ISSN 2229-5518

A new approach using Camshift Algorithm for multiple Vehicle Tracking

Mohammed Mohsin AL- Azzi, Prof. Huang Dong Jun, Thekra Abbas

Abstract— Cameras and video technology have become integral in our day to day lives. Surveillance is one area that has greatly benefited from video technologies. This in turn increases the need for automatic video surveillance algorithms that can track objects and raise alarm if needed. Tracking of people is one such area. On the other hand, CAMSHIFT is a tracking algorithm that has been widely applied in face tracking in the past. It has however not been used in vehicle tracking. This paper therefore presents a modified CAMSHIFT algorithm that can be used in tracking of vehicles in video sequence. This will be covered in two ways; detection of moving objects of interest from frame to frame then evaluate the performance of our modified CAMSHIFT algorithm on different video sequences. We'll use frame difference to achieve object tracking. The results reveal that.

Keywords: CAMSHIFT algorithm, tracking, shadow removal

.

1 Introduction

Object tracking is one of the most important tasks in computer Surveillance In video, assists in understanding the movement patterns of people to detect suspicious vehicles or events ,Object tracking can be divided into series of steps, such as object representation, feature selection for tracking, object detection background subtraction, and object segmentation. The detection of a moving object and tracking of different objects in a video or video sequence is a very important task in the surveillance videos, analysis and monitoring of traffic, tracking and detection of humans and different gesture recognitions in human-machine interface [1]. Object trackers can be categorized into three different categories such as point trackers, kernel trackers and silhouette trackers. Point tracking methods can be further sub-categorized into deterministic methods such as [2, 3] and statistical methods [4, 5]. Kernel based object tracking Objects could be represented with the help of a single point (e.g. centroid) [1] or a set of points [2], by primitive geometric shapes (rectangles or ellipses) [3], silhouettes and contours for tracking complex non-rigid objects[4]. Feature selection also plays an important role in object tracking. The most commonly used features are color, edges, optical flow and texture [6,7] Various color spaces other than RGB are used for tracking purposes such as HSV because RGB color space does not correspond to the color differences perceived by humans [8,9]. Commonly used edge detectors are Canny Edge detector [10] .Background modeling can also the basic step in many video analysis applications used to extract foreground or moving objects from the video frames.
Change in a scene or foreground objects could be extracted from a video sequence by subtracting the background image from each frame. In general the background is considered to be constant or slowly changing due to luminance changes. In practice the background pixels are always changing, for that reason we need a model which accounts for gradual changes. Several techniques have been proposed in literature for modeling the variation of the background information [11, 12, and 13]. Modeling of each stationary background color pixel with a single 3D (YUV color-space) Gaussian was proposed. A single 3D (YUV) Gaussian however is not suitable for outdoor scenes [14] since in certain location multiple colors can be seen due to repetitive motion, shadows or reflectance. Thus the Mixture of Gaussians (MoG) for modeling a single pixel color was proposed in [15].
In this paper, we use MoG method for background modeling. Contour detection provided by Intel Open Source Computer Vision Library (OpenCV) [16] is used for object detection and representation. CAMSHIFT kernel based object tracking is used to track the object with color information using Hue Saturation Value color-space, as it has been considered as a better representative of color perceived by human vision [17]. Motion contour detection. Further we proceed with mask image formation for extracting exact object which is to be tracked [18].

* *Mohammed Mohsin AL- Azzi, Ministry of Higher Education Iraq, Research & Development Department.*
* *Huang Dong Jun is a Professor in school of information science and engineering in central south university, china.*
* *Thekra Abbas, College of Science, Almustanseriah University Baghdad, Iraq*

2 Tracking of Moving Object

2.1 Camshift Algorithm Principle

Camshift and particle filtering algorithms have become the main object tracking algorithms, in which the camshift algorithm is the most widely used tracking algorithm. Camshift algorithm is a nonparametric method with dynamic distribution and gradient estimate by using probability density function and is used for quick and efficient tracking based on the color probability distribution of the object [19].
It extends the mean shift algorithm to a continuous image sequence, making mean shift operation for all frames of the video image and taking the results of the previous frame as the initial track win down value of the next frame mean shift algo-

ISSN 2229-5518

rithm. object tracking, but when the background is complex and the objects are increased, Camshift tracking results will be greatly affected. Meanwhile, the traditional Camshift tracking window size is fixed, tracking window cannot adaptively adjust its size according to the size of the target, which leads to low tracking accuracy when the object shape and size are frequently changed various rates.

