A Multiple Objects Tracking Method Based on a Combination of Camshift and Object Trajectory Tracking

Guo-wu Yuan[1(✉)], Ji-xian Zhang[1], Yun-hua Han[2], Hao Zhou[1], and Dan Xu[1]

[1] College of Information Science and Engineering, Yunnan University, Kunming 650091, China
yuanguowu@sina.com
[2] Yunnan Shui On Construction Materials Investment Holding Co. LTD, Kunming 650041, China

Abstract. Multiple objects tracking in dynamic background is one of the key techniques in computer vision. An improved method of multiple objects tracking based on a combination of Camshift and object trajectory tracking is presented in this paper. The algorithm uses Harris corner matching to estimate background movement parameters, adopts two-frame difference to detect moving objects, combines object trajectory tracking with Camshift track moving objects. Our improved algorithm can achieve satisfactory effect not only in tracking multiple objects, but also in tracking continuously the objects which are static, re-enter the current scene or recover motion. The experiments show that the improved algorithm can achieve better result in the accuracy and robustness of detecting and tracking moving objects for dynamic background.

Keywords: Multiple objects tracking · Object trajectory tracking · Camshift · Harris corner

1 Introduction

Tracking multiple objects in dynamic background is a challenging research in computer vision. It includes several key steps: background correction between adjacent frames, moving object detection, establishing the corresponding relationship between moving objects and acquiring the motion parameters of moving objects.

In recent years, more and more tracking algorithms have been proposed, for example, Kalman filtering [1,2], Camshift[3,4], particle filter [5,6]. These algorithms can achieