

## Research Article

# CamShift Tracking Method Based on Target Decomposition

**Chunbo Xiu, Shian Wei, Rongfeng Wan, Yi Cheng, Jing Luo, and Huixin Tian**

*Key Laboratory of Advanced Electrical Engineering and Energy Technology, School of Electrical Engineering and Automation, Tianjin Polytechnic University, Tianjin 300387, China*

Correspondence should be addressed to Chunbo Xiu; [xiuchunbo@tjpu.edu.cn](mailto:xiuchunbo@tjpu.edu.cn)

Received 6 December 2014; Revised 23 January 2015; Accepted 23 January 2015

Academic Editor: Pasquale Memmolo

Copyright © 2015 Chunbo Xiu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to avoid the inaccurate location or the failure tracking caused by the occlusion or the pose variation, a novel tracking method is proposed based on CamShift algorithm by decomposing the target into multiple subtargets for location separately. Distance correlation matrices are constructed by the subtarget sets in the template image and the scene image to evaluate the correctness of the location results. The error locations of the subtargets can be corrected by resolving the optimization function constructed according to the relative positions among the subtargets. The directions and sizes of the correctly located subtargets with CamShift algorithm are updated to reduce the disturbance of the background in the tracking progress. Simulation results show that the method can perform the location and tracking of the target and has better adaptability to the scaling, translation, rotation, and occlusion. Furthermore, the computational cost of the method increases slightly, and its average tracking computational time of the single frame is less than 25 ms, which can meet the real-time requirement of the TV tracking system.

## 1. Introduction

Target tracking is a key technology in computer vision field. It can be widely applied to industry, surveillance, robotics, human machine interface, and so forth [1–4]. It is a challenging task in real-world scenarios to track a target precisely due to the variability of the target's movements, shapes, clutter, and occlusions. There are many factors which impact the tracking performance [5–8]. Firstly, the motion of the target is random and uncertain. The target may change its posture in the tracking process, which can lead to the scaling, translation, or rotation of the target in the tracking image. The size and shape of the target in the current scene may be different from those in the previous one or the target template. Secondly, the illumination in the scene may be instable. Uneven illumination or variant brightness can cause disturbances to target tracking. Besides, complex background affects the tracking performance badly. Note that the complex background means that color of background is close to that of the target rather than that the content of the background is complex. In other word, it is difficult to distinguish the target from the background with the similar color.

Some tracking algorithms have been proposed to deal with the problems above. For instance, superpixel tracking

can handle heavy occlusion and recover from drifts [9]. And patch-based tracking algorithm decomposes a target into several patches based on appearance similarity and spatial distribution to recognize the target in the complex background [10]. Generally, improving the accuracy of the target model is one of the effective methods to enhance the location accuracy, but it has the cost of computational time. Therefore, a method is needed to model an accurate target and perform target tracking precisely with less computation. MeanShift [11, 12] algorithm and CamShift (Continuously Adaptive Mean-SHIFT) [13] algorithm have attracted more attention of scholars for their characteristics of fast matching. Particularly, CamShift can adaptively adjust the size of the target area, so that it has good adaptability to the pose variation of the target [14, 15]. These two methods have good tracking performance under the condition of high-contrast between the target and the background. However, when the background is complex, the adaptive adjustment of the size of the target area can be disturbed by some background information, which may cause lower modeling accuracy, location error, or even tracking failure [16]. Therefore, some improved methods are proposed to enhance the accuracy of the location. For instance, one of the effective ways is to fuse some other features [17, 18], such as shape, edge, or texture, to



FIGURE 1: The initial template.

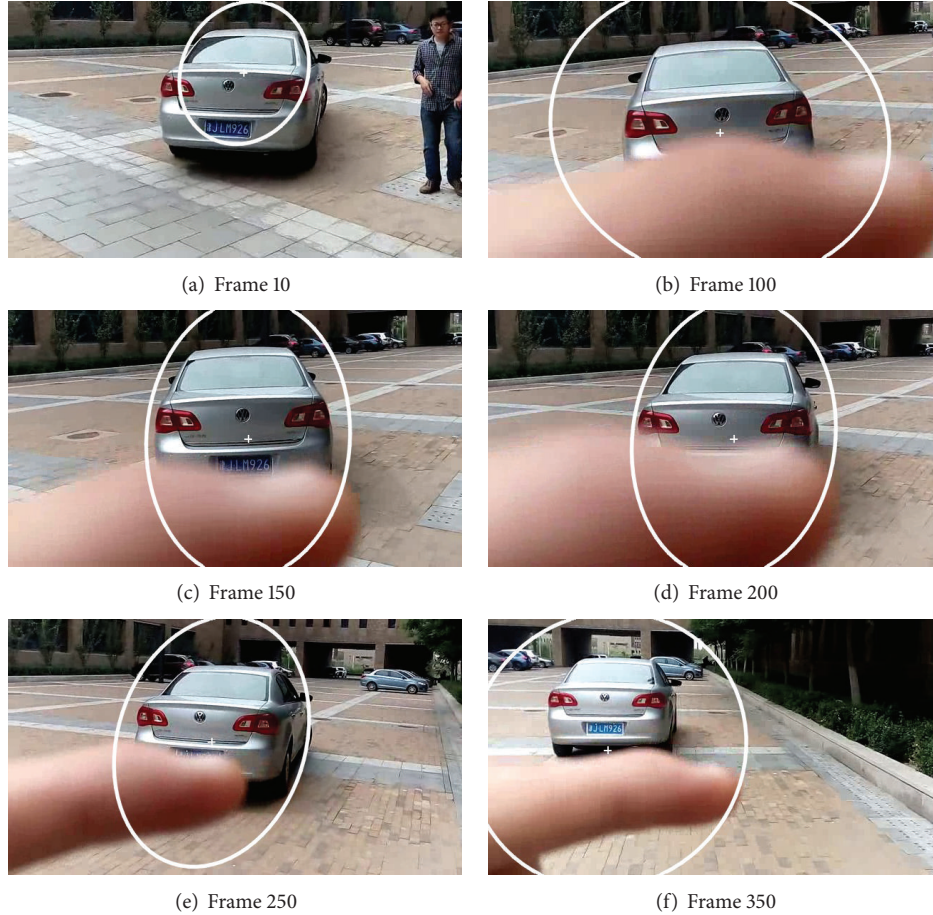


FIGURE 2: The tracking results of the basic CamShift algorithm.

improve the modeling precision. However, that needs a larger computational cost and makes the model more complex.