2.2 Improvement of Camshift Algorithm

The traditional Camshift algorithm uses color histogram as characterization of the target template for matching and only contains the frequency of a certain color value occurred in the image and loses the space location information of the pixel, when the background color is similar to the object and the emergence of occlusion and the background and occlusion will generate interference, resulting in the tracking method based on color histogram losing the target; meanwhile, the traditional color histogram is a global color histogram, computation is larger and is not conducive to real-time tracking [20].

Therefore, the improved algorithm will continue the good characteristics of the color histogram and use a simplified main color histogram and an edge direction histogram which has robustness for the situation that the occlusion and background object color are similar, together to describe the moving object, thus achieving the accurate and robust moving object tracking in complex environments. For the above deficiencies for Camshift algorithm, this paper will improve it, and the specific improvement measures are as follows, firstly the quadratic difference method based on background motion estimation for object extraction is used, and the tracking process integrates the simplified main color histogram and the above-concerned edge direction histogram as the target characteristics secondly, the obtained center is the initial search point, for similarity detection of the target tem plate and the candidate target, and we can search the optimal matching position and then to get the exact location of the target object in the current frame.

The specific implementation process of the improved Camshift algorithm is shown in Figur.1The improved algorithm introduces the object detection algorithm based on background motion parameter estimate, to achieve the effective tracking of the traditional Camshift algorithm in the dynamic context, thus expanding its range of applications. Meanwhile, in order to improve its tracking robustness in dynamic Moving Object Tracking in Video Surveillance System

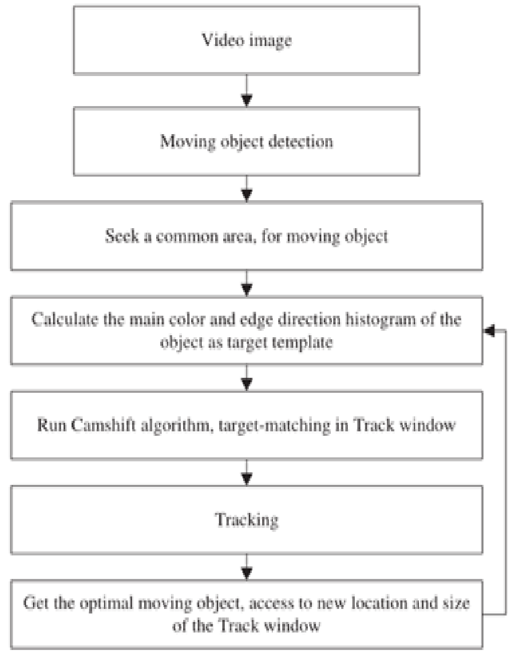

figure. 1 Implementation process of improved Camshift algorithm

context, the improved algorithm continues the good characteristics of the color histogram and uses a simplified main color histogram and an edge direction his -togram which has robustness for the situation that the occlusion and background object color are similar, together to describe the moving object, which greatly improves the anti-jamming capability and robustness of tracking [21].

3 VEHICLE Tracking in the street

3.1 Background Estimation and Post Processing

In this study Vehicle tracking, we first used Mixture of Gaussian (MoG) method to calculate a good background model. The Open CV functions are an implementation of the Gaussians Mixture Model in [22]. In this implementation, the model assumes that each pixel in the scene is modeled by a mixture of K Gaussian distributions where different Gaussians represent different colors. The weight parameters of the mixture represent the time proportions that those colors stay in the scene. Thus, the probable background colors are the ones which stay longer and are more static. The probability that a certain pixel has a value of x N at time N can be written as p(x N) as shown in eq.(1).

