Eurasia Journal of Science and Technology	ISSN: 2663-1024	Volume 4, Issue 1, pp 29-34	APR 2022

RESEARCH ON TRAFFIC TARGET TRACKING METHOD IN COMPLEX ENVIRONMENT BASED ON MACHINE VISION

Likai Wang[1], Weiwei Liu[1], Dan Li[2*]

[1]Traffic police detachment of Xuzhou Public Security Bureau
Xuzhou, China
[2]Xuzhou University of Technology
Xuzhou, China
[*]Corresponding author: lidanonline@163.com

Abstract
At present, with the development of society, road vehicles are increasing, and the highway carrying capacity is relatively insufficient, resulting in serious traffic congestion and frequent accidents. How to select effective tracking methods and extract invariant features needs to be improved. Aiming at some problems existing in urban road vehicle tracking, this paper proposes a vehicle tracking algorithm in complex environment based on continuous adaptive Meanshift algorithm and integrating local invariant features. The improved algorithm can adaptively update different feature weights when dealing with vehicle deformation, frost and fog weather, background noise interference, light variation, occlusion and other problems. The complementarity between features is insufficient, and has good robustness to complex environment.
Keywords: traffic monitoring; target tracking; Camshift; feature fusion.

1. INTRODUCTION

Vehicle tracking technology in urban road traffic monitoring system[1] can monitor and record vehicle behavior information in real time and obtain vehicle information parameters, which further lays the foundation for vehicle behavior recognition. At present, most tracking methods rely on single color information to represent the tracked target. Kalman filter tracking[2,3] has low computational complexity and can track the center of the vehicle, but the vehicle running state is required to meet the Gaussian distribution, and it is easy to fail to track most nonlinear and non Gaussian environments in reality. The block matching tracking algorithm uses the matching degree between images for tracking, but it is unable to set a reasonable threshold through the allocation strategy in the tracking process, and its anti affine transformation ability is poor. Particle filter algorithm[4,5] has strong anti-interference ability, but there is mutual exclusion of accuracy and time consumption, and particles are prone to degradation in the tracking process. Continuous adaptive Meanshift algorithm is a fast pattern matching algorithm based on kernel density without parameters. It can effectively solve the problem of target deformation. Its calculation is simple and time-consuming, but it is easy to lead to target tracking failure when the target color is close to the background, occlusion and illumination change. In short, the above methods are not universal. For example, it is easy to lose the target when there are lighting, vehicle or deformation, frost and fog weather, unclear target edge or area characteristics, occlusion and other factors in the scene.

In view of the above situation, how to select effective tracking methods and extract invariant features need to be improved. Aiming at some problems existing in urban road vehicle tracking, this paper proposes a vehicle tracking algorithm in complex environment based on continuous adaptive Meanshift algorithm and local invariant features. By improving the ability of identifying vehicle targets, it provides a strong theoretical basis for intelligent road traffic monitoring system.

2. CAMSHIFT ALGORITHM AND PROBLEMS

Camshift algorithm is based on Meanshift algorithm[6,7] and uses the distribution characteristics of image color probability density to track the target. Because the kernel function window width of Meanshift algorithm is fixed, the tracking effect is affected. When the target scale changes significantly, the positioning will be inaccurate. Camshift uses the pixel value and the second-order moment of the search window to estimate the target size and direction angle, so as to adaptively adjust the window size and angle. In addition, Camshift selects HSV color model as tracking feature to reduce the influence of illumination. The algorithm is described as follows:

Select the search window with the initial size of s and the position center of (x_c, y_c), with the internal pixel of $\{x_i, i=1,2,...n\}$, the window center of y and the width of h. Calculate the HSV color probability distribution within the range of 1.1 times of the search window with (x_c, y_c) as the center. If there are m-level color features, the normalized color histogram is:

$$q_u(y) = C_h \sum_{i=1}^{n} \delta[b(x_i) - u],\ u = 1,2,...m \tag{1}$$

The target color distribution is q_u , $\sum_{u=1}^{m} q_u = 1$. The pulse function is δ and the normalization coefficient is C_h . Estimation of color probability distribution using kernel function:

$$\hat{f}_k(x) = \frac{1}{n \cdot h^2} \sum_{i=1}^{n} k\left(\left\| \frac{y - x_i}{h} \right\|^2\right) \tag{2}$$

$$\hat{p}_u(y) = C_h \sum_{i=1}^{n} k\left(\left\| \frac{y - x_i}{h} \right\|^2\right) \delta(b(x_i) - u) \tag{3}$$

