Improved Fast Mean Shift Algorithm

Mahmoud Mirabi [1], Saman Hosseini [2], Kaveh Keyvani Asl [3]

[1] Islamic Azad University, Tehran Central Branch, Dept. of Electrical Engineering, Tehran, Iran
e-mail: mahmoud.mirabi@gmail.com
[2] Islamic Azad University, Qazvin Branch, Dept. of Electrical Engineering, Qazvin, Iran,
e-mail: Saman_ho12345@yahoo.com
[3] Islamic Azad University, Qazvin Branch, Dept. of Electrical Engineering, Qazvin, Iran,
e-mail: kaveh.keyvani@gmail.com

Abstract: *In this paper an enhanced version of Fast Mean-Shift for real time multiple human tracking is presented. Previous works demonstrated that tracking by Kalman filter amended with Fast Mean-Shift, constructs a robust framework for multi-Human tracking. An important part of Fast Mean-Shift algorithm is the computing of difference image. In this work we addressed two problems that make fault in computing of difference image. These problems often occur during the human bodies tracking in a pathway where current background often changes with respect to reference image. The proposed method was tested for real word data and the simulation results show the feasibility and efficiency of the proposed method.*

1 Introduction

Multiple object tracking is an extensively investigated subject in field of visual surveillance [1]. The main difficulty comes from the fact that observed data may be contaminated with noise of scene, missing observations or clutter. Such problems can encounter for instance in segmentation-based object detection [1,2], where under-segmentation, over-segmentation and false detections are frequent problems ultimately lead to failures when tracking is performed. Recent works showed that statistical methods are valuable to extract the change regions from background. The Gaussian mixture model (GMM) is the most representative background model. As an example, Stauffer et al,[3] presented an adaptive background mixture model for real-time tracking.
Kalman filters are very important tools and are often used for tracking moving objects. Kalman filters are typically used to make predictions for the following frame and to locate the position or identify the related parameters of moving object [4]. The combination of GMM and Kalman filter becomes a good framework for multiple object tracking.
For the image segmentation problem, Mean-Shift Clustering is commonly used. Comaniciu et al. [5, 6] proposed the mean-shift approach to find clusters in joint spatial and colour space. A recent work also combining the Kalman filter and Mean-Shift Algorithm appeared in [7] , as using the Kalman filter to predict the possible position on the object in next frame of video image, and the Mean-Shift algorithm to search in this neighbouring range. The exhaustive search in neighbourhood of the predicted target location for the best target candidate is, however, a computationally intensive process. As a solution some authors proposed to use the Fast Mean-Shift instead of the Mean-Shift algorithm there [8, 10]. The key contribution of their work is to use the fast computation of Mean-Shift algorithm for the cases when the Kalman filter fails due to measurement errors. The Fast Mean-Shift algorithm relies on correct computation of image difference. In this work we address two problems that make fault to get the correct image difference. These problems often occur during the human bodies tracking in a pathway. In this situation the current background is often changed with respect to the reference image because of a parked car or a moving car that passes behind the objects. For the solution we propose to take GMM advantages of the image and optical flow. Simulation results have shown that the proposed method fulfilled the good tracking issue for tracking the multiple human bodies in a pathway.
The paper is organized as follows: section 2 gives a detailed description for tracking algorithm subsequence; the experimental results are set in section 3. Finally, the paper is concluded in section 4.

2 Tracking algorithm

2.1 Object segmentation

Evidently, before we start with tracking of moving objects, we need to extract the moving objects from background. We used the background subtraction to segment the moving objects.
Each background pixel is modelled using a mixture of Gaussian distributions.

$$P(X_t) = \sum_{t=1}^{k} (\omega_{i,t} * f(X_t \mid \mu_{i,t}, \sum i,t)) \qquad (1)$$

Where k is the number of Gaussians (k is a small number from 3 to 5). X_t is the current pixel value vector, which consists of red, green, blue components intensity. $\omega_{i,t}$ is an estimate of ith Gaussian weight in the mixture at time t; $\mu_{i,t}, \sum_{i,t}$ are respectively the mean value and the covariance matrix of ith Gaussian in the mixture at time t. For computational reasons the red, green and blue pixel components are assumed to be independent [3]. And f is a

Gaussian probability density function. Then we decided that the pixel belongs to the background if:

$$P(X_t) > c_{thr} \quad (2)$$

Where c_{thr} is the threshold value.