ISSN 2229-5518

$$p(x_N) = \sum_{j=1}^{k} w_j \cdot \eta(x_N, \theta_j) \qquad (1)$$

Where w_j is the weight parameter of the j th Gaussian component. η (x_N ;θ_j) x is the Normal distribution of the j th component. Static single-color objects trend to form tight clusters in the color space while moving ones form widen clusters due to different reflecting surfaces during the movement. The measure of this was called the fitness value [23]. The K distributions are ordered based on the fitness and the first B distributions are used as a model of the background of the scene where B is estimated as shown in (2) equation.

$$B = \arg\min_{b} \sum_{j=1}^{b} w_j > T \qquad (2)$$

Minimum fraction of the background model is threshold T. In other words, it is the minimum prior prospect that background is in the view. Background subtraction is performed by marking a foreground pixel any pixel and that is more than 2.5 standard deviations away from any of the B distributions. Figure.1 shows the original frame and the background obtained after applying Gaussian mixture model.

Figure 1. An example of the background results obtained with the Adaptive Gaussian Mixture Model

Next, we need to remove the shadows in the foreground obtained with the previous background subtraction method. In [24] the authors assume that we can consider a pixel as shaded background or shadow if it has similar chromaticity but lower brightness than those of the same pixel in the background image. Thus, with an appropriate threshold T, we can remove shadows from the foreground image. Eq. (3) shows the decision either a certain pixel belongs to shadow or not.

$$Shadow\{x,y\} = \begin{cases} 1 & if\ brightness_{img} < brightness_{bg} \\ & \&chromaticity_{img} = chromaticity_{bg} \pm T \\ 0 & otherwise \end{cases} \qquad \textbf{(3)}$$

The shadow removal is applied on our sample video sequence. Figure.2 shows a frame from the video before and after shadow removal.

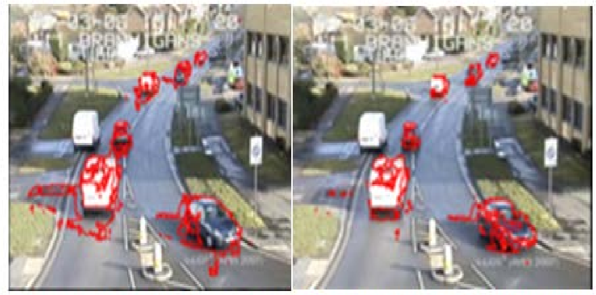

Figure 2. Foreground objects after background subtraction (left) and after shadows removal (right).

Eventually, the morphological filtering is applied using closing operation. Unlike the eventually, the morphological filtering is applied using closing operation. Unlike the opening which removes small objects, closing removes small holes. Figure.3 shows one frame of our sample video sequence before and after the application of morphological closing operation.

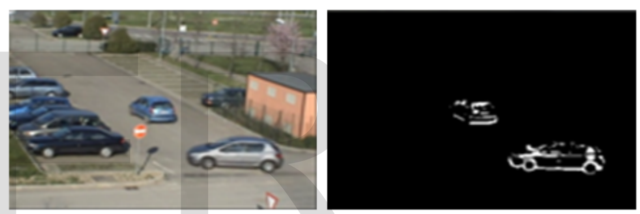

Figure 3. Foreground objects without morphological

We can see that the closing operation removes small holes in the foreground, and also small points from the foreground around the vehicle. After the background subtraction, we apply Vehicle Tracking in detection algorithm, to detect all separate Vehicles in the frame. For this, we proceed in two steps: As a first step a contour is detected. A contour is represented in OpenCV by a sequence of points. First, we use the OpenCV function to find contours around all separate foreground regions. In our case, we retrieve only the extreme outer contours and the function compresses, vertical, horizontal, and diametrical segments, leaving only their ending points. Then, for each found contour, we define the smallest bounding box in which all the points of the contour are included, and we calculate its area. If this area is under some threshold, we remove this contour from the sequence to keep only contours of a potential vehicle and remove those that cannot be a whole vehicle.

