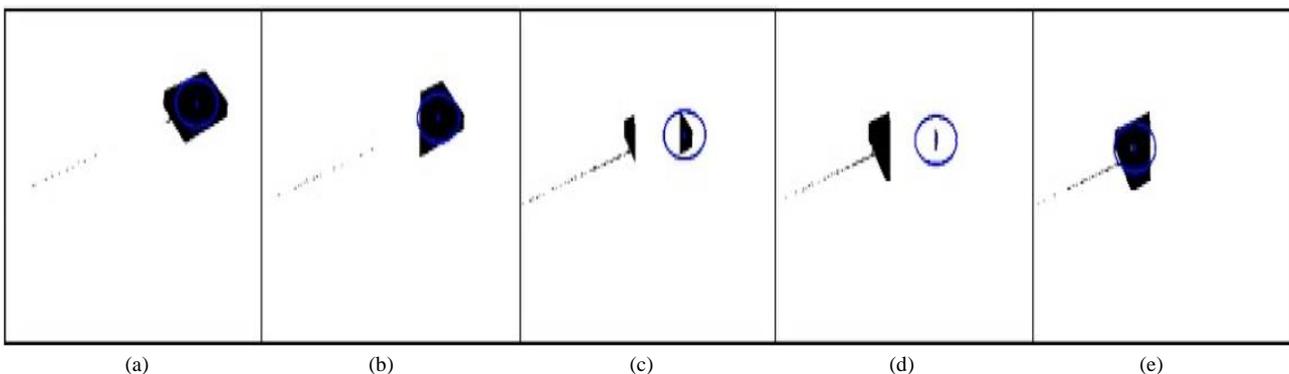


Fig.5. (a) Object without occlusion moving right to left (b) Partially occluded object at upper right portion (c) tracker identifies the partially occluded portion of object (d) tracker identifies a very small portion of object (e) (Blue circle) tracker leaves the right occluded portion and catches the left portion of object

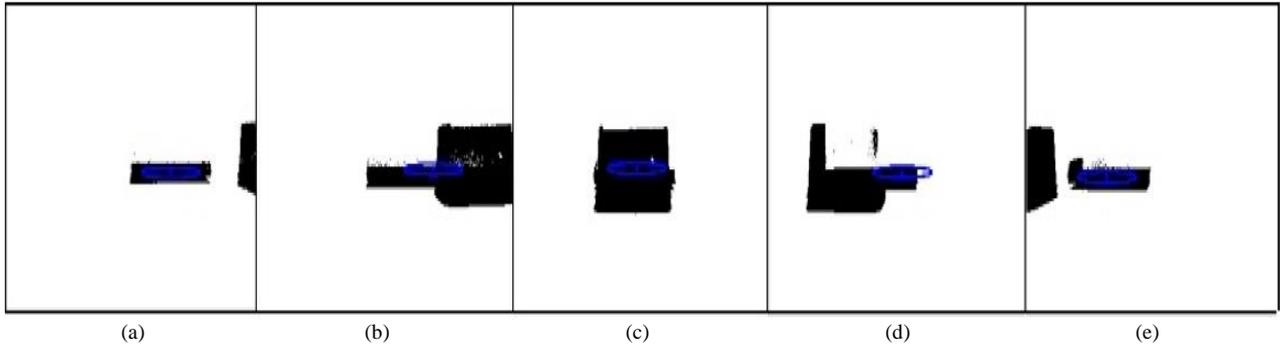


Fig.6. (a) Object without occlusion moving right to left (b) Partially occluded by object coming from right to left (c) tracker identifies the object completely occluded (d) tracker identifies the actual object (e) (Blue circle) tracker still recognizes object after the occluding object leaves the image

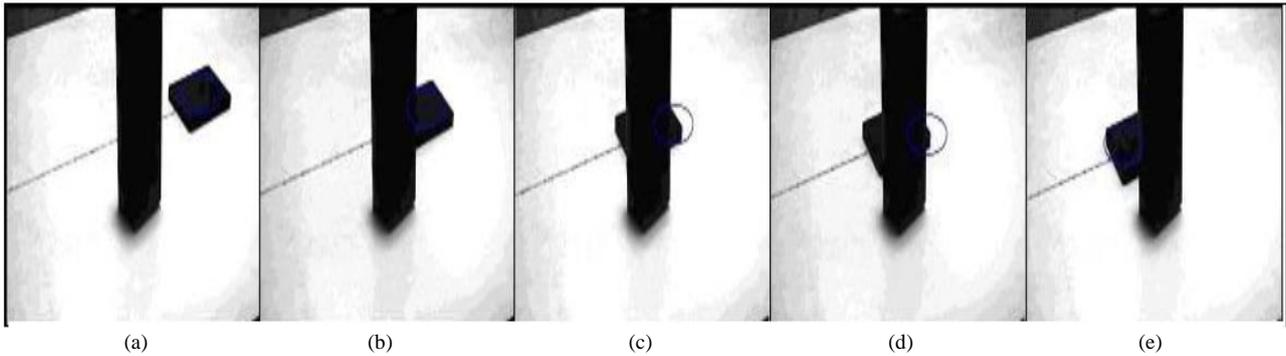


Fig.7. Tracking of object moving from right to left occluded by a pole (using proposed scheme). Blue circle represents tracker.

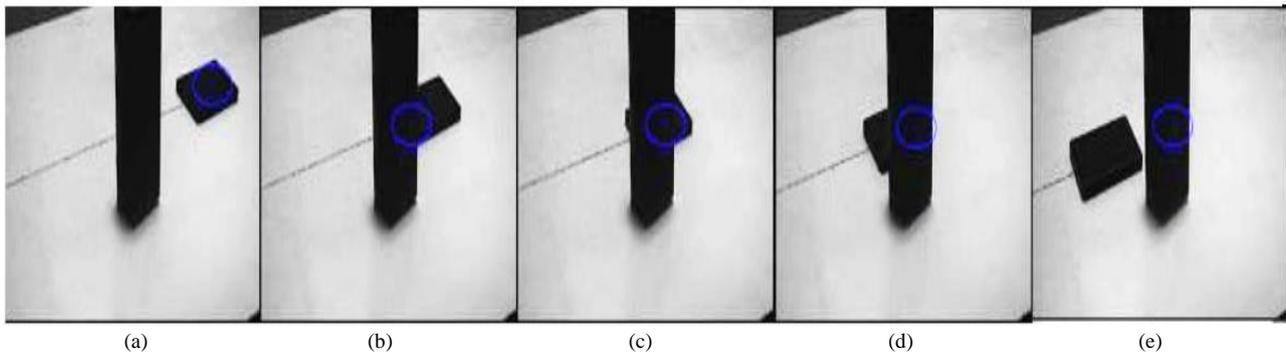


Fig.8. Tracking of object moving from right to left occluded by a pole (using Mean Shift Algorithm). Tracker fails to follow the moving object occluded by still object.

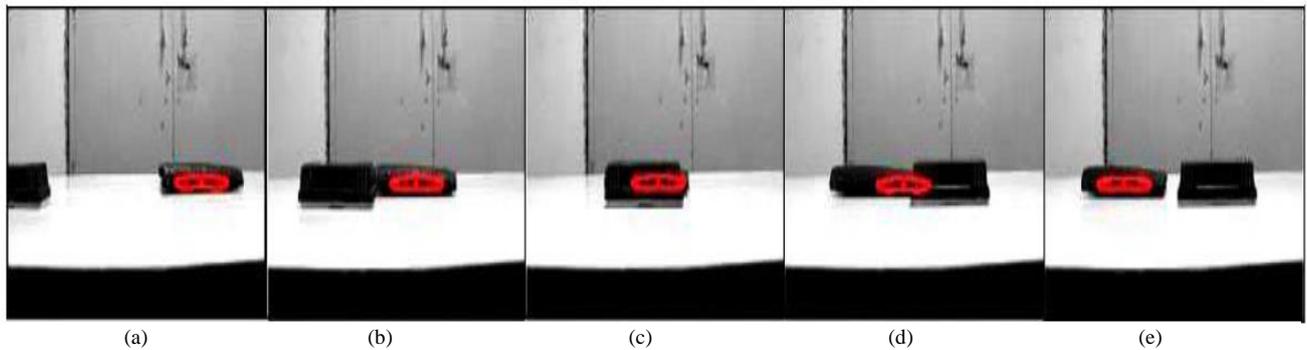


Fig.9. Tracking of object moving from right to left occluded by a moving object (using proposed Algorithm). Tracker (red circle) succeeds to follow the moving object occluded by moving object. Occlusion is moving in the opposite direction from that of our object.