The color distribution of the target in frame i is $\hat{\mathrm{p}}_{\mathrm{u}}\left(y_i\right)$. The similarity of color distribution is measured by $\hat{\rho}\left(y\right)$ Bhattacharrya coefficient.

$$\hat{\rho}(y_{i+1}) = \sum_{u=1}^{m} \sqrt{\hat{\mathrm{p}}_{\mathrm{u}}(y_{i+1}) \mathrm{q}_u}$$

$$\approx \frac{1}{2} \sum_{u=1}^{m} \sqrt{\hat{\mathrm{p}}_{\mathrm{u}}(y_i) \mathrm{q}_u} + \frac{1}{2} \sum_{u=1}^{m} \hat{\mathrm{p}}_{\mathrm{u}}(y_{i+1}) \sqrt{\frac{\mathrm{q}_u}{\hat{\mathrm{p}}_{\mathrm{u}}(y_i)}} \tag{4}$$

$$w(x_i) = \sum_{u=1}^{m} \delta(b(x_i) - u) \sqrt{\frac{\mathrm{q}_u}{\hat{\mathrm{p}}_{\mathrm{u}}(y_i)}} \tag{5}$$

In the $i+1$ frame, find the new target center y_{i+1} to make q_{u} and $\hat{\mathrm{p}}_{\mathrm{u}}(y_{i+1})$ most similar. q_u $\hat{\mathrm{p}}_{\mathrm{u}}(y_i)$ and weight are known, and the similarity $\sum_{u=1}^{m} \hat{\mathrm{p}}_{\mathrm{u}}(y_{i+1}) w(x_i) = C_h \sum_{i=1}^{n} k\left(\frac{\| y_{i+1} - x_i \|^2}{h^2}\right) w(x_i)$ is maximized, that is, the probability density estimation of the search window at center y_{i+1} . The algorithm adopts Epanechnikov kernel function, and y_{i+1} calculates the centroid through continuous iteration. The vector formula is as follows:

$$M_h(x) = \frac{\sum_{i=1}^{n} G\left(\frac{x_i - x}{h}\right) w(x_i) x_i}{\sum_{i=1}^{n} G\left(\frac{x_i - x}{h}\right) w(x_i)} - x \tag{6}$$

In order to adapt the direction and size of the moving target, the pixel value and the second-order moment of the search window are used to estimate. The algorithm reinitializes the position and size of the new search window as the initial position of the next frame, continues to calculate the HSV color probability distribution within 1.1 times of the search window until convergence, and repeatedly executes the above algorithm to track the moving target. Camshift algorithm has the advantages of simple calculation, fast matching speed and high real-time performance. It can automatically adapt to the changes of target size and direction angle. However, because it only uses image color features, it is easy to be affected by color similar distractors in complex environment, resulting in tracking failure. As shown in figure 1, 87, 110, 118 and 122 frames of the video are intercepted for testing. The tracked vehicle is blocked by the bridge pillar during movement, and the body color is very close to the bridge pillar color, forming a large interference. It can be seen that Camshift algorithm only uses color probability, which is easy to be attracted by the bridge column and lose the target.

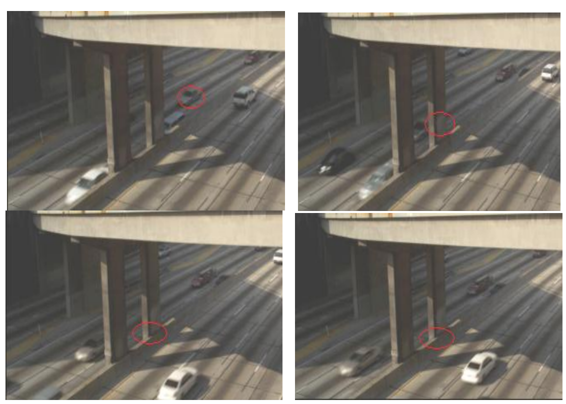


Figure 1 Camshift target tracking

3. LOCAL INVARIANT FEATURE EXTRACTION AND OPTIMIZATION

A. Feature extraction

SIFT (scale invariant feature transform) scale invariant feature transform algorithm [8,9] extracts the local features of the image, which can adapt to various affine transformations, and has good invariability to occlusion, brightness change, noise, scaling and translation. Its fusion with Camshift algorithm can better increase the tracking stability. SIFT algorithm uses Gaussian difference multi-scale transformation to find key points, and realizes feature matching through the invariance of image direction feature vector. The process is described as follows:

Firstly, the Gaussian difference scale space is established. The original image I(x,y) is convoluted with Gaussian function G(x,y,σ) at different scales. Get L(x,y,σ).

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{7}$$

The Gaussian difference scale space formula is obtained by subtracting the images located in adjacent scales:

$$
\begin{aligned}
D(x,y,\sigma) &= (G(x,y,k\sigma)-G(x,y,\sigma)) * I(x,y) \\
&= L(x,y,k\sigma)-L(x,y,\sigma) \\
&\approx (k-1)\sigma^2\nabla^2 G
\end{aligned} \tag{8}
$$

$$
G(x,y,\sigma)=\frac{1}{2\pi\sigma^2}e^{-(x^2+y^2)/2\sigma^2} \tag{9}
$$

Then, the extreme points of the difference pyramid are detected. Hessian matrix is used to eliminate the strong edge response of edge pixels due to DOG operator.

The main direction of the key points is calculated. In order to make the descriptor invariant during rotation, the local feature is used to assign the reference direction to the feature points. In the following formula, the direction and modulus at the sampling point are represented by θ(x, y) and m(x, y).

$$
m(x,y)=\sqrt{(L(x+1,y)-L(x-1,y))^2-(L(x,y+1)-L(x,y-1))^2} \tag{10}
$$

$$
\theta(x,y)=\alpha\tan2((L(x,y+1)-L(x,y-1))/L(x+1,y)-L(x-1,y)) \tag{11}
$$

Finally, the key points are described and the feature vector is generated. In the neighborhood space of key points, in 4×4 size window, the gradient value in the 8-direction of each sub window is calculated. Therefore, each key point uses 128 dimensions, 4×4×8 dimensional vector.

B. Eigenvector optimization

Although SIFT algorithm is anti affine transformation, because the key point vector is up to 128 dimensions and the time complexity is very high, it is not suitable to directly apply SIFT algorithm to the traffic environment that needs real-time processing. On the premise of ensuring the quantity and accuracy of the generated key points, this paper introduces the sample principal component analysis(PCA) method[10,11] in statistical analysis to reduce the dimension of the key point features, so as to improve the real-time performance of the system. The specific process is as follows:

Set the number of key points as *n*, and use *p* eigenvectors to describe a key point information to form a sample matrix. Let a sample of capacity *n* of $X=(X_1,X_2,...,X_n)^T$ be $x_i=(x_{i1},x_{i2},...,x_{ip})^T, i=1,2,...n.$ $\Sigma=\frac{1}{n-1}\sum_{k=1}^{n}(x_k-\bar{x})(x_k-\bar{x})^T$ is the sample covariance matrix, $\bar{x}=(\bar{x}_1,\bar{x}_2,...,\bar{x}_p)^T$, $x_j=\frac{1}{n}\sum_{i=1}^{n}x_{ij}, i=1,2,...p$. $\hat{e}_i=(\hat{e}_{i1},\hat{e}_{i2},...,\hat{e}_{ip})$ is the orthogonal unitary eigenvector, $\hat{e}_i\hat{e}_i^T=1$, $i=1,2,...p$. $y_i=\hat{e}_{i1}x_1+\hat{e}_{i2}x_2+...+\hat{e}_{ip}x_p$ is the principal component of the ith sample, $i=1,2,...p$ .Substitute n observations, $x_k=(x_{k1},x_{k2},...,x_{kp})^T, k=1,2,...n$. The y_i score of the ith principal component is its n observed values $y_{ki}(k=1,2,...n)$. $\hat{\lambda}_i/\sum_{i=1}^{p}\hat{\lambda}_k(i=1,2,...p)$ is the contribution rate of principal components of the ith sample of X, $\sum_{i=1}^{p}\hat{\lambda}_i$ is the total variance of the sample. $\sum_{i=1}^{m}\hat{\lambda}_i/\sum_{k=1}^{p}\hat{\lambda}_k$ is the cumulative principal component contribution rate of the first m samples.

The moving target is detected by Gaussian background model reconstruction, and the foreground is processed by morphological corrosion expansion and shadow elimination. As shown in figure 2, the gray value of pixels in the moving target area is used as the feature description vector for the analysis of sample principal components. As can be seen from the contribution rate of principal components in table 1 below, the contribution rate of the first six cumulative principal components has reached 93%, covering most of the information tables.

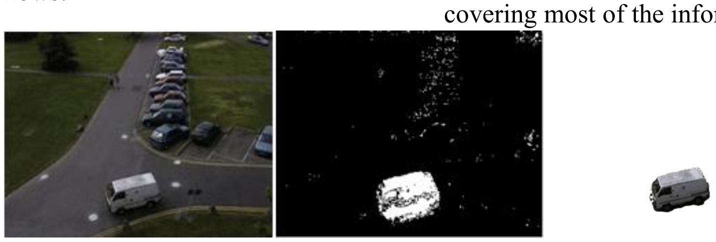
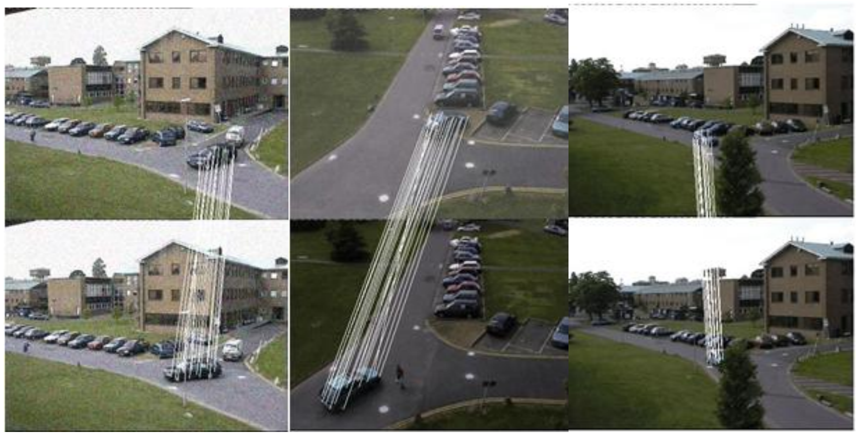


Figure 2 target extraction

Table 1 Contribution rate of principal components of vehicle diagram

principal component	characteristic value	Principal component contribution rate	Cumulative contribution rate
1	4.6	44%	44%
2	2.1	19%	63%
3	1.3	11%	74%
4	1.0	9%	83%
5	0.7	7%	89%
6	0.4	4%	93%



(a)Low resolution (b)brightness change (c)occlusion

Figure 3 Feature matching

Table 2 Comparison of matching efficiency

Matching environment	Threshold=0.55		Threshold=0.65		Threshold=0.75	
	Original SIFT	improved	Original SIFT	improved	Original SIFT	improved
a	532.4	69.5	560.4	75.2	572.1	77.5
b	596.2	73.7	681.5	78.5	682.2	79.3
c	490.7	68.1	515.7	72.1	524.1	74.2

In order to verify the matching effect of the improved SIFT algorithm after dimension reduction, this paper selects the threshold of 0.6 and the image size is 320 * 240. The matching effect is compared between traffic maps in different environments. As shown in figure 3, (a), (b) and (c) are the matching effects of low resolution, brightness change and occlusion environment respectively. Due to the high cumulative contribution rate, the matching effect will not be affected by dimensionality reduction. Table 2 shows the comparison results of the matching time between the original SIFT algorithm and the improved algorithm when the threshold is 0.55, 0.65 and 0.75 respectively. The unit is milliseconds. It can be seen that the SIFT algorithm combined with SPCA processes high-dimensional data under the idea of dimension reduction, uses less uncorrelated variables to reflect most of the variable information, greatly reduces the dimension of feature vector and improves the efficiency of feature matching.