After that, we need to define a bounding box around the detected contour. The difficulty is that most of the time, the contour is not perfect, and we may have two distinct contours for the same vehicle, for example, one for body and other for the head or the arms. These contours are in general very close or even overlapping. So, we just define a bounding box around the different contours, and detect which bounding boxes are

ISSN 2229-5518
overlapping or very close and merge them into one single box. We perform this operation for each contour, and then create a sequence of bounding boxes, corresponding to the different main areas in the foreground. The detected contour and bounding box in one of the frame of our sample video sequence is shown in figure.4.

3.2 Vehicles TRACKING by CAMSHIFT

Now, we need to track the detected Vehicles, and for this, we will use the CAMSHIFT algorithm. The CAMSHIFT algorithm can be summarized in the following steps [22]:
(1) Set the region of interest (ROI) of the probability distribution image to the entire image.
(2) Select an initial location of the Mean Shift search window. The selected location is the target distribution to be tracked.
(3) Calculate a color probability distribution of the region centered at the Mean Shift search window.
(4) Iterate Mean Shift algorithm to find the centroid of the probability image. Store the 0th moment (distribution area) and centroid location.
(5) For the following frame, center the search window at the mean location found in step 4 and set the window size to a function of the 0 th moment. Then go to Step 3. The creation of the color histogram corresponds to steps 1 to 3. The first step is to define the region of interest (ROI) which is the bounding box corresponding to the detected vehicle that we want to track. Then, we need to calculate the color histogram corresponding to this vehicle.
For that we use the HSV color space, and calculate a one dimensional histogram corresponding to the first component: hue. We also define a mask for the histogram calculation, which is the foreground image, to calculate the histogram only for the vehicle, and not for the background inside the bounding box. But the results obtained with this method were not satisfying, because in the case where the background has almost the same color as the vehicle, it is not possible to detect the difference between the two in the back-projection image. That is why we decided in a second time to use a three dimensional histogram, using the three components of the HSV color space: hue, saturation, value. With this method, we were able to find the location of the vehicle in the whole frame, even with a similar background. In all cases the histogram bins values are scaled to be within the discrete pixel range of the 2D probability distribution image using eq.(4).

$$\left\{ \hat{p}_u = \min\left(\frac{255}{\max\{\hat{q}\}}\hat{q}_u, 255 \right) \right\}_{u=1\cdots m} \quad (4)$$

at is, the histogram bin values are rescaled to the range [0, 255], where pixels with the highest probability of being in the sample histogram will map as visible intensities in the 2D histogram back- projection image. Now we have a histogram of the moving Vehicle, we need to find this vehicle in all the frames (steps 4 to 5). For that end, we calculate the back-projection of the histogram in the subsequent frame. For each pixel of all input images we put, in the back-projection image, the value of the histogram bin corresponding to the pixel. In terms of statistics, the value of each output image pixel is the probability of the observed pixel that it is a pixel of the tracked object, given the distribution (**histogram**). Eventually, using the previous location of the vehicle, we detect the new position of the moving vehicle and use it as starting search window for the next frame. The search window center could be computed from:

$$M_{00} = \sum_x \sum_y I(x,y) \quad (5)$$

$$M_{10} = \sum_x \sum_y x I(x,y) \quad (6)$$

$$M_{01} = \sum_x \sum_y y I(x,y) \quad (7)$$

$$x_c = \frac{M_{10}}{M_{00}}; y_c = \frac{M_{01}}{M_{00}} \quad (8)$$

Where M_{00} in eq.(5) is the $zero^{th}$ moment, M_{10} in eq.(6) and M_{01}in eq.(7) are the first moments, these moments could be used to compute the next center position of the tracking window x c and y c as shown in eq.(8). Then, back to step 3, we calculate the new histogram of the Vehicle to update the previous one, using a slow update, to keep the difference between different vehicles if they are overlapping. The vehicle tracked and their corresponding tracking windows using CAMSHIFT algorithm are shown in figure.5. But when Vehicles are crossing, it happens that both of the tracking windows follow the vehicle in the foreground, for example if the two vehicles have similar colors.