In order to improve the reliability of the target tracking algorithm, an improved CamShift tracking strategy is proposed to enhance the performance of the target tracking. The target is decomposed into multiple subtargets. Each subtarget is tracked separately by CamShift algorithm. The error locations of subtargets can be corrected according to the relative positions among all the subtargets. The target can be synthesized by these subtargets. In this tracking strategy, the error correction can enhance the reliability of target tracking result in high location accuracy. Particularly, this strategy shows the good adaptability when the target is occluded. Different from the superpixel tracking algorithm [9] and

patch-based tracking algorithm [10], the method proposed in this paper adopts the CamShift algorithm to perform the subtarget tracking, and the correct tracking result can be obtained by the error correction as long as some of the subtargets are located correctly.

## 2. Subtarget Set and Distance Correlation Matrix

Suppose that the target  $A$  contains  $n$  distinguishing feature areas. Each distinguishing feature area can be regarded as a subtarget. All the subtargets are grouped into a subtarget set  $U$ . Thus, the target can be described by the subtarget set  $U$ .

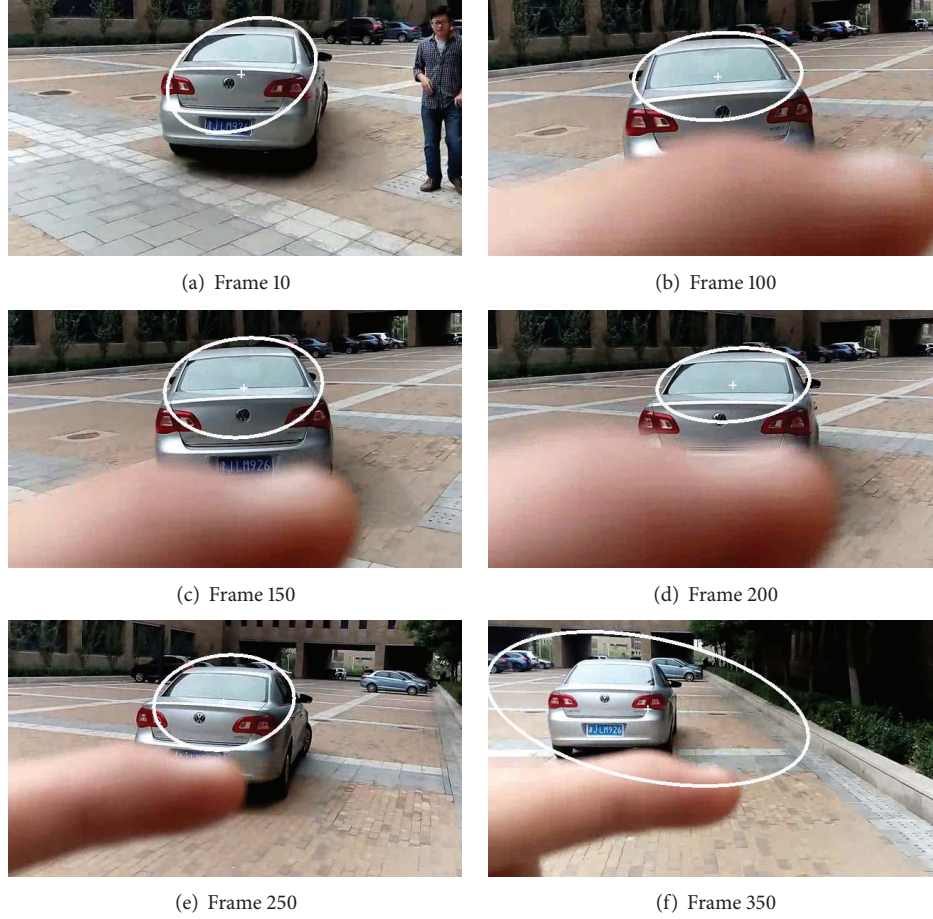


FIGURE 3: The tracking results of the CamShift algorithm based on multifeature fusion.

One element in the subtarget set represents the center coordinates of one subtarget accordingly. Set the center coordinates of the subtarget  $i$  as  $(x_i, y_i)$ ; the subtarget set  $U$  can be described as

$$U = [(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)]^T. \quad (1)$$

The distance between the  $i$ th and the  $j$ th subtargets is defined as

$$d_{ij} = \text{sqrt} \left[ (x_i - x_j)^2 + (y_i - y_j)^2 \right], \quad (2)$$

where  $(x_i, y_i)$  and  $(x_j, y_j)$  are the center coordinates of the  $i$ th and the  $j$ th subtargets in the subtarget set  $U$ .

The distance correlation matrix  $D_n$  of the subtarget set can be defined as

$$D_n = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nn} \end{bmatrix}. \quad (3)$$

The target motion can be seen as the rigid body motion, and the ratios of the relative distances among all the subtargets are fixed. According to these relative distances, error locations of subtargets can be corrected.

### 3. Target Tracking Based on Target Decomposition

After the target decomposing, the hue histogram of each subtarget is modeled in the HSV color space, CamShift algorithm is used to locate every subtarget, and the error locations of subtargets can be corrected by optimization calculation. The position of the target in the scene can be determined according to the locations of subtargets. The target tracking process can be described as follows.

*Step 1.* Set the position coordinates of the  $i$ th pixel in the  $k$ th subtarget region as  $(x_i^{(k)}, y_i^{(k)})$ ,  $i = 1, 2, \dots$ , and the center position coordinates of the  $k$ th subtarget as  $(x_c^{(k)}, y_c^{(k)})$ . The hue histogram of the  $k$ th subtarget can be defined as

$$\hat{q}_k = \{\hat{q}_u^{(k)}\}, \quad u = 1, 2, \dots, n \quad (4)$$

$$\hat{q}_u^{(k)} = C_q \sum_{i=1}^n K \left( \left\| \frac{(x_i^{(k)}, y_i^{(k)}) - (x_c^{(k)}, y_c^{(k)})}{h} \right\| \right) \cdot \delta [b(x_i^{(k)}, y_i^{(k)}) - u], \quad (5)$$



