

# Moving vehicle target tracking based on improved Mean shift algorithm

Fei Lei<sup>1, a</sup>, Xiaoqiong Meng<sup>2, b</sup>

<sup>1</sup>Electronic Information & Control Engineering College, Beijing University of Technology, Beijing, 100124, China

<sup>2</sup> Electronic Information & Control Engineering College, Beijing University of Technology, Beijing, 100124, China

<sup>a</sup> leife@bjut.edu.cn, <sup>b</sup>13126705903@163.com

**Keywords:** Bhattacharyya distance; Mean shift; Particle filter; Target tracking

**Abstract.** This article proposes a tracking algorithm based on mean shift and particle filter according to the characteristics of moving vehicles. Firstly, the algorithms establish auto target motion model that the core is RGB color histogram. Secondly, using the Bhattacharyya distance measures similarity of particle region and the target model, and updating the weights of particles according to the similarity. Lastly, the measure improves the estimated position of the particle by using mean shift algorithm, which makes the candidate regions of these particles can be more close to the true target position. Experimental results show that the algorithm has good real-time performance and robustness, and can achieve stable tracking of target vehicle.

## Introduction

Moving target tracking technology<sup>[1]</sup> is widely used in many fields such as military, industrial production, daily life and so on, which through the monitoring of the camera to capture video images, and then real-time or subsequent processing, to the traffic management personnel even to deal with the traffic problems has brought convenience.

This paper mainly studies the tracking of the vehicle, which directly affects the reasonable treatment of the traffic problems. The tracking algorithms can be divided into two parts: one is the traditional mean shift tracking algorithm has been widespread attention and extensive research because of the characteristics of the without parameters, small amount of calculation, deformation of the target, rotation changes adaptability. However, the algorithm is not satisfactory for small target tracking, and the iterative process is easy to fall into local optimal solution<sup>[2]</sup>. Second, the introduction of the particle filter algorithm and particle filter can handle because of background clutter, occlusion caused by nonlinear and multimodal processing, can achieve parallel tracking. On the other hand, the tracking robustness is strong, but the complexity of the algorithm is higher, it is difficult to meet the multi target real time tracking. By reference to previous research results, the mean shift algorithm and particle filter algorithm combined method make full use of mean shift algorithm of real-time and particle filter algorithm's robustness<sup>[5]</sup>. This measure greatly improved the tracking effect.

## Kernel density estimation and Mean shift Algorithm

### A Kernel density estimation

Kernel function estimation is a kind of nonparametric density estimation to characterize the target, and the kernel density estimation is defined as a set of data points on a set of N data points about  $S = \{x_i\}_{i=1, \dots, n}$ , Assuming its unknown probability density function is  $f(x)$ , and the kernel function is  $K_h(x)$ . The density of the  $x$  points can be calculated as follows:

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - x_i) \quad (1)$$

Where  $x$  presents the central point of the kernel function, and the graph is symmetric with

respect to the point  $x$ . In one dimensional space, the kernel function  $K_h(x)$  should satisfy the following conditions:

$$K_h(x) = K_h(-x), K_h(x) > 0 \quad \int K_h(x)dx = 1, \quad K_h(x) = \frac{1}{h} K\left(\frac{x}{h}\right).$$

Among them,  $h$  is the bandwidth of the kernel function.

By formula (1) the estimate of the kernel function can be understood as the weighted average of the relative offset at each sample point. Usually we need to select multivariate kernel function in multidimensional space, which can be obtained the radial kernel function by rotating the single variable kernel function in  $R_d$  space. Assuming that the distribution of the unknown probability density generate the set of  $x_i (i=1, \dots, n)$ . All of them are sample points. The count is  $N$  and the dimension is  $d$ , and  $x_i \in R^d$ . So the point of  $x$ 's kernel density estimation function in space can be showed:

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^n K_H(x - x_i) \quad (2)$$

Where  $K_H$  presents kernel function that can be expressed as:

$$K_H(x) = |H|^{-\frac{1}{2}} K\left(|H|^{-\frac{1}{2}} x\right) \quad (3)$$

In the mean shift algorithm, a single parameters for the  $h$  of the bandwidth matrix usually be used, type (2) is changed as:

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right) \quad (4)$$

To sum up, as long as the sample is sufficient, the distribution of the characteristics form no need to assume, and it does not need to set the parameters of the model. The kernel function is estimated to be as close as an unknown probability density distribution function. In the algorithm, the probability density of the eigenvalues is often expressed as formula (5):

$$\hat{f}_{h,k}(x) = \frac{C_{k,d}}{nh^d} \sum_{i=1}^n k\left(\left\|\frac{x - x_i}{h}\right\|^2\right) \quad (5)$$

Where  $C_{k,d}$  presents normalized coefficient.