4. VEHICLES tracking using CAMSHIFT

In order to solve this problem, we decided to add more information to the back-projection image, and we decided to use the motion coherence .When two Vehicles are going in two opposite directions, the motion will allow us to follow the right vehicle.

4.1. Experimental Results

4.1.1 Single vehicle tracking

In this study the background subtraction we apply Single vehicle detection algorithm, all separate vehicles in the frame find contours around all separate foreground regions , for each , found contour we define the smallest bounding circle in which all the points of the contour are included and we calculate its area, function to find contours around all separate foreground regions If this area is under some threshold, and then create a sequence of bounding circles The detected contour and bounding circle in one of the frame of the effectiveness video sequence is shown in figure.4

Fig.4The detected contour and bounding circle

4.1.2 Multi vehicle tracking

In this paper we proposed the effectiveness video is used to the CAMSHIFT algorithm with motion information track the multi vehicle, before and after the occlusion which means color information is enough to track the vehicle in occlusion for effectiveness in figure (5). It has the same color the camshift algorithm it

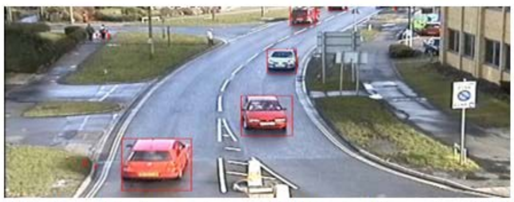

Figure .5 Each vehicles has contours (Vehicles tracking using CAMSHIFT only).

has ability to distinguish the vehicles color similar or various .Same video is used to test our proposed tracking algorithm using both color histogram and motion information; the video frames before and after occlusion are shown in figure.
We can see here effectiveness of multi vehicles crossing in the opposite directions; the motion in this case allows us to track multi vehicle after the occlusion. The tracking windows have the same tracking numbers as before and after the occlusion that shows the algorithm can track the vehicles even after occlusion before and after the occlusion that shows the algorithm

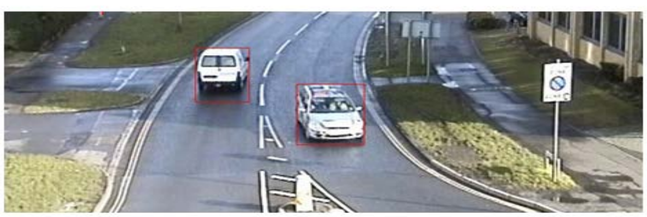

Figure .6 contour multi vehicles crossing in the opposite

can track the vehicle even after occlusion.

5. CONCLUSIONS AND FUTURE WORK

From the present study we can conclude that camshift algorithm can effectively detect the multi vehicles. Hence in future we can extend our work by combining camshift and optical flow algorithm to differentiate between cars and humans the reason behind it when a car come close to an human or in case of any accident we cannot recognize weather it is an human or car because if car comes close its contour becomes bigger and human body will be victim of occlusion by using some useful algorithms as discussed above we can resolve the problems in video surveillance.

REFERENCES

[1] Veenman, C. Reinders, M., and Backer, E. 2001. Resolving motion cor respondence for densely moving points. IEEE Trans. Patt. Analy. Mach. Intell. 23, 1, 54–72.

[2] SALARI, V. AND SETHI, I. K. 1990. Feature point correspondence in the presence of occlusion. IEEE Trans.Patt. Analy. Mach. Intell. 12, 1, 87–91.

[3] VEENMAN, C., REINDERS, M., AND BACKER, E. 2001. Resolving motion correspondence for densely moving points. IEEE Trans. Patt. Analy. Mach. Intell. 23, 1, 54–72.