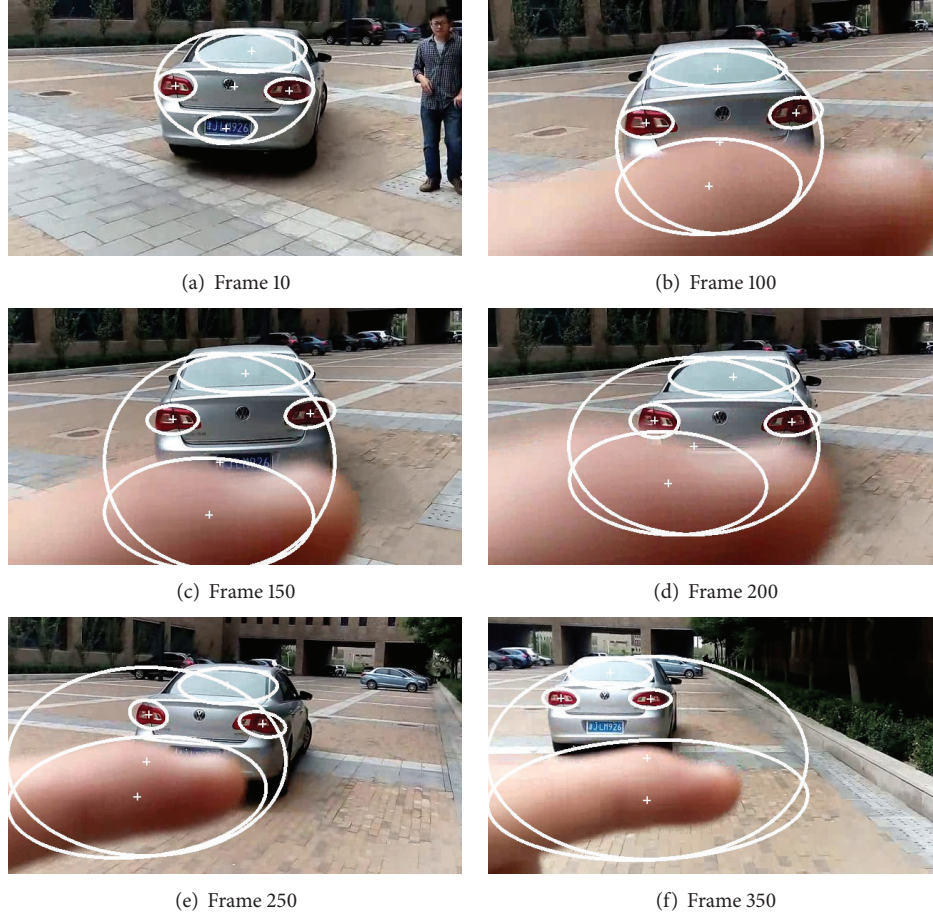


FIGURE 4: The tracking results of our method without the correction.

where  $n$  is the hue levels.  $\hat{q}_u^{(k)}$  is the probability distribution of the hue,  $b(x_i^{(k)}, y_i^{(k)})$  represents the characteristic value of the hue quantified of the pixel  $(x_i^{(k)}, y_i^{(k)})$ , and  $\delta$  is the Kronecker delta function defined as

$$\delta[b(x_i^{(k)}, y_i^{(k)}) - u] = \begin{cases} 1, & \text{if } (b(x_i^{(k)}, y_i^{(k)}) = u) \\ 0, & \text{else.} \end{cases} \quad (6)$$

$K$  is the Gaussian basis functions defined as

$$K\left(\left\|\frac{(x_i^{(k)}, y_i^{(k)}) - (x_c^{(k)}, y_c^{(k)})}{h}\right\|\right) = \exp\left(-\left(\frac{\sqrt{(x_i^{(k)} - x_c^{(k)})^2 + (y_i^{(k)} - y_c^{(k)})^2}}{h}\right)^2\right), \quad (7)$$

where  $h$  is the bandwidth of the Gaussian basis functions.

*Step 2.* The probability projection image of the search window of each subtarget can be built by the back-projection of the hue histogram. Set  $u$  to be the hue characteristic value of

the pixel  $(x_i^{(k)}, y_i^{(k)})$  in the search window of the  $k$ th subtarget. The gray value  $I(x_i^{(k)}, y_i^{(k)})$  of the pixel  $(x_i^{(k)}, y_i^{(k)})$  in the probability projection image can be defined as

$$I(x_i^{(k)}, y_i^{(k)}) = \left\lfloor \frac{\hat{q}_u^{(k)}}{\max\{\hat{q}_j^{(k)} \mid j = 1, 2, \dots, n\}} \times 255 \right\rfloor, \quad (8)$$

where the symbol  $\lfloor \cdot \rfloor$  represents the rounding operation. The larger the gray value of the pixel in the probability projection image, the larger the probability that the pixel belongs to the target area.

*Step 3.* Zero and first order moments of the search window  $k$  can be calculated as

$$M_{00}^{(k)} = \sum_{x_i^{(k)}} \sum_{y_i^{(k)}} I(x_i^{(k)}, y_i^{(k)}), \quad (9)$$

$$M_{10}^{(k)} = \sum_{x_i^{(k)}} \sum_{y_i^{(k)}} x_i^{(k)} I(x_i^{(k)}, y_i^{(k)}), \quad (10)$$

$$M_{01}^{(k)} = \sum_{x_i^{(k)}} \sum_{y_i^{(k)}} y_i^{(k)} I(x_i^{(k)}, y_i^{(k)}). \quad (11)$$