## B meanshift tracking algorithm

In this paper, we use the Mean shift algorithm to establish a histogram model based on the RGB color distribution for the target region and the candidate target region of interest. In the first, the target is detected by the target detection or by hand, and the histogram of the target in the feature space is calculated; Then, the histogram of the current frame is calculated by the same method; After then the similarity between the target and the candidate target is measured by Bhattacharyya coefficient, and the weight coefficient and the iteration are updated; Finally find the optimal target location. Specific algorithm steps are as follows:

a) In the initial frame image, the target area is selected manually, and a reference object model is established based on the RGB kernel histogram in the region:  $\{\hat{q}_u\}_{u=1, \dots, M}$ ;

b) The target center in the last frame is  $y_0$ , The candidate target model  $\{\hat{p}_u(y_0)\}_{u=1, \dots, M}$  and the similarity function  $\rho(\hat{p}(y_0), \hat{q}) = \sum_{u=1}^M \sqrt{\hat{p}_u(y_0) \hat{q}_u}$  should be calculated;

c) Calculate the weight coefficient:  $w_i$ ;

d) The new location of the target in the current frame  $y_1$ , that can be showed as :

$$y_1 = \frac{\sum_{i=1}^{N_h} x_i w_i g\left(\left\|\frac{y-x_i}{h}\right\|^2\right)}{\sum_{i=1}^{N_h} w_i g\left(\left\|\frac{y-x_i}{h}\right\|^2\right)} \quad (6)$$

e) Update candidate target model as  $\{\hat{p}_u(y_1)\}_{u=1,\dots,M}$ , and calculate the similarity function that is

$$\rho(\hat{p}(y_1), \hat{q}) = \sum_{u=1}^M \sqrt{\hat{p}_u(y_1) \hat{q}_u}$$

f) If there is the condition of  $\rho(\hat{p}(y_1), \hat{q}) > \rho(\hat{p}(y_0), \hat{q})$ , the candidate target model should be updated. At this time, order  $y_1 = 0.5(y_1 + y_0)$ , and go to step e).

g) If there is the condition of  $\|y_1 - y_0\| < \varepsilon$ , the iteration should end, and naturally the tracking stop; If not, make the formula of  $y_0 = y_1$ , return to step b) to calculate.

As an efficient optimization algorithm based gradient, Mean shift has a faster convergence rate and higher computational efficiency. But it is easy to fall into local optimal in the context of background interference and occlusion, which can't reach the global optimum. And when the target motion property changes, it may cause the target to lose, failure and so on, which can't recover. So the effectiveness of the algorithm is at the expense of the cost of the robustness.

## Particle Filter Algorithm

Particle filter is a kind of approximate Bayesian filtering method based on Monte Carlo method, which provides an effective method to solve the target tracking in non-Gauss and non-linear systems. Particle filtering method by looking for a group of spread in state space random samples, and the probability density function is approximated. In order to achieve the optimal Bayesian estimation, the posterior probability density should be approximated.

From the perspective of Bayesian theory, tracking problem is from a series of observation data named  $z_{1:k}$  to reason the state of  $x_k$  in the moment of k. That is to construct a posteriori probability density function, which is  $p(x_{0:k} | z_{1:k})$ , and the final estimate is  $p(x_k | z_{1:k})$ . The model and observation model are as follows:

$$\begin{aligned} x_k &= f_k(x_{k-1}, w_k) \\ z_k &= h_k(x_k, n_k) \end{aligned} \quad (7)$$

Where  $X_k$  presents the system state variable,  $W_k$  presents system noises, and  $Z_k$  stands for system observation value.  $n_k$  is observation noise. The probability density distribution of the states can be expressed as follows:

$$\hat{p}(x_{0:k} | z_{1:k}) = \frac{1}{N} \sum_{i=1}^N \delta_{x_{0:k}}^i(dx_{0:k}) \quad (8)$$

Where  $\{x_{0:k}^i, w_k^i\}_{i=1}^N$  is the set of particle to express System posterior probability density function named as  $p(x_{0:k} | z_{1:k})$ .

In practice, in addition to a small number of systems to meet Gauss, uniform distribution. Usually we need to use some of the sampling algorithm to the majority of the non - standard function of the sampling. In practice, the sequential importance sampling (SIS) algorithm is one of the multiple sampling methods to realize recursive estimation.