In this model weights values of all components and also means and variances of close components are updated frame by frame to track the background model. A component is close if the probability of the pixel value using that component exceed than a threshold value.

2.2 Kalman filter

Kalman filter is a powerful estimation tool in linear systems where the state is assumed to be distributed by a Gaussian model. Kalman filter consists of two sets of equations: the predict equations and update equations [9]. The predict equations are responsible for predicting the next state using the current state and the error covariance. This is often mentioned as prior estimate. The Update equations are responsible for the feedback, in other words, for incorporating a new measurement into the priori estimate to obtain an improved posterior estimate. The state variable of Kalman filter in this work is object location , its velocity and the width of rectangle represent the width of an object. The state-space representation of the tracker used in Kalman filter is given in Eq. (3)

$$\begin{bmatrix} x_t \\ y_t \\ Vx_t \\ Vy_t \\ w_t \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta_t & 0 & 0 \\ 0 & 1 & 0 & \Delta_t & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ Vx_{t-1} \\ Vy_{t-1} \\ w_{t-1} \end{bmatrix} + n_t \quad (3)$$

where, x_t and y_t are the predicted coordinates of the object and Vx_t and Vy_t are the velocities in the respective direction, w_t represents the width of the object rectangle, Δ_t represents the time interval of state correction and n_t is the white Gaussian noise with diagonal covariance matrix Q.

2.3 Improved Fast Mean-Shift

Mean-Shift algorithm is a nonparametric technique to locate density extrema or modes of a given distribution by an iterative procedure [10]. Starting from a location *x* the local Mean-Shift vector represents an offset to *x′*, which is a translation towards the nearest mode along the direction of maximum increase in the underlying density function. The local density within the local neighbourhood of a kernel is estimated by kernel density estimation where at a data point *a* kernel weights *K(a)* combine with weights *I(a)* associates with the data. Fast computation of the new location vector *x′* can be performed as:

$$x' = \frac{\sum_a K''(a-x) ii_x(x)}{\sum_a K''(a-x) ii(x)} \quad (4)$$

Where *K″* represents the second derivative of kernel *K*, differentiated with respect to each dimension of image space, i.e. the *x*- and *y*-coordinates. The functions ii_x and *ii* are the double integrals, i.e. two-dimensional integral images in the form of:

$$ii_x(x) = \sum_{x_i < x} I(x_i) x_i \quad (5)$$

and

$$ii(x) = \sum_{x_i < x} I(x_i) \quad (6)$$

Where I(x) is the difference image intensity that is normalized by mapping maximum intensity value to unit intensity and its entire range scaled proportionally. Difference Image is computed by forming the difference between the current image of an image sequence and a reference image representing the background. The problem comes from the situations when a big object passes behind the objects or the current background has changed with respect to the reference background. In our work this problem happens when a moving car passes behind the objects or the background is changed by a parked car or something else. We address this problem in two cases:

1) Current background has changed with respect to the reference image.
2) A moving car passes behind of the overlapped objects.

In case 1 the problem comes from the change of the background during the tracking process. Therefore for avoiding this problem, we proposed to use GMM of the background to construct the difference image. Let's W be the rectangle that surrounded the overlapped objects. This rectangle could be determined after segmentation level using GMM. The difference image I_d that is needed for Fast Mean-Shift algorithm is computed as:

$$I_d(x,y) = \begin{cases} 1 - P_B(I(x,y)) & (x,y) \in W \\ 0 & otherwise \end{cases} \quad (7)$$

Where P_B is approximated background model which is obtained by GMM background subtraction. This image can be used instead of the normalized difference image.

For the solution of case 2, we used the fact that two distinct objects will never overlap each other if they have equal velocities. In other word overlapping will happen if the objects have different velocities. So we can use the speed of the objects as separation criteria. For this purpose, the related optical flow is computed for each pixel that belongs to W (the rectangle that surrounded the overlapped objects). Afterwards, the computed optical flows are compared to the predicted speeds of the overlapped objects. Each point belongs to the object for which its speed is more similar to the point optical flow. Dot product is used to evaluate the similarity. After separating the car from the other objects, the remained objects can be located easily using fast Mean-Shift algorithm.

