Moving object tracking using particle filter and observational model based on multi-feature composition

N.Behzadfar
1. R&D Dep. Saadat Co.,Tehran, Iran
2. Electrical Eng. Dep. Tarbiat Modares University,Tehran, Iran
Email: behzadfar_neda@yahoo.com

M.Ansarian
R&D Dep. Saadat Co.,Tehran, Iran
Email: ansarian_545@yahoo.com

M.Sadaghiani
R&D Dep. Saadat Co.,Tehran, Iran
Email: moti.sa84@gmail.com

Abstract— **moving object tracking in a sequence of an image is one of the favorable issues in machine vision. Recently, particle filter have based developed as a powerful method in this field. Particle filter is a following method which estimates a target route in video image sequences by probability approaches. In many cases, the follower encounters with problems such as: local lighting variations or abrupt movements. In most of these cases, target tracking is missing, so an appropriate filter coupled observational model is required to improve follower performance and increase the efficiency. An observational model is used in order to improve filter performance. However, this color feature based model contains less computational volume and is more rapid but does not provide good performance at the presence of background color or some objects with similar color. In this paper, an observational model is proposed that performs using particle filter accompanied by mean shift algorithm based on incorporated color and edge features. The results show that the introduced method is not sensitive to color and intensity change while it also has a good performance.**

Keywords-tracking, particle filterr , mean shift algorithm, color , edge.

I. INTRODUCTION

Moving object tracking which is done before video analysis has many usages in machine vision. Object tracking and recognition are employed in different fields such as: surveillance and maintenance systems; statistical measurements; auto mobile following [1]. Various methods using probability approaches have been used in this field to estimate a route for a target in video image sequences. One of the most widely used is a particle filter [2-3]. Initial idea of this filter was estimating an object position in future according to previous observation and position. In present study, the state estimation and dynamics of the previous method are increased by adding some parameters to the initial idea because of non-Gaussian noises, complexity of observations and also not relying completely to accuracy of the previous stages results [5].

This filter performs based on multi-estimation instead of one; so it can select best estimation as an object state. To do so, we take some samples of previous state of an object in order to estimate new state based on new observation and previous states. Particle filter encounters with different problems like: local lighting variation and abrupt movements. In most of these cases, target tracking is missed, so we require observational model to improve performance.

Observational model in many cases is considered based on color. In one hand, the color oriented tracking technique has some advantages such as: less computational tasks, rotation and object transformation independency but in the other hand this algorithm fails when objects with same color exist at a scene or there are many variations in environmental light [6]. To overcome this problem, we decided to add edge feature as a color feature complementary. Therefore, the mean shift algorithm firstly provides a target mode by the aid of histogram weights.

Color information is extracted in the target model, and then the edge information (contains object's boundary pixels) will be added to the previous integral information by the aid of an edge identification method. This information is available for the particle filter.

In the following, we review object tracking based on mean shift algorithm, edge and color features fusion and particle filter. We used the PETS2000-2007 datasets [7] to evaluate our tracking system.

II. OVERALL REVIEW OF PROPOSED METHOD

A. involved features

In this paper, we use edge and color features which are extracted from marked region in first frame. An ellipse window is defined about the region of interest in an initial frame. Because partial changes in object appearance do not cause any changes in the object histogram, we offer features based on histogram [8].

Color features:

Boundary pixels are less important due to the interference and overlapping with background, so we assign weight to each pixel based on its distance to central pixel. In this regard, color $\hat{q}_{u_c}$ for target model is estimated by equation (1):

$$\hat{q}_{u_c} = c_c \sum_{i=1}^{n} k(\| x_i^* \|^2)\, \delta[b_c(x_i^*) - u_c] \quad (1)$$

Where $\{x_i^*\}_{i=1,2,\ldots,n}$ is position of normalized pixels in defined region as target model. Center of this region is zero. Kernel profile, k(x) assigns lower weights to pixels that are further from center. Cc Constance is defined to satisfy condition: $\sum_{u_c=1}^{m_c} \hat{q}_{u_c} = 1$ and δ is corner delta function.

The function $b_c : R^2 \rightarrow \{1,\ldots,m_c\}$, relates each $x^*{}_i$ to its histogram box. Histogram of color features $u_c = 1,\ldots,m$, for target candidates are calculated by equation (2) using equal kernel profile:

$$\hat{p}_{u_c}(y) = c_c \sum_{i=1}^{n} k(\| y - x_i \|^2)\delta[b_c(x_i) - u_c] \quad (2)$$

Edge feature:

Edges are pixels where the intensity changes abruptly. An edge in an image is usually taken to mean the boundary between two regions having relatively distinct grey levels. To extent edge feature, we change colorful image to gray scale intensity. Then gradient at point (x,y) was calculated using edge recognizing method “canny”.