### A sequential importance sampling

In SIS algorithm, the probability distribution is the same as that of  $p(x_k | z_{1:k})$ . Probability

density  $q(x_{0:k} | z_{1:k})$  is known and easy to sample from the distribution function. There we need to change formula of  $q(x_{0:k} | z_{1:k})$  as:

$$q(x_{0:k} | z_{1:k}) = q(x_0) \prod_{j=1}^k q(x_j | x_{0:j-1}, z_{1:j}) \quad (9)$$

Probability density function showed as  $p(x_{k-1} | z_{1:k-1})$  of the first moment have sample points of  $N$ , which can be set to  $\{x_{k-1}^i\}_{i=1}^N$ . That function can be expressed as:

$$p(x_{k-1} | z_{1:k-1}) = \sum_{i=1}^N w_{k-1}^i \delta(x_{k-1} - x_{k-1}^i) \quad (10)$$

Update Probability density function in the  $k$  moment, which can be showed as:

$$p(x_k | z_{1:k}) \approx \sum_{i=1}^N w_k^i \delta(x_k - x_k^i) \quad (11)$$

Where modified weight can be calculated as:

$$w_k^i = w_{k-1}^i \frac{p(z_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, z_k)} \quad (12)$$

Which weight normalization is:

$$w_k^i = w_k^i / \sum_{j=1}^N w_k^j \quad (13)$$

## B Resample

There is a common problem in SIS calculation that is the particle degeneracy phenomenon. In a continuous iterative computing process, the weights of most of the particles will become very small. In order to ensure the total number of particles, the maximum of the particles are assigned to the discarded particles, which complement the new particle set of  $\{x_k^{(i)}, w_k^{(i)}\}_{i=1}^N$ , so that the total number of samples is kept constant..

The above method is to resampling, resampling is an effective solution to the problem of particle degradation method. In this paper, the process is:

- 1) The random numbers in  $\{\mu_l; l=1, 2, \dots, n\}$  among  $[0, 1]$ , are generated. There needs the count of  $n$ .
- 2) The integer  $m$  is found by the search algorithm, which satisfies the following conditions:

$$\sum_{j=0}^{m-1} w_j < \mu_l < \sum_{j=0}^m w_j$$

- 3) Record sample of  $x_k^{(m)}$  as a sample of the new sample concentration.

- 4) Make the interval  $[0, 1]$  by type (2) into small cells about counts of  $n$ , when random number named  $\mu_l$  fall on a  $m$  interval that is  $I_m = (\lambda_{m-1}, \lambda]$ , so the sample of  $x_k^{(m)}$  will be copied. Where

$$\lambda_i \text{ can be showed as: } \lambda_i = \sum_{j=0}^i w_j \quad (i=1, 2, \dots, n)$$

Particle filter algorithm can estimate the state of the posterior probability density function. In order to capture the change of the state space, a certain number of particles are required to ensure the validity of the sampling, So the number of particles increases with the increase of the number of States, which leads to the increase of the amount of particles, which makes it difficult to realize the real-time tracking of particle filter.”

## Combing Mean Shift and Particle Filter Algorithm

Based on the advantages of Mean shift algorithm and particle filter algorithm, this paper combines the two algorithms. Using color feature of particle filter to track, and using mean shift algorithm of particle clustering analysis in the process of tracking, which moves the particles to the local advantages, and optimized to reduce the tracking a single target in the number of particles. Therefore in updating particles, each particle are as much as possible get the larger weights, which only fewer particles will be able to fully describe the target state. The concrete steps are as follows:

### 1) Initialization:

In a frame of a moving vehicle video image, the target tracking area is selected manually, and the number of particles is initialized. A reference object model based on RGB color histogram is established, and the particle sets which are  $\{x_0^i\}_{i=1}^N$  generated, and the weights are  $N^{-1}$ .

### 2) Prediction:

Updating the particle state with the state transition equation, which formula is  $X_k = AX_{k-1} + BW_k$ . Using the particle set of k-1 moments to predict the state of each particle in K time, and get the new particle position that is  $\{\tilde{x}_k^i\}_{i=1}^N$ . Where A presents unit matrix,  $W_k$  is system noise, and  $B = [b_1, b_2]^T$  stands for particle propagation radius. By the way,  $b_1, b_2$  are constant.

### 3) Mean shift particle optimization:

In the neighborhood of the propagation radius, an iterative optimization is carried out according to the formula (6), so that the maximum value of the particle is converged to the maximum of the neighborhood. Finally we will get the new particles that state is the set of  $X_k$  named  $\{x_k^i\}_{i=1}^N$ .