2.3.1 Optical flow

In order to calculate the optical flow for a certain point, we make use of an area around of it. We assume the frame taken with an exposure time Δ_t where the motion blur occurred for more than a couple of pixels, as it is common in video frames. Therefore in order to calculate the optical flow of the whole region we run the following described algorithm. The algorithm can be divided in two stages: first there is the extraction of the orientation of the velocity vector, and second the calculation of its magnitude. In the first stage there exists a pre-processing step, in order to have better results with the initial Fourier Transform. The methods are masking the window with a Gaussian window. Then the orientation of the velocity vector is extracted using Radon transform [11]. The second stage is where the application of Cepstrum provides us with the magnitude of the velocity vector [12]. The algorithm in Fig 1 calculates the velocity vector for an input image segment.

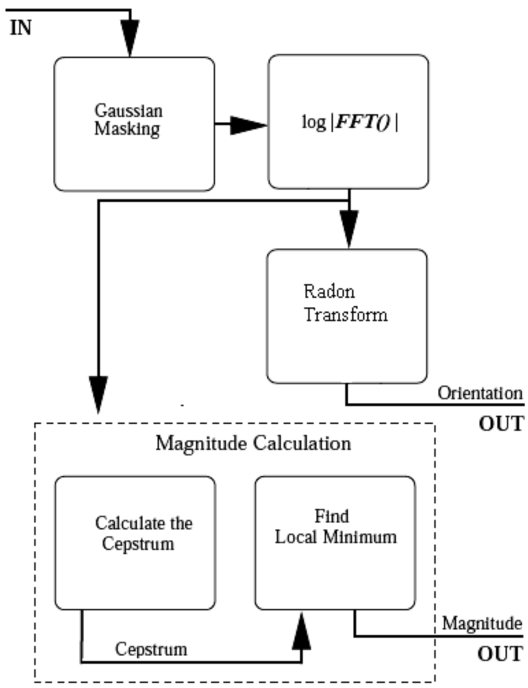

Figure 1: Block diagram of the algorithm used to extract the velocity vector.

2.3.2 Gaussian masking

The Gaussian convolution mask is important as a smoothing mask. It has optimal properties in a particular sense: it removes small-scale texture and noise as effectively as possible for a given spatial extent in the image:

$$m(x,y) = e^{(-x^2-y^2)} \tag{8}$$

Here m is the Gaussian mask where the centre of the patch is considered as the origin of the mask.

2.3.3 Radon transform

Given a function f(x, y) or more generally a measure, we define its Radon transform by [11]:

$$R(f)(x,\theta) = \int_{-\infty}^{\infty} f(x\cos\theta - y\sin\theta + x\sin\theta + y\cos\theta)dy \tag{9}$$

This corresponds to integrating f over a line in R^2 of distance x to the origin and at an angle θ to the y-axis. To implement the Radon transform, we first assume that I is a square image. The content of I is assumed to be of finite support against a black background. Let g(x, y) be the patch image, and θ a vector of t with equally spaced values from 0 to 180(1 − 1/t). For each $j = 1,..,t$ compute the discrete Radon transform for $\theta(j)$. Call this matrix R. Now determine the angle $\theta(i, j)$ for which the Radon transform assumes its greatest values. Finally, we find the five largest entries in the ith column of R, for each $j = 1,..,t$ and sum them. The result is a vector v of length t, where each entry corresponds to an angle θ. The maximum entry of v provides the estimate for θ.

2.3.4 Amplitude computation

A method for identifying the optical flow is to compute the two-dimensional Cepstrum of the blurred image g(x, y). The Cepstrum of g(x, y) is given by[11,12] :

$$\mathrm{C}(g(x,y)) = \mathrm{f}^{-1}(\log|f(g(x,y))|) \tag{10}$$

The Cepstrum of a patch image shows two significant negative peaks at a distance L from the origin. An estimate for the length of optical flow is this value L. The cepstral method for angle detection is more susceptible to noise than the Radon transform method. Hence, we improve the result for the length detection by first estimation of the angle via the Radon transform method in Section 2.3. First, we de-noise the Cepstrum of the noisy image using a Gaussian filter. Then we rotate the Cepstrum by the angle θ estimated using the Radon transform. Assuming this angle estimate is reasonable, the peaks will lie close to the horizontal axis. Any peaks outside a small strip around the x-axis are caused by noise effects; we only need to look for the peaks inside the strip. Furthermore, the peaks should appear at opposite positions from the origin. Thus we can amplify them by reflecting the image across the y-axis and then adding it. The result is that the peaks will add up and in the rest, some noise will be removed. Once we have made these corrections for noise, the estimated length of the motion blur is the distance between the negative peak and the y-axis, multiplied by an appropriate geometric correction factor.