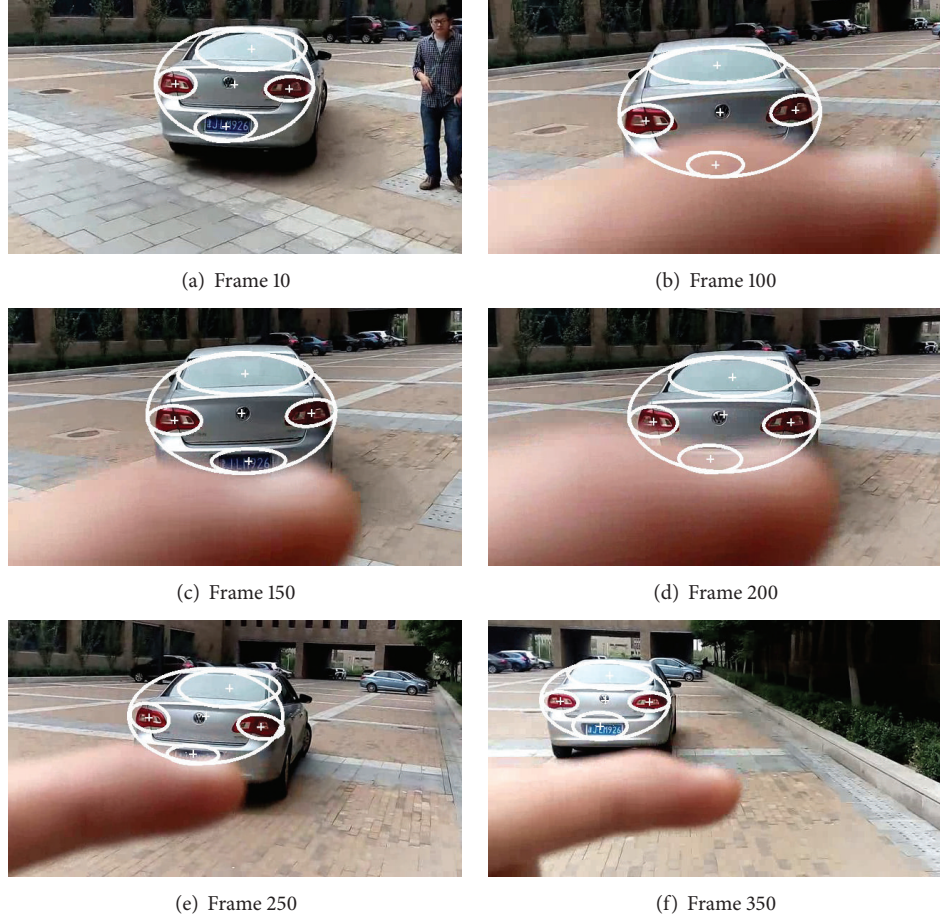


FIGURE 5: The tracking results of our method with the correction.



FIGURE 6: The initial template.

*Step 4.* The center of mass  $(x_w^{(k)}, y_w^{(k)})$  of the search window  $k$  can be calculated as

$$x_w^{(k)} = \frac{M_{10}^{(k)}}{M_{00}^{(k)}}, \quad (12)$$

$$y_w^k = \frac{M_{01}^{(k)}}{M_{00}^{(k)}}. \quad (13)$$

*Step 5.* Adaptively adjust the size of the search window  $k$  as

$$s_k = 2\sqrt{\frac{M_{00}^{(k)}}{256}}. \quad (14)$$

Move the center of the search window of the subtarget  $k$  to its center of mass. Repeat Step 2 to Step 5 until the shift distance is less than the threshold.



FIGURE 7: The tracking results of the basic CamShift algorithm.

*Step 6.* Judge the location correctness of the subtargets.

Suppose that the coordinate  $(x'_i, y'_i)$  is the located position of the subtarget  $i$ . The new subtarget set  $U'$  in the scene can be defined as

$$U' = [(x'_1, y'_1), (x'_2, y'_2), \dots, (x'_n, y'_n)]^T. \quad (15)$$

The new distance correlation matrix  $D'_n$  of the subtarget set in the scene can be defined as

$$D'_n = \begin{bmatrix} d'_{11} & d'_{12} & \cdots & d'_{1n} \\ d'_{21} & d'_{22} & \cdots & d'_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ d'_{n1} & d'_{n2} & \cdots & d'_{nn} \end{bmatrix}. \quad (16)$$

The correctness of the subtargets located can be judged according to the difference between the two distance correlation matrixes  $D_n$  and  $D'_n$ . Firstly,  $(n^2 - n)/2$  centers  $\{c_{ij}, i = 1, 2, 3, \dots, n-1; j = 1, 2, 3, \dots, n; \text{ and } i < j\}$  can be got by  $D_n$  and  $D'_n$  as follows:

$$c_{ij} = \frac{d'_{ij}}{d_{ij}}. \quad (17)$$

Secondly, based on the centers above,  $(n^2 - n)/2$  kernel functions can be defined as

$$g_{ij} \left( \left\| \frac{x - c_{ij}}{h_{ij}} \right\| \right) = e^{-\|(x - c_{ij})/h_{ij}\|^2}, \quad (18)$$

where  $i = 1, 2, \dots, (n-1), j = 2, 3, \dots, n$ , and  $i < j$ .





FIGURE 8: The tracking results of the CamShift algorithm based on multifeature fusion.

The sum function  $G$  of the kernel functions above is defined as

$$G = \sum_{i,j} g_{ij}. \quad (19)$$

The position corresponding to the maximum of the sum function  $G$  is determined as the kernel center  $c_g$ , and the judgment matrix  $E$  is defined as

$$E = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1n} \\ e_{21} & e_{22} & \cdots & e_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ e_{n1} & e_{n2} & \cdots & e_{nn} \end{bmatrix}, \quad (20)$$

where

$$e_{ij} = e_{ji} = \begin{cases} 1, & \text{if } |c_{ij} - c_g| < \eta \\ 0, & \text{else,} \end{cases} \quad (21)$$

where  $\eta$  is the center threshold. If all elements in the row  $i$  are zero, that is,  $e_{ij} = 0, j = 1, 2, \dots, n$ , it can be determined that the subtarget  $i$  is located wrongly.

**Step 7.** Correct the error locations of subtargets.

If the error location subtargets exist, they can be corrected by the subtargets located correctly. Suppose that there are  $m$  subtargets located correctly, the center coordinates of the subtarget  $j$  is  $(x'_j, y'_j)$ , where  $j = 1, 2, \dots, m$ , and the center





FIGURE 9: The tracking results of our method without the correction.

coordinates corrected of the error location subtarget  $i$  is  $(\hat{x}_i, \hat{y}_i)$ . The distance between them can be defined as

$$d''_{ij} = \text{sqrt} \left[ (\hat{x}_i - x'_j)^2 + (\hat{y}_i - y'_j)^2 \right]. \quad (22)$$

The evaluation function  $J$  can be defined as

$$J = \sum_{j=1}^m \left( \frac{d''_{ij}}{d_{ij}} - c_g \right)^2. \quad (23)$$

The best corrected center coordinates are determined where the evaluation function above is minimized.

*Step 8.* Adjust the direction angle and size of the subtargets located correctly.