### 4) Observation:

The probability density function of the observed values is obtained by the Bhattacharyya distance between the candidate regions of each particle and the reference target model. Which can be showed as:

$$p(z_k | x_k^i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{d^2(\hat{p}_u(y), \hat{q}_u)}{2\sigma^2}\right\}$$

Where  $\sigma$  presents mean square deviation. The smaller the Bhattacharyya distance is, the greater the similarity will get, and the weight of the particle will larger. Therefore, the weights of particles in the K moment can be obtained by formula (12), and the normalized weight is obtained by formula (13).

### 5) Output target information:

The estimation of target position can be presented as :

$$p(x_k | z_k) \approx \sum_{i=1}^N w_k^i \cdot x_k^i$$

### 6) Update template:

If the current position of the histogram and the current template Bhattacharyya distance is bigger than a certain threshold value, which need to update the target template according to the type (14). The formula is:

$$p_{new} = (1 - \alpha) \times p + \alpha \times q, \quad 0 < \alpha < 1 \quad (14)$$

### 7) Resampling:

According to the importance weight, the weight of particles that counts are N is sampled from the set of  $\{x_k^i\}_{i=1}^N$ , and the weight value is  $N^{-1}$ .

### 8) Make equation about $k=k+1$ , repeat step 2) to step 7).

The algorithm flow chart is shown in figure 1:

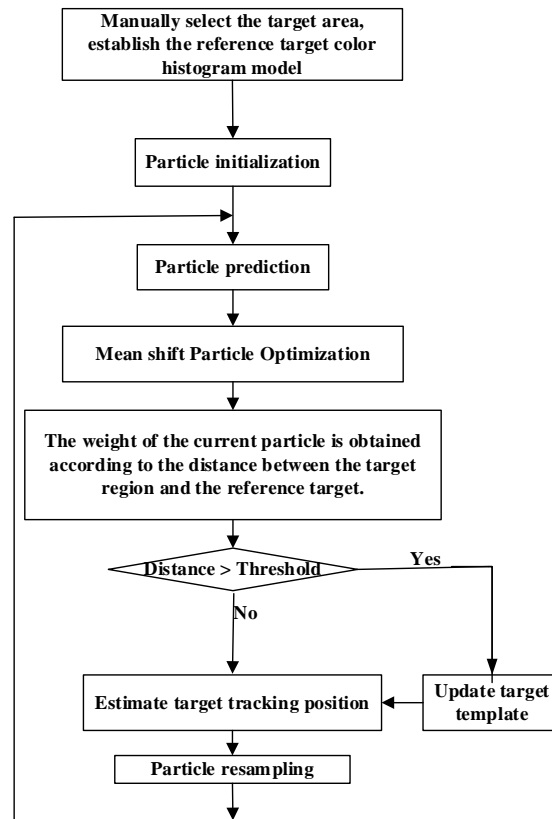


Figure 1: Moving vehicle target tracking algorithm flow chart

## Test results

In the experiment, the video of the moving vehicle is tracked by the method of this paper, and the tracking results are compared with the mean shift algorithm. The effectiveness of the proposed algorithm is verified in this paper. Manually select the moving target block diagram, a total of 351 frame images of the video sequence, the picture size is 350 x 240, and color depth is 24 bits.

Figure 2 is the result of a tracking algorithm using mean shift algorithm. As can be seen from Figure 2, from the beginning of the 30 frames the moving target's speed get faster, which results to tracking results began to fail .The target completely make failure in the fiftieth frame.