3 EXPERIMENTAL RESULTS

In this section we show the experimental results to evaluate the proposed method. Two sequences of outdoor environment scene are used to test the assertions. Each sequence is comprised of 15 frames. The first experiment deals with the problem of case 1 that changing of the background, failed the traditional Fast Mean-Shift. In our work the background is often changed by parking a car beside the street. One frame of test sequence is shown in Fig 2. The result of tracking using the proposed method and traditional Fast Mean-Shift is set in Fig 2a and Fig 2b respectively.

(a) (b)

Figure 2: a) tracking using GMM's output as difference image.
b) Tracking using traditional Fast Mean-shift.

As can be seen, the overlapped objects are easily tracked by the improved Fast Mean-Shift algorithm. Whereas the traditional Fast Mean Shift cannot be adopted for this situation (Fig.2-b). The second experiment demonstrated the performance of the proposed method when the problem of case 2 occurred. As mentioned before this problem often happens when a moving vehicle immediately changes the background pixels. In our work the main problem comes from the situations that one or more object influenced by this background changing. In Fig.3 a sample problem and the tracking result by traditional Fast Mean-Shift and our algorithm are shown. As can be seen a vehicle has changed the background of the overlapped objects. In this case the improved Fast Mean-Shift precisely detected the true position of human, while the traditional algorithm failed as previous situation. This method increases computational complexity of the main algorithm , however it should be applied for getting a suitable results for tracking purpose in the aforementioned issues.

(a) (b)

Figure3: a) Applying Fast Mean-Shift algorithm after segmentation by optical flow. b) Tracking using traditional Fast Mean-shift

4 CONCLUSION

In this work we proposed an enhanced version of Fast Mean-Shift algorithm for multi-human object tracking. We addressed two problems in our work. The problems deal with the situation that computing of correct difference image is failed because of changing of the reference image. They often happen when a parked car or moving car passes behind the objects changes the reference background of the scene. We solved the problem of parking car by applying GMM of the background to construct the difference image. The problem of moving car is solved by segmenting the objects from the car based on the optical flow concept. Using this method we can easily separate the car from human bodies. The proposed algorithm is tested in real world data, and efficiency of the algorithm is proved by simulation results.

5 References

[1] Collins R., Lipton A., Kanade T., Fujiyoshi H., Duggins D., Tsin Y., Tolliver D., Enomoto N., and Hasegawa O.: A system for video surveillance and monitoring: VSAM final report, Carnegie Mellon University, Robotics Institute, In Tech. Rep. CMU-RI-TR-00-122000.

[2] Fuentes L. M., and Velastin S. A.: People tracking in surveillance applications, In Proceedings of 2nd IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, Kauai, Hawaii, 2001.

[3] Stauffer, C., and Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. Proc. CVPR, Fort Collins, Colorado, USA, pp. 246–252, 1999.

[4] Broida, T., and Chellappa, R. : Estimation of object motion parameters from noisy images, IEEE Trans. Patt. Analy. Mach. Intell. vol. 8, no. 1, pp. 90–99. 1986.

[5] Comaniciu, D., and Meer, P. Mean-Shift: A robust approach toward feature space analysis, IEEE Trans. Patt. Analy. Mach. Intell. vol. 24, no. 5, pp. 603–619, 2002.

[6] Pulford G. W.:Taxonomy of multiple target tracking methods, IEE Radar, Sonar and Navigation, vol. 152, no. 5, pp.291- 304, 2005.

[7] Wang L., Hu S., and Zhang X.: Detecting and tracking of small moving target under the background of sea level, 9th International Conference on Signal Processing, pp. 989- 992, 2008.

[8] Ahmed Ali, Kenji Terada*: A General Framework for multi-Human Tracking using Kalman Filter and Fast Mean-Shift Algorithms*. Journal of Universal Computer Science, vol. 16, no. 6 (2010), 921-937

[9] Bar-Shalom, Y. and Foreman, T.: Tracking and Data Association. Academic Press Inc. 1988.

[10] Beleznai C., Frühstück B., and Horst D.: Human tracking by fast Mean-Shift mode seeking, Journal of Multimedia, vol. 1, no. 1, pp. 1-8, 2006.

[11] Hui Ji; Chaoqiang Liu; Motion Blur Identification from Image Gradients. Computer Vision and Pattern Recognition, 2008. CVPR 2008 IEEE.

[12] Wenzhao Tan; Jin Zhang; Gang Rong ; Huirong Chen; Networking, Sensing and Control, 2004 IEEE.