Second order moments of the search window  $k$  can be calculated as

$$M_{11}^{(k)} = \sum_{x_i^{(k)}} \sum_{y_i^{(k)}} x_i^{(k)} y_i^{(k)} I(x_i^{(k)}, y_i^{(k)}), \quad (24)$$

$$M_{20}^{(k)} = \sum_{x_i^{(k)}} \sum_{y_i^{(k)}} (x_i^{(k)})^2 I(x_i^{(k)}, y_i^{(k)}), \quad (25)$$

$$M_{02}^{(k)} = \sum_{x_i^{(k)}} \sum_{y_i^{(k)}} (y_i^{(k)})^2 I(x_i^{(k)}, y_i^{(k)}). \quad (26)$$



FIGURE 10: The tracking results of our method with the correction.



FIGURE 11: The initial template.





FIGURE 12: The tracking results of the basic CamShift algorithm.

Thus,

$$a_k = \frac{M_{20}^{(k)}}{M_{00}^{(k)}} - (x_w^{(k)})^2, \quad (27)$$

$$b_k = \frac{M_{11}^{(k)}}{M_{00}^{(k)}} - x_w^{(k)} y_w^{(k)}, \quad (28)$$

$$c_k = \frac{M_{02}^{(k)}}{M_{00}^{(k)}} - (y_w^{(k)})^2. \quad (29)$$

The major axis of the ellipse area of the subtarget  $k$  is calculated as

$$L_k = \begin{cases} L'_k, & \text{if } \frac{|L_k - L'_k|}{L_k} < \varepsilon \\ \alpha L_k + (1 - \alpha) L'_k, & \text{else,} \end{cases} \quad (30)$$

where

$$L'_k = \sqrt{\frac{(a_k + c_k) + \sqrt{b_k^2 + (a_k - c_k)^2}}{2}}. \quad (31)$$

The minor axis of the ellipse area of the subtarget  $k$  is calculated as

$$W_k = \begin{cases} W'_k, & \text{if } \frac{|W_k - W'_k|}{W_k} < \varepsilon \\ \alpha W_k + (1 - \alpha) W'_k, & \text{else,} \end{cases} \quad (32)$$

where

$$W'_k = \sqrt{\frac{(a_k + c_k) - \sqrt{b_k^2 + (a_k - c_k)^2}}{2}}. \quad (33)$$





FIGURE 13: The tracking results of the CamShift algorithm based on multifeature fusion.

The direction angle of the ellipse area of the subtarget  $k$  is calculated as

$$\theta_k = \begin{cases} \theta'_k, & \text{if } \frac{|\theta_k - \theta'_k|}{\theta_k} < \varepsilon \\ \alpha\theta_k + (1 - \alpha)\theta'_k, & \text{else,} \end{cases} \quad (34)$$

where

$$\theta'_k = \frac{1}{2} \arctan \left( \frac{b_k}{a_k - c_k} \right). \quad (35)$$

The direction angles and sizes of the error location subtargets are not adjusted.

*Step 9.* Return to Step 1 to track the target in the next scene.

In (17),  $c_{ij}$  is the ratio of the distances  $d'_{ij}$  and  $d_{ij}$ . When all the subtargets are located correctly, all the ratios  $\{c_{ij}, i = 1, 2, 3, \dots, n-1; j = 1, 2, 3, \dots, n; \text{ and } i < j\}$  are equal for the geometric similarity regardless of whether the scale of the target varies or not. In other words, the kernel center  $c_g$  represents the geometric similarity ratio of the targets in the template image and the scene image. When the difference between  $c_{ij}$  and  $c_g$  is large, at least one subtarget,  $i$  or  $j$ , may not be located correctly. The corrected position of the subtarget located wrongly corresponds to the minimization result of the evaluation function  $J$ . So, in (17)–(21), the ratio of the relative distances between the subtargets in the target template and the scene is used to judge the correctness of the tracking results, which can ensure that the tracking strategy is also effective to the scale variation of the target. Therefore,

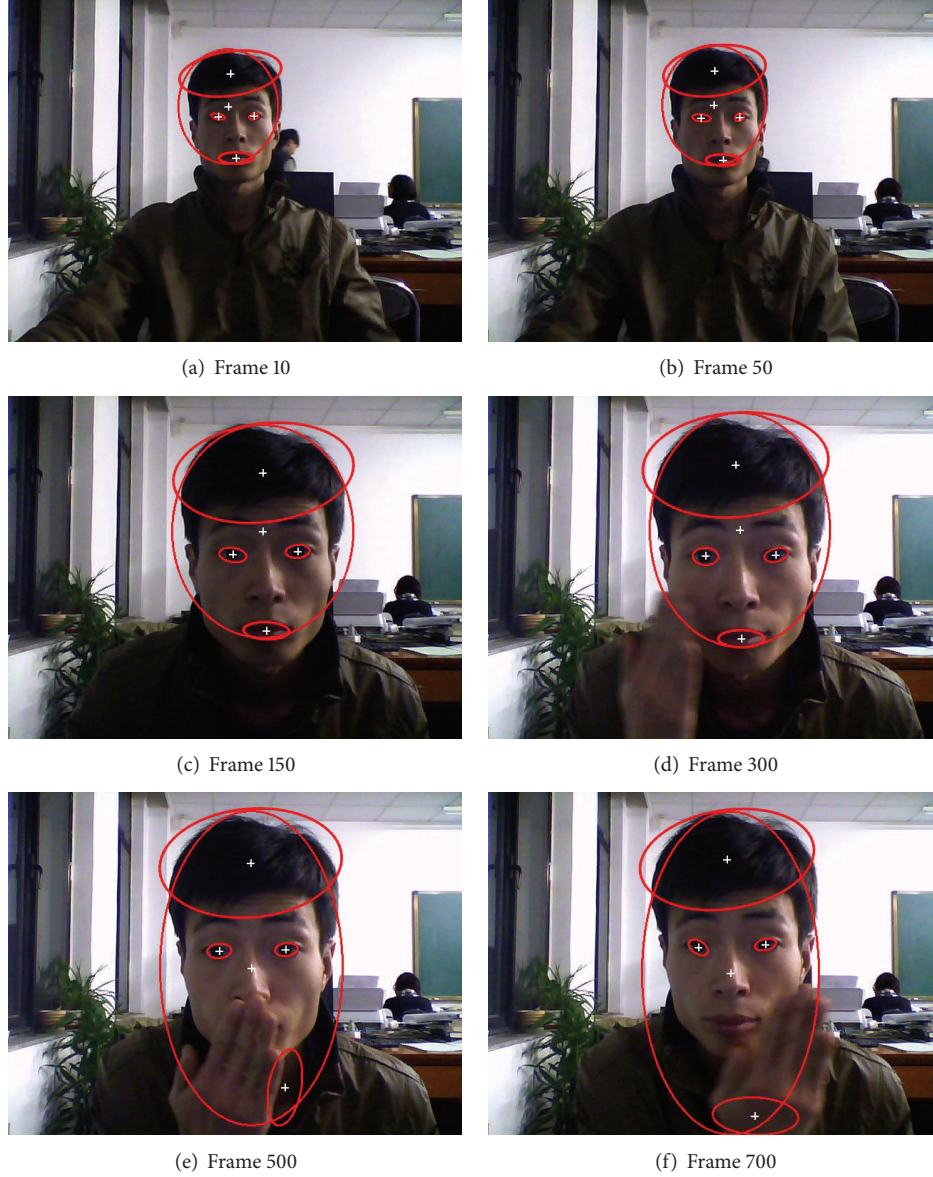


FIGURE 14: The tracking results of our method without the correction.

the tracking strategy has good adaptability to the translation, rotation, scaling, and occlusion of the target.