4. FEATURE FUSION

Camshift and SIFT algorithms can extract the global and local features of the image respectively. Under the observation model based on color information and gray level, the target tracking is further stable, and has good robustness to the scale scaling, occlusion, environmental noise and brightness change of the tracked target. In this paper, a multi feature template based on color and gray is established to track moving targets, and multiple features are described by gray and color histograms. The multi feature fusion process is as follows:

The histogram is used to describe different feature spaces respectively, and the feature probability density distribution function of candidate targets and target templates is established.

$$\hat{p}_{u_j}(y) = C_j \sum_{i=1}^{n} k\left(\left\| \frac{y - x_i}{h} \right\|^2\right) \delta(b_j(x_i) - u_j) \text{ ,}$$

$$\hat{q}_{u_j}(y) = C_j \sum_{i=1}^{n} k\left(\left\| \frac{y - x_i}{h} \right\|^2\right) \delta(b_j(x_i) - u_j) \quad (12)$$

In the above formula, u_j is the feature vector, K is the feature type, and j is the j-th feature space. $j = 1,2,3,...K$. $\{x_1, x_2, ...x_n\}$ is the window pixel set, and the weight is modified according to the following formula (13). $a_1, ...a_k$ is the weight coefficient, $\sum_{j=1}^{k} a_j = 1$,m is the quantization level, and multi feature weighting is performed.

$$w_i = \sum_{j=1}^{k} a_j \sum_{u_j=1}^{m_j} \delta(b_j(x_i) - u_j) \sqrt{\frac{\hat{q}_{u_j}}{\hat{p}_{u_j}(y_0)}}$$

$$= a_1 w_i(u_1) + ...a_k w_i(u_k) \quad (13)$$

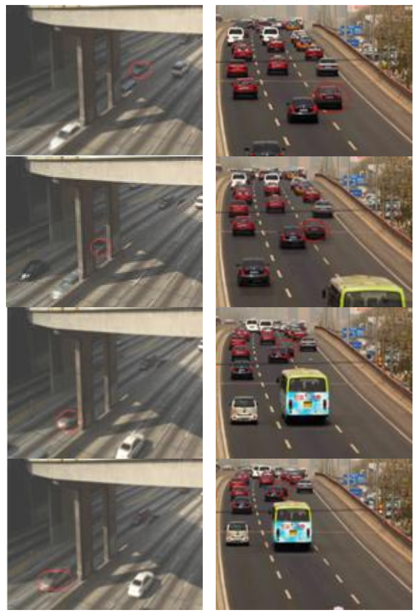
In the process of target movement, the environment changes constantly, and the weight of each feature in different environments should also change adaptively. Let the contribution of feature *j* at a certain time be M_j .

$$M_j = \frac{\sum_{u_j=1}^{m_j} MIN(a_{u_j}, c_{u_j}) - \sum_{u_j=1}^{m_j} MIN(a_{u_j}, b_{u_j})}{\sum_{u_j=1}^{m_j} MIN(a_{u_j}, c_{u_j})} \quad (14)$$

Where a_{u_j} , b_{u_j} , c_{u_j} represent the feature histogram of the target, background and the whole region.

$\sum_{u_j=1}^{m_j} MIN(a_{u_j}, c_{u_j})$, $\sum_{u_j=1}^{m_j} MIN(a_{u_j}, b_{u_j})$ are the intersection value of the histogram of the target and the whole region, foreground and background in space *j*. Weight coefficient $a_j = \frac{M_j}{M_1 + M_2 + M_3 + ... M_K}$, $j = 1, 2, 3, ... K$, and $\sum_{j-1}^{k} a_j = 1$.

By establishing multi feature templates, the interference of the background in different spaces is reduced. Using the different contribution degree of different features, the gray and color models are adaptively set different weights in the tracking process. If the contribution is small, the ability to distinguish between foreground and background is weak. Otherwise, the ability to distinguish is strong, which ensures the stability of tracking. When complete occlusion occurs, the position of the moving target center at the next time can be predicted by quadratic polynomial prediction method.