A threshold value T was considered to reduce noise impact.

$$M(x,y) = \{ \begin{matrix} G(x,y), G(x,y) \geqslant T \\ 0 \qquad\qquad ow \end{matrix} \quad (3)$$

Edge histogram for target model was obtained from equation (4).

$$\hat{q}_{ue} = c_e \sum_{i=1}^{n} k(\| x_i^* \|^2)\delta[b_e(x_i^*) - u_e] \quad (4)$$

Edge histogram $u_e = 1,\ldots,m$ for target candidate using equal kernel histogram is obtained as follow:

$$\hat{p}_{ue}(y) = c_e \sum_{i=1}^{n} k(\| y - x_i \|^2)\delta[b_e(x_i) - u_e] \quad (5)$$

B. mean shift algorithm

Full searching in order to object position tracking is a time-consuming work, so here a recursive algorithm was used. For the autonomous vehicle application, the task is to first define an object of interest, by segmentation and/or by interactive selection, then by tracking the object as it moves within the camera field of view. Transition from current position to new one was performed using $\hat{y}_1$ vector.

$$\hat{y}_1 = \frac{\sum_{i=1}^{n} x_i w_i\, g(\| y - x_i \|^2)}{\sum_{i=1}^{n} w_i\, g(\| y - x_i \|^2)} \quad (6)$$

Figure 1. Left image: original image, right: edge extracted from image.

Where $\hat{y_1}$ is current coordinate of ellipse and is considered as target. We use equation (7) to obtain a relationship that determines impact of edge and color features at final position of target in current frame.

$$y_f = \alpha_c \hat{y}_{1c} + \alpha_e \hat{y}_{1e} \quad (7)$$

$\hat{y}_{1c}, \hat{y}_{1e}$ are target positions in current frame based on color and edge feature represent. α_c, α_e also are coefficients of color and edge which determines impact of each feature at the final position and obtained as follows:

$$\alpha_c = \frac{\rho[\hat{p}_{u_c}(\hat{y}_{1c}), \hat{q}_{u_c}]}{\rho[\hat{p}_{u_c}(\hat{y}_{1c}), \hat{q}_{u_c}] + \rho[\hat{p}_{ue}(\hat{y}_{1e}), \hat{q}_{u_e}]} \quad (8)$$

$$\alpha e = \frac{\rho[\hat{p}_{ue}(\hat{y}_{1e}), \hat{q}_{ue}]}{\rho[\hat{p}_{u_c}(\hat{y}_{1c}), \hat{q}_{u_c}] + \rho[\hat{p}_{ue}(\hat{y}_{1e}), \hat{q}_{u_e}]} \quad (9)$$

C. Particle filter

The particle filter is a bayesian sequential importance sampling technique, which recursively approximates the posterior distribution using a finite set of weighted samples. It consists of essentially two steps: prediction and update. Assume that set $\{x_{0:k}^i, w_k^i\}_{i=1}^{N_s}$ is subsequent density function of $p(x_{0:k} | z_{1:k})$ in which $\{x_{0:k}^i i = 0,\ldots,N_s\}$ is set of points with weights $\{w_{0:k}^i i = 0,\ldots,N_s\}$ and $x_{0:k} = \{x_j, j = 0,\ldots,N_s\}$ is consist of all states until k time. Weights are normalized such that $\sum_i w_k^i = 1$, then we estimate subsequent density at time k as equation (10).

$$p(x_{0:k} | z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_{0:k} - x_{0:k}^i) \quad (10)$$

Weights are chosen based on "importance sampling" principle [9]. If we extract $x^{i}_{0:k}$ samples from importance density, $q(x_{0:k} \mid z_{1:k})$ weights in equation (10) are explained by equation (11).

$$w^{i}_{k} \alpha \frac{p(x^{i}_{0:k} \mid z_{1:k})}{q(x^{i}_{0:k} \mid z_{1:k})} \tag{11}$$

Samples can represent an approximate of $p(x_{0:\ k\pm1} \mid z_{1:\ k-1})$. Target is obtaining $p(x_{0:\ k} \mid z_{1:\ k})$ using new group of samples. If we formulate importance density, then:

$$q(x_{0:\ k} \mid z_{1:\ k}) = q(x_{k} \mid x_{0:\ k\ \ 1}, z_{1:\ k}) q(x_{0:k-1} \mid z_{1:\ k-1}) \tag{12}$$

We can obtain $x^{i}_{0:\ k} \propto q(x_{0:\ k} \mid z_{1:\ k})$ samples by completing each of current samples $x^{i}_{0:\ k\ \ 1} \propto q(x_{0:\ k\ \ 1} \mid z_{1:\ k-1})$ with newest $x^{i}_{k} \propto q(x_{k} \mid x_{0:k-1}, z_{1:\ k})$.