#### 4. Simulation Results

The tracking strategy proposed in this paper is used to perform the target tracking in the complex circumstances of the target partial occlusion, the pose variation, and target scaling. The feasibility and effectiveness of the proposed method is shown by the comparative analysis with the basic CamShift algorithm and the improved CamShift algorithm based on multifeature fusion [18].

*4.1. Tracking Experiment for the Target Occluded Partially.* Figure 1 shows the initial template, the white ellipse, in the first frame scene. The rear elevation of the car is chosen as

the initial template which contains two red taillights, one blue license plate, and one black rear window. The target partial occlusion occurs during the video sampling.

Comparison tracking results of different tracking methods are shown in Figures 2 to 5.

The tracking results shown in Figure 2 are obtained by the basic CamShift algorithm (Method 1). It is clear that the results located by the basic CamShift algorithm are inaccurate. Particularly, because the size of the target changes with the target motion, a lot of background information, which causes serious interference with the target location, is included in the target area.

The tracking results shown in Figure 3 are obtained by the CamShift algorithm based on multifeature fusion (Method 2). Multiple features can improve the location accuracy to some extent. But it still cannot restrain the interference of





FIGURE 15: The tracking results of our method with the correction.



FIGURE 16: The initial template.





FIGURE 17: The tracking results of the basic CamShift algorithm.

the occlusion, so large location error is inevitable in the simulation results above.

The tracking results shown in Figure 4 are obtained by our method but without the correction of the error location subtarget (Method 3). It cannot correctly locate the subtarget occluded, which leads to the location inaccuracy of the target.

The tracking results shown in Figure 5 are obtained by our method with the correction of the error location subtarget (Method 4). Though subtargets occluded exist in the tracking process, they can be located accurately according to other subtargets, which can ensure the accurate target location results. After targets move out of the occlusion region, all the subtargets go back to the normal tracking. Thus, the tracking result can be always right, and the high location accuracy can be obtained in the long-term tracking process.

Table 1 shows the center deviation between the target located and the real target. And Table 2 shows the percentage

TABLE 1: The center deviation between the target located and the real target.

	The center deviation/pixel					
	(a)	(b)	(c)	(d)	(e)	(f)
Method 1	27	25	33	33	38	64
Method 2	20	53	39	53	38	61
Method 3	7	40	77	54	99	105
Method 4	7	6	14	8	8	2

deviations of the major axis and minor axis of the ellipse target area. The center deviation is the distance between the located center and the real center of the target. And the percentage deviation of the major (minor) axis is the ratio of the absolute deviation between the major (minor) axis of the ellipse target area containing all the subtargets and the target



FIGURE 18: The tracking results of the CamShift algorithm based on multifeature fusion.

TABLE 2: The percentage deviations of the major (minor) axis of the ellipse target area.

	The percentage deviation/%											
	(a)	(b)	(c)	(d)	(e)	(f)						
	Major	Minor	Major	Minor	Major	Minor	Major	Minor	Major	Minor	Major	Minor
Method 1	13.0	40.1	79.3	133.8	14.8	134.9	17.6	140.3	35.9	170.7	145.2	300
Method 2	6.2	19	7.8	31.6	11.6	15.0	10.2	33.5	5.9	7.6	168.8	123.2
Method 3	4.8	5.6	10.5	45.0	30.1	87.9	43.0	54.6	91.6	103.8	164.7	190.9
Method 4	4.8	5.6	4.1	1.1	4.8	2.4	5.3	1.8	3.4	2.3	2.3	3.0

template in the scene to the major (minor) axis of the target template in the scene.

The size of the scene image is  $640 \times 360$ . From the experimental data, whether according to the center deviation or the percentage deviation, the tracking strategy proposed in this paper has better tracking performance. The target center

located is closer to the real target center, and the target area is certainly consistent with the target template.

**4.2. Tracking Experiment for the Target with Pose Change.** Figure 6 shows the initial template, the red ellipse, in the first frame scene. The rear elevation of the car is also chosen



FIGURE 19: The tracking results of our method without the correction.

TABLE 3: The center deviation between the target located and the real target.

	The center deviation/pixel					
	(a)	(b)	(c)	(d)	(e)	(f)
Method 1	20	33	70	152	6	24
Method 2	10	40	95	204	44	38
Method 3	7	4	9	69	41	23
Method 4	7	4	9	24	18	4

as the initial template which contains two red taillights, one blue license plate, and one black rear window. The pose of the target changes during the video sampling. The color of both the car body and its side window is the same as the target, which seriously disturbs the location results.

Comparison tracking results of different tracking methods are shown in Figures 7 to 10.

The tracking results shown in Figure 7 are obtained by the basic CamShift algorithm (Method 1). The tracking result is unstable because of the background interference. The target area contains a lot of background, which cause the inaccurate location of the target.

The tracking results shown in Figure 8 are obtained by the CamShift algorithm based on multifeature fusion (Method 2). Though the multiple features can restrain the interference of the background but cannot restrain the interference caused by the car body or its window. The car can be tracked, but the tracked areas are different from the initial target template. So the location accuracy is not satisfactory.

The tracking results shown in Figure 9 are obtained by our method but without the correction of the error location subtarget (Method 3). Some subtargets disappear from the





FIGURE 20: The tracking results of our method with the correction.

TABLE 4: The percentage deviations of the major (minor) axis of the ellipse target area.