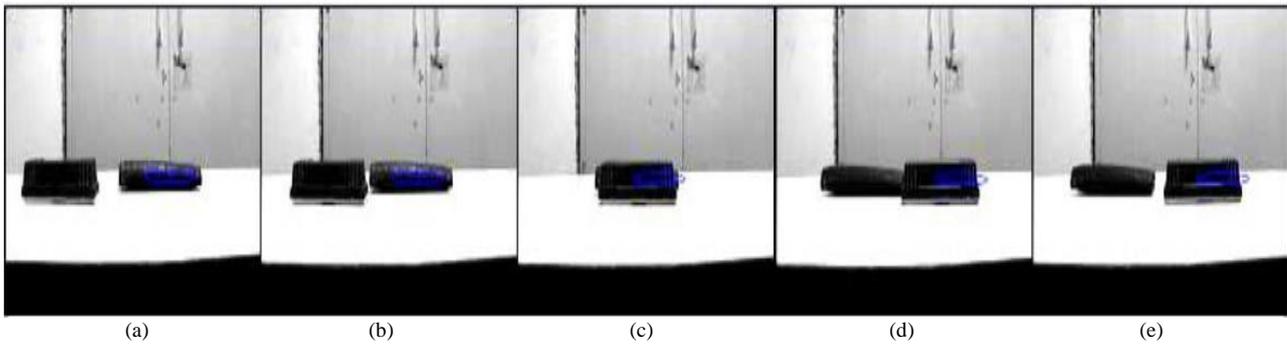


Fig.10. Tracking of object moving from right to left occluded by a moving object (using Mean Shift Algorithm). Tracker fails to follow the moving object occluded by still object. Occlusion is moving in the opposite direction from that of our object.

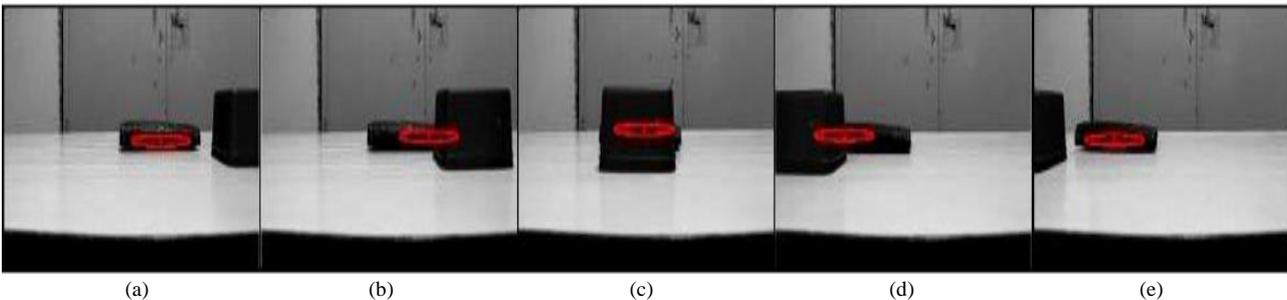


Fig.11. Tracking of object moving from right to left occluded by a moving object (using proposed Algorithm). Tracker succeeds to follow the moving object occluded by moving object. The occlusion is moving in the same direction as that of the object

V. CONCLUSIONS

The paper presented a centroid based boundary detection algorithm. The centroid is obtained from first order moment of moving object. The technique is utilised for detecting the occurrence of occlusion. In case the occlusion is detected centroid of partially or completely occluded moving object is obtained from previous frames using linear prediction. We have tested our algorithm on real world videos with different types of high degree occlusions. The results show that the algorithm has been able to successfully track the object across all the videos. The proposed algorithm requires that there should be colour contrast between the background and the object to be tracked. However, the proposed algorithm works efficiently in the scenario when the object of interest and the occluding object are of the same colour.