[4] BROIDA, T. AND CHELLAPPA, R. 1986. Estimation of object motion parameters from noisy images. IEEE Trans.Patt. Analy. Mach. Intell. 8, 1, 90–99.

[5] STREIT, R. L. AND LUGINBUHL, T. E. 1994. Maximum likelihood method for probabilistic multi-hypothesis tracking. In Proceedings of the International Society for Optical Engineering (SPIE.) vol. 2235. 394–405.

[6] SERBY, D., KOLLER-MEIER, S., AND GOOL, L. V. 2004. Probabilis tic object tracking using multiple features. In IEEE International Confe rence of Pattern Recognition (ICPR). 184–187.

[7] COMANICIU, D., RAMESH, V., ANDMEER, P. 2003. Kernel-based object tracking. IEEE Trans. Patt. Analy. Mach Intell. 25, 564–575.

[8] YILMAZ, A., LI, X., AND SHAH, M. 2004. Contour based object tracking with occlusion handling in video acquired using mobile cam eras. IEEE Trans. Patt. Analy. Mach. Intell. 26, 11, 1531–1536.

[9] HARALICK, R., SHANMUGAM, B., AND DINSTEIN, I. 1973. Tex tural features for image classification. IEEE Trans.Syst. Man Cybern. 33, 3, 610–622

[10] LAWS, K. 1980. Textured image segmentation. PhD thesis, Elec tric al Engineering, University of Southern California.

[11] PASCHOS, G. 2001. Perceptually uniform color spaces for color tex ture analysis: an empirical evaluation. IEEE Trans. Image Process. 10, 932–937.

[12] CANNY, J. 1986. A computational approach to edge detection. IEEE Trans. Patt. Analy. Mach. Intell. 8, 6, 679–698.

[13] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, “Pfinder: Real- time tracking of the human body,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, pp. 780-785, 1997.

[14] A. Monnet, A. Mittal, N. Paragios, and V. Ramesh, “Background modeling and subtraction of dynamic scenes,” Oct. 2003, pp. 1305-1312 vol.2.

[15] M. Irani and P. Anandan, “Video indexing based on mosaic repre sent tations,” Proceedings of the IEEE, vol.86, no. 5, pp. 905-921, May1998.

[16] X. Gao, T. Boult, F. Coetzee, and V. Ramesh, “Error analysis of back ground adaption,” vol. 1, 2000, pp. 503-510 vol.1.

[17] C. Stauffer and W. Grimson, “Learning patterns of activity using real- time tracking,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 22, no. 8, pp. 747-757, Aug 2000.

[18] Power PW, Schoonees JA (2002) Understanding background mixture models for foreground segmentation. Proceeding image and vision com puting, New Zealand 11(06):267–271.

[19] Alexandropoulos T, Loumos V, Kayafas E (2004) A Block clus tering technique for real-timeobject detection on a static back ground,2nd International IEEE conference on intelligent sys- tems, vol 23(1),pp 169–173.

[20] Elgammal A, Harwood D, Davis LS (2000) Background and fore ground modeling using non-parametric kernel density estima tion for visual surveillance. IEEE 90(7):1151–1163.

[21] Z. Zhong (ed.), Proceedings of the International Conference on Infor mation Engineering and Applications (IEA) 2012, Lecture Notes in Electrical Engineering 217, DOI: 10.1007/978-1-4471-4850-0_26, © Springer-Verlag London 2013.

[22] PASCHOS, G. 2001. Perceptually uniform color spaces for color tex ture analysis: an empirical evaluation. IEEE Trans. Image Process. 10,932–937

[23] K.J. Somaiya ,Multiple Objects Tracking Using CAMShift, I.J. Image, Graphics and Signal Processing, 2013, 6, 43-48.

[24] Elgammal A, Harwood D, Davis LS (2000) Background and foreground modeling using non-parametric kernel density esti mation for visual surveillance. IEEE 90(7):1151–116323]] KANADE, T., COLLINS, R., LIPTON, A., BURT, P., AND WIXSON, L. 1998. Advances in cooperative.