(a) Occlusion (b) high speed

Figure 4 vehicle tracking

Table 3 Statistics of average vehicle tracking time(ms)

number	Camshift(ms)	Kalman	Particle filte	The paper
1	46.8	56.2	85.1	51.2
2	59.4	67.6	95.9	65.5
3	55.5	65.1	90.7	59.6
4	41.7	52.5	79.5	55.4
5	48.3	61.7	86.1	56.3

As can be seen in figure 4 (a), when the color of the vehicle and the bridge column is similar and there is occlusion, the color features are not disturbed. By using the optimized local invariant features to adaptively increase the weight, the target can still be tracked well. After the target drives out of the shadow area in (a), it continues to track stably. In (b), The vehicle in the is moving fast, and some frames have dynamic virtual shadow, and the tracking effect in the presence of fog and haze weather. Experimental results show that the proposed algorithm is robust to light change, target affine transformation, occlusion, haze weather and other environments.

Table 3 compares the tracking of different tracking algorithms in five groups of different traffic scene videos. The time unit is milliseconds. The above table lists the average time spent calculated by each group. Compared with the Kalman and particle filter algorithms, the improved algorithm in this paper has better time complexity. Although the complexity is slightly higher than the original Camshift algorithm, it has strong robustness against complex environment and meets the needs of real-time monitoring of road traffic.

5. CONCLUSION

Aiming at the complex environment, this paper proposes an adaptive weight update target tracking algorithm based on the fusion of HSV color features and local invariant features based on Camshift. In order to reduce the complexity of the algorithm, the principal component analysis method is used to express the original sample information with uncorrelated new sample information, so as to reduce the dimension, avoid information overlap, reduce the matching time of feature vectors, and describe different feature spaces with histograms. The characteristic probability density distribution function of candidate target and target template is established. The improved algorithm can adaptively update different feature weights for vehicle deformation, frost and fog weather, background noise interference, light change, occlusion and other problems. The features complement each other and have good robustness to complex environments. It has a good application prospect in urban road traffic real-time monitoring system.

ACKNOWLEDGMENT

This work was supported in part by the Xuzhou Science and Technology Plan Project under Grant KC21303, Jiangsu Industry University Research Cooperation Project under Grant BY2021159, the sixth "333 project" of Jiangsu Province.

REFERENCES

[1] Gao, J. . "Analysis of the Application of Traffic Monitoring System in Traffic Unraveling during Subway Construction." Electrical Technology of Intelligent Buildings (2020).

[2] Wang, F. , M. Liu , and S. Wang . "Kalman filter tracking of sequence spot centroid ablated by femtosecond laser." Microwave and Optical Technology Letters 63.2(2021).

[3] Kumar, D. ."Hybrid Unscented Kalman Filter with Rare features for Underwater Target tracking using Passive Sonar Measurements." Optik - International Journal for Light and Electron Optics 226.no. 3(2021):165813.

[4] Du, S. , and Q. Deng . "Unscented Particle Filter Algorithm Based on Divide-and-Conquer Sampling for Target Tracking." Sensors 21.6(2021):2236.

[5] Yang, J. , et al. "Particle filter algorithm optimized by genetic algorithm combined with particle swarm optimization." Procedia Computer Science 187.4(2021):206-211.

[6] Fang, C. , et al. "Comparative study on poultry target tracking algorithms based on a deep regression network." Biosystems Engineering 190(2020):176-183.

[7] Hu, B. , G. Chen , and Q. Liu . "UAV attitude angle measurement system based on machine vision." 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC) IEEE, 2020.

[8] Kumar, M. , and S. Gupta . "2D-human face recognition using SIFT and SURF descriptors of face's feature regions." The Visual Computer 37.11(2021).

[9] Parashivamurthy, R. , C. Naveena , and Y. Kumar . "SIFT and HOG features for the retrieval of ancient Kannada epigraphs." IET Image Processing (2021).

[10] Lv, W. , et al. "Research and Application of Intersection Clustering Algorithm Based on PCA Feature Extraction and K-Means." Journal of Physics: Conference Series 1861.1(2021):012001 (7pp).

[11] Zhang, Y. , D. Xiao , and Y. Liu . "Automatic Identification Algorithm of the Rice Tiller Period Based on PCA and SVM." IEEE Access PP.99(2021):1-1.

[12]