REFERENCES

- [1] T.Zhao, R.Nevatia, "Tracking multiple humans in crowded environment", CVPR, Washington DC, 2004, pp.406-413.
- [2] F.Oberti, S.Calcagno, M.Zara, "Robust tracking of humans and vehicles in cluttered scenes with occlusions", IEEE International conference on image processing, Rochester, New York, USA, 2002.
- [3] L. Li, S. Ranganath, H. Weimin, and K. Sengupta "Framework for Real-Time Behavior Interpretation From Traffic Video" IEEE Tran. On Intelligen Transportation Systems, , Vol. 6, No. 1, pp. 43-53, 2005.
- [4] Michael Kass, Andrew Witkin, and Demetri Terzopoulos, "Snakes: Active contour models", IJCV, 1 (4):321-331, 1988.
- [5] Laurent D.Chhen and Ron Kimmel, "Global Minimum for Active Contour Model: A Minimal Path Approach", Int. J of Computer Vision, vol. 24, pp.57-78, 1997
- [6] D. Xu, Jenq-Neng Hwang, Jun Yu, "An accurate region based object tracking for video sequences", IEEE 3rd Workshop Multimedia Signal Processing, pp. 271-276,1999.
- [7] M.Yachida, M.Asada, and S.Tsuji, "Automatic analysis of moving image", IEEE Trans. Pattern Anal. Mach. Intell. Vol.PAM1-3, No.1, pp.12-20, 1981.
- [8] D. Comaniciu, V. Ramesh, P. Meer: Real-Time Tracking of Non-Rigid Objects using Mean Shift, IEEE Conf. Computer Vision and Pattern Recognition (CVPR'00), Hilton Head Island, South Carolina, Vol. 2, 142-149, 2000.
- [9] R. T. Collins, "Mean-shift blob tracking through scale space", In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2003.
- [10] D. Comaniciu, V. Ramesh, P. Meer, "Real-Time Tracking of Non-Rigid Objects using Mean Shift", IEEE Conf. Computer Vision and Pattern Recognition (CVPR'00), Hilton Head Island, South Carolina, Vol. 2, 142-149, 2000
- [11] Haizho, A. & Xu Guangyou, "Moving object detection and tracking based on background subtraction", Proceeding of Society of Photo-Optical Instrument Engineers (SPIE), Vol. 4554, pp. 62-66, 2001.
- [12] Lipton A; Fujiyoshi, H. & Patil, R., "Moving target classification and tracking from real-time video", Proceeding of IEEE Workshop Applications of Computer Vision, pp. 8- 14, 1998.
- [13] Desa, S. M. & Salih, Q. A. , "Image subtraction for real time moving object extraction", Proceeding of Int. Conf. on Computer Graphics, Imaging and Visualization (CGIV'04), pp. 41-45, 2004.

- [14] Piccardi, Massimo. "Background subtraction techniques: a review." *Systems, man and cybernetics*, 2004 IEEE international conference on. Vol. 4. IEEE, 2004.
- [15] Toussaint, G. Grids, Connectivity and Contour Tracing. Available online: <http://www-cgri.cs.mcgill.ca/~godfried/teaching/pr-notes/contour.ps>.
- [16] Stringa, E.,". Morphological change detection algorithms for surveillance applications, " *Proceeding of British Machine Vision Conf.*, pp. 402–412, 2000.
- [17] R. Mukundan and K. R. Ramakrishnan, *Moment Functions in Image Analysis: Theory and Application*, Singapore: World Scientific, 1998.
- [18] Jackson, L. B. *Digital Filters and Signal Processing*. 2nd Edition. Boston: Kluwer Academic Publishers, 1989, pp. 255–257. J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.
- [19] Accurate real-time object tracking with linear prediction method. P.Y. Yeoh and S.A.R. Abu-Bakar Dept. of Microelectronics and Computer Engineering Faculty of Electrical Engineering Universiti Teknologi Malaysia 81310 Skudai, Johor, Malaysia.

Authors' Profiles



Dr. Jyotsna Singh received her B. Tech degree in Electronics from Harcourt Butler Technological Institute, Kanpur, India in 1995 and M. Tech degree in Signal Processing from Netaji Subhas Institute of Technology, Delhi University, Delhi, India, in 2001. She is working as Assistant Professor in Netaji Subhas Institute of Technology, Delhi University, Delhi for the last 14 years. She has been teaching courses on Digital Signal Processing, Adaptive Signal Processing and Electronics. She received her Ph.D degree in Electronics and Communication Engineering from the University of Delhi, India. Her research interests include Speech Recognition, Multimedia Security and Image Compression.
Email: jyotsna@nsit.ac.in
Phone: 011-25000118

How to cite this paper: Jyotsna Singh, "Tracking of Moving Object Using Centroid based Prediction and Boundary Tracing Scheme", *International Journal of Image, Graphics and Signal Processing(IJIGSP)*, Vol.9, No.8, pp.59-66, 2017. DOI: 10.5815/ijigsp.2017.08.07