	The percentage deviation/%											
	(a)	(b)	(c)	(d)	(e)	(f)						
	Major	Minor	Major	Minor	Major	Minor	Major	Minor	Major	Minor	Major	Minor
Method 1	40.5	111.7	170.5	83.1	184.5	206.1	170.7	70.0	269.9	296.9	468.3	542.6
Method 2	29.7	68.0	26.3	61.0	100.5	83.3	142.4	60.0	35.8	51.5	90.6	68.8
Method 3	2.3	6.4	3.6	2.7	6.7	3.5	21.4	11.2	19.1	6.2	15.4	6.6
Method 4	2.3	6.4	3.6	2.7	6.7	3.5	7.4	10.0	6.9	2.1	7.7	3.3

scene because of the pose change of the car. The disappearing subtargets cannot be located correctly. For instance, because the left light disappears, in Figure 9(d), it is wrongly located to the right light, which causes the location error of the target.

The tracking results shown in Figure 10 are obtained by our method with the correction of the error location subtarget (Method 4). The disappearing subtargets can be located correctly according to the relative distances among

TABLE 5: The center deviation between the target located and the real target.

	The center deviation/pixel					
	(a)	(b)	(c)	(d)	(e)	(f)
Method 1	58	78	81	136	153	148
Method 2	57	76	80	133	137	133
Method 3	7	8	6	5	65	74
Method 4	7	8	6	5	8	4

TABLE 6: The percentage deviations of the major (minor) axis of the ellipse target area.

	The percentage deviation/%											
	(a)		(b)		(c)		(d)		(e)		(f)	
	Major	Minor	Major	Minor	Major	Minor	Major	Minor	Major	Minor	Major	Minor
Method 1	5.1	50.0	7.8	50.3	5.8	16.9	5.2	6.5	17.3	6.4	5.2	8.8
Method 2	3.6	26.0	5.0	61.5	13.6	13.2	10.8	10.6	3.4	4.9	4.3	22.5
Method 3	1.7	2.6	3.5	3.7	1.2	0.7	0.8	1.2	1.5	33.1	8.6	48.0
Method 4	1.7	2.6	3.5	3.7	1.2	0.7	0.8	1.2	1.1	2.4	3.0	1.6

all the subtargets, for instance, in Figure 10(d), which ensures the accuracy of the target tracking. When the disappearing subtargets reappear in the latter frame again, for instance, left light in Figures 10(e) and 10(f), they can be tracked normally. The target can always be tracked accurately.

Tables 3 and 4 show the center deviation and the percentage deviations of the major (minor) axis separately.

The size of the scene image is  $640 \times 480$ . From the experimental results, the same conclusion as the above can be got. The tracking strategy proposed in this paper has better tracking performance. Each subtarget has different distinguishing feature, which can prevent the single color to disturb the tracking results of all the subtargets. And though the occlusion, the change of the pose, and background can cause the error location of some subtargets, the relative distances among the subtargets can correct the error location subtargets, which ensure the good adaptivity of the tracking strategy. Therefore, the tracking strategy can be applied in some complex cases.

**4.3. Tracking Experiment for the Target with the Scale Variation.** Figure 11 shows the initial template, the red ellipse, in the first frame scene. The head is chosen as the initial template which contains eyes, mouth, and hair. The scale of the target varies and the partial occlusion occurs during the tracking process. Besides, the skin color regions outside of the template, for instance the neck, can disturb the location results.

Comparison tracking results of different tracking methods are shown in Figures 12 to 15.

The tracking results shown in Figure 12 are obtained by the basic CamShift algorithm (Method 1). The tracking results shown in Figure 13 are obtained by the CamShift algorithm based on multifeature fusion (Method 2). Because the skin color is the main color of the target, the tracking results obtained by the two methods above are about the same. Both the neck and the hand disturb the location results.

The tracking results shown in Figure 14 are obtained by our method but without the correction of the error location

TABLE 7: The center deviation between the target located and the real target.

	The center deviation/pixel					
	(a)	(b)	(c)	(d)	(e)	(f)
Method 1	25	174	252	179	121	129
Method 2	19	39	14	51	174	135
Method 3	18	112	106	100	103	95
Method 4	4	1	4	4	7	3

subtarget (Method 3). The correct tracking results can be obtained when only the scale of the target increases and no occlusion occurs. When the hand covers the mouth in Figures 14(e) and 14(f), error location occurs at the later stage of the target tracking.

The tracking results shown in Figure 15 are obtained by our method with the correction of the error location subtarget (Method 4). Though the scale of the target varies during the tracking process, the correct tracking results can still be obtained because the geometric similarity ratio of the subtargets is used to correct the location results. It is also proved that the method is adaptive to the scaling of the target.

Tables 5 and 6 show the center deviation and the percentage deviations of the major (minor) axis separately.

The size of the scene image is  $640 \times 480$ . From the experimental results, the location accuracies of method 1 and method 2 are similar because Camshift algorithm is suitable to track the target with a large number of the skin color regions. However, the two methods cannot distinguish the skin color regions in the background or the target, so the location error is big. Our method can correct the target position by the geometric similarity ratio of the subtargets, so the location accuracy can be improved.

**4.4. Tracking Experiment for the Target from PETS 2001 Dataset.** A target in the Camera1.mov from PETS 2001 Dataset is tracked by the methods above. Figure 16 shows the initial template, the red ellipse, in the first frame scene. The



TABLE 8: The percentage deviations of the major (minor) axis of the ellipse target area.

	The percentage deviation/%											
	(a)		(b)		(c)		(d)		(e)		(f)	
	Major	Minor	Major	Minor	Major	Minor	Major	Minor	Major	Minor	Major	Minor
Method 1	45.1	147.5	619.8	139.6	596.4	168.3	592.4	57.5	778.3	397.1	797.3	297.0
Method 2	36.3	117.5	84.6	160.4	91.7	90.2	224.1	142.5	417.6	365.7	783.6	206.1
Method 3	60.4	112.5	603.3	268.8	661.9	339.0	710.1	372.5	764.9	434.3	776.7	487.9
Method 4	2.2	5.0	1.1	2.1	3.6	9.8	6.3	5.0	9.4	8.6	5.5	15.2

TABLE 9: Comparisons on computational costs.

Experiment	Method 1	Method 2	Method 3	Method 4
Tracking experiment for the target occluded partially	12.85	31.62	16.32	18.43
Tracking experiment for the target with pose change	13.72	32.31	16.82	19.56
Tracking experiment for the target with the scale variation	14.32	35.44	19.43	22.31
Tracking experiment for the target from PETS 2001 Dataset	13.85	34.75	17.45	20.04

rear elevation of the car is still chosen as the initial template which contains two taillights, one license plate, and one rear window. The target tracking is performed in the overtaking process.