$p(x_{0:\ k} \mid z_{1:\ k})$ is declared by from $p(x_{k} \mid x_{k-1}), p(z_{k} \mid x_{k}), p(x_{0:k-1} \mid z_{1:\ k-1})$ in order to calculate updating weight equation.

III. SIMULATION RESULTS

We implemented the proposed algorithm on some images in MATLAB. Ellipse shows target region is tracked by this algorithm.

Figure 2. Location of tracked points in first 10 frames.

There may be overlapping in some images. To investigate algorithm accuracy, we applied it on these images.

Figure 3. Location of tracked points.

IV. DISCUSSITON

The mean shift algorithm accompanied by edge and color features had been used in previous study [1]. In this paper, tracking accuracy is increased using particle filter. Results of two algorithms are compared in figure 4. Also we compare an average distance of target point with tracked point in both proposed algorithm and previous study [1]. Results are listed in table I.

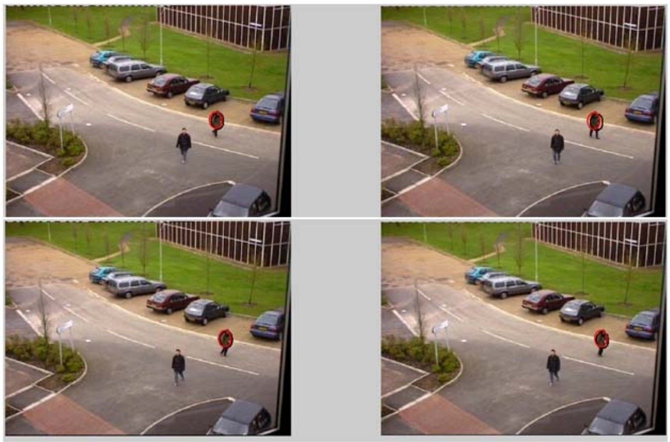

Figure 4. Four frame image, red: algorithm is [1], black: proposed method in this paper.

TABLE I. COMPARISON OF EMPLOYING ALGORITHM IN [1] AND PROPOSED METHOD.

Average distance of target point to tracked point in two method	
Algorithm in [1]	7.3252
Proposed algorithm	3.4402

V. CONCLUSION

In present paper, we tracked moving objects in a sequence of image using particle filter and an observational model based on edge and color feature fusion.

In this method, the edge feature was used to improve system performance in various lighting environment and at the presence of similar colors. The results state that the algorithm is able to track moving objects accurately.

REFRENCES

[1] M. Janipoor, R. Niaraki, “Moving onject tracking based on mean shift algorithm and features fusion”, Artifical intelligence and signal processing, pp.48-53, 2011.

[2] D. Beymer, P. McLauchlan, B. Coifman, and J. Malik, "A realtime computer vision system for measuring tracking parameters," in Computer Vision and Pattern Recognition, pp. 495-501, 1997.

[3] M. Grei_enhagen, V. Ramesh, D. Comaniciu, and H. Niemann, "Statistical modeling and performance characterization of a real-time dual camera surveillance system," in Computer Vision and Pattern Recognition, vol. 2, pp. 335-342, 2000.

[4] M. Isard and A. Blake, “CONDENSATION: Conditional density propagation for visual tracking,” International Journal of Computer Vision, vol. 29, pp.5-28, 1998.

[5] N.J. Gordon, D.J. Salmond, A.F.M. Smith, “Novel approach to nonlinear/non-Gaussian,” IEE PROCEEDINGS-F, Vol. 140, No. 2,1993.

[6] D.Ross J. Lim, and M. Yang, “Adaptive probabilistic visual tracking with incremental subspace update”, in Proceeding of the Eighth european conference on computer vision, pp.215-227, 2004.

[7] www. anc.ed.ac.uk/demos/tracker/pets2000.html.

[8] H. Tang, Y. Shi, J. Xia, and H. Yin, “A fast and accurate boundary tracking of moving objects in video”, IEEE international conference on information and automation, pp. 681-685, 2008.

[9] D. Comaniciu, “Kernel-Bsed object tracking”, IEEE transaction on pattern analysis and machine intelligence, vol. 25, no.5, pp. 564-577, 2003.