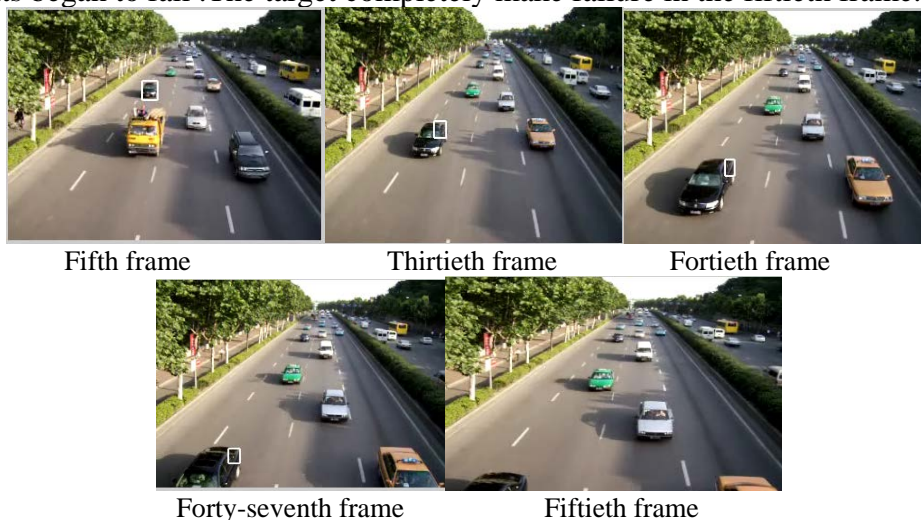


Figure 2: Results of target tracking using mean shift algorithm

Figure 3 is the tracking results of the algorithm. In early tracking, the 5 frame tracking results showed no obvious advantages compared to Figure 2; Over time, the target can still be accurately tracked in the same frames of Figure 2. The effect of target tracking is not affected by the interference so that the algorithm of this paper has a good tracking effect.

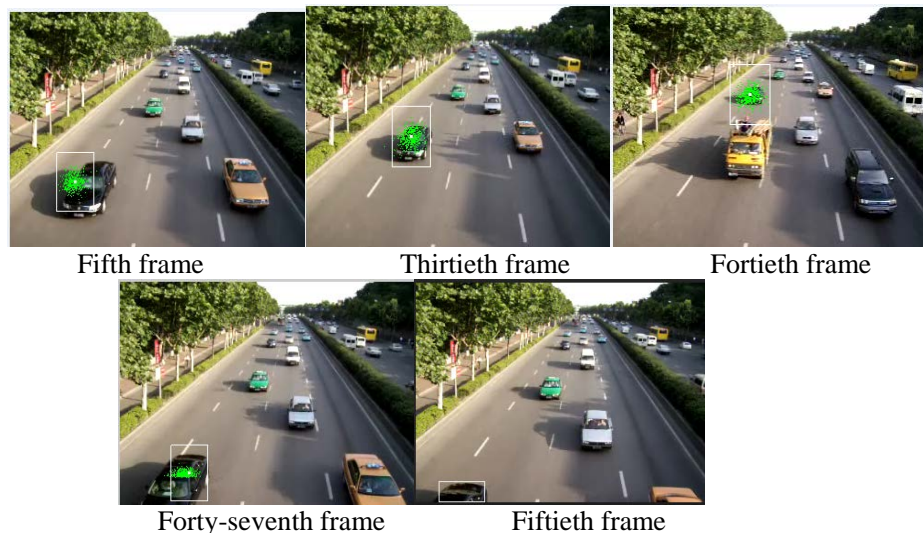


Figure 3: mean shift and particle filter combined algorithm results

## Conclusion

From the above experimental results, we can see that the proposed algorithm in this paper has significantly improved compared with the mean shift algorithm in tracking target accuracy. The tracking effect is more obvious especially after a brief occlusion, which reflects the strong adaptability.

## Acknowledgement

This paper realized the car moving target tracking algorithm. The improved mean shift algorithm takes full advantage of the mean shift algorithm's real-time and particle filter algorithm's robustness by combining particle filter and mean shift, which make the tracking effect be greatly improved. The HOG<sup>[5]-[7]</sup> feature representation and the Bhattacharyya coefficient similarity measure of the multi target tracking algorithm can be used in the future research; In addition, this paper is only in the ordinary PC machine to do the research. In order to apply to the embedded system, it should be considered that algorithm be ported to FPGA and DSP<sup>[4]</sup> to achieve target tracking.

## References

- [1] Cheng Y. Mean shift, mode seeking and clustering [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, 17(8): 790-799
- [2] Shan C, Wei Y, Tan T, et al. Real time hand tracking by combining particle filtering and mean shift [C] // International Conference on Automatic Face and Gesture Recognition. 2004: 669-674
- [3] WU You fu, SHEN Jun, DAI Mo. Traffic Object Detections and Its Action Analysis [J] Pattern Recognition Letters, 2005, 26(13): 1963-1984.
- [4] Isard M, Blake A. Condensation-conditional density propagation for visual tracking [J]. Int. J. Computer Vision, 1998, 29(1): 5-28. 1032-1063.
- [5] Comaniciu D, Meer P. Mean shift: A robust approach toward feature space analysis [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2002, 24(5): 603-619.
- [6] Dalal N, Triggs B. Histograms of oriented gradients for human detection [C] // Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2005: 886-893.
- [7] Lei fei, Zhao Xiaoxia. Adaptive Background Estimation of Underwater Using Kalman-Filtering [C]. The 3rd International Congress on Image and Signal Processing. 16-18 Oct. 2010: 64-67.