Comparison tracking results of different tracking methods are shown in Figures 17 to 20.

From Figure 17, most of colors in the target are similar to that of the background, so the basic CamShift algorithm is hard to track the target effectively. Likewise, from Figure 18, though the CamShift algorithm based on multifeature fusion can improve the tracking performance, it still fails at last. From Figure 19, it can be seen that two taillights and one license plate, which is significantly different from the background, can be tracked effectively. However, it is difficult to track the rear window which has the similar color to that of the road in the background. Therefore, the whole target cannot be located effectively. After the error correction, as shown in Figure 20, the target can be tracked correctly.

Table 7 shows the center deviation between the target located and the real target. And Table 8 shows the percentage deviations of the major axis and minor axis of the ellipse target area.

The size of the scene image is  $640 \times 480$ . From the experimental data, when the target can be decomposed into several subtargets with different saliency colors, the target tracking can be performed easily by the subtargets located correctly. Since the different subtarget can be a saliency subtarget in different background, the method proposed in this paper has better adaptability than others.

**4.5. Comparisons on Computational Costs.** The VS2010 is chosen as the environment for software development to perform the experiments above. The computer configuration is AMD A4-6219 APU with AMD Radeon R3 Graphics and 8 G RAM. Computational costs, the average tracking computational time of the single frame, are compared in Table 9.

Though the number of the subtargets tracked is increased in our method, the size of each subtarget is small. Therefore,

from the experiments results, it can be seen that the computational cost of our method increases slightly. The computational cost on the error correction is little. Instead, the computational cost of the CamShift algorithm based on multifeature fusion is large because of the extraction of the fused features. Furthermore, the average tracking computational time of the single frame of every method above is less than 40 ms and that of our method is less than 25 ms. Therefore, all the methods above can meet the real-time requirement of the TV tracking system.

## 5. Conclusions

In order to meet the requirement of the target precise location and tracking in the complex cases, the tracking strategy based on the target decomposition is proposed. The target is decomposed into some subtargets with the distinguishing features, which are grouped into a subtarget set. Every subtarget can be located independently, and the relative distances among the subtargets can be used to correct the error location results, which can enhance the adaptability of the tracking method in the complex tracking cases. When the target is occluded partially or does some complex motion, such as the scaling, translation, rotation, and pose variation, the tracking strategy can still get better tracking performance.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This study is supported by the National Natural Science Foundation of China (Project nos. 61203302 and 61403277) and the Natural Science Foundation of Tianjin (Project no. 14JCYBJC18900).

## References

- [1] M. Morshidi and T. Tjahjadi, "Gravity optimised particle filter for hand tracking," *Pattern Recognition*, vol. 47, no. 1, pp. 194–207, 2014.
- [2] R. Belaroussi and M. Milgram, "A comparative study on face detection and tracking algorithms," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7158–7164, 2012.
- [3] S.-I. Jang, K. Choi, K.-A. Toh, A. B. J. Teoh, and J. Kim, "Object tracking based on an online learning network with total error rate minimization," *Pattern Recognition*, vol. 48, no. 1, pp. 126–139, 2015.
- [4] L. Q. Li, W. X. Xie, and Z. X. Liu, "Bearings-only maneuvering target tracking based on truncated quadrature Kalman filtering," *AEU—International Journal of Electronics and Communications*, vol. 69, no. 1, pp. 281–289, 2015.
- [5] S. Brandes, Z. Mokhtari, F. Essig, K. Hünninger, O. Kurzai, and M. T. Figge, "Automated segmentation and tracking of non-rigid objects in time-lapse microscopy videos of polymorphonuclear neutrophils," *Medical Image Analysis*, vol. 20, no. 1, pp. 34–51, 2015.
- [6] C. Shen, X. Lin, and Y. Shi, "Moving object tracking under varying illumination conditions," *Pattern Recognition Letters*, vol. 27, no. 14, pp. 1632–1643, 2006.
- [7] B. J. Fan, Y. Cong, and Y. K. Du, "Discriminative multi-task objects tracking with active feature selection and drift correction," *Pattern Recognition*, vol. 47, no. 12, pp. 3828–3840, 2014.
- [8] B. Karasulu and S. Korukoglu, "Moving object detection and tracking by using annealed background subtraction method in videos: performance optimization," *Expert Systems with Applications*, vol. 39, no. 1, pp. 33–43, 2012.
- [9] S. Wang, H. Lu, F. Yang, and M.-H. Yang, "Superpixel tracking," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 1323–1330, November 2011.
- [10] J. Q. Wang and Y. Yagi, "Patch-based adaptive tracking using spatial and appearance information," in *Proceedings of the 15th IEEE International Conference on Image Processing (ICIP '08)*, pp. 1564–1567, October 2008.
- [11] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.
- [12] J. Wang and Y. Yagi, "Integrating color and shape-texture features for adaptive real-time object tracking," *IEEE Transactions on Image Processing*, vol. 17, no. 2, pp. 235–240, 2008.
- [13] G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," *Intel Technology Journal*, vol. 2, no. 2, pp. 1–15, 1998.
- [14] M. H. Yin, J. Zhang, H. G. Sun, and W. X. Gu, "Multi-cue-based CamShift guided particle filter tracking," *Expert Systems with Applications*, vol. 38, no. 5, pp. 6313–6318, 2011.
- [15] Z. W. Wang, X. K. Yang, Y. Xu, and S. Y. Yu, "CamShift guided particle filter for visual tracking," *Pattern Recognition Letters*, vol. 30, no. 4, pp. 407–413, 2009.
- [16] R. Muñoz-Salinas, E. Aguirre, M. García-Silvente, and A. Gonzalez, "A multiple object tracking approach that combines colour and depth information using a confidence measure," *Pattern Recognition Letters*, vol. 29, no. 10, pp. 1504–1514, 2008.
- [17] X. Yun and G. Xiao, "Camshift ship tracking algorithm based on multi-feature adaptive fusion," *Opto-Electronic Engineering*, vol. 38, no. 5, pp. 52–58, 2011.
- [18] J.-H. Lin, D.-H. Liu, and X.-K. Shao, "Multi-feature fusion Camshift algorithm and its further improvement," *Journal of Computer Applications*, vol. 32, no. 10, pp. 2814–2816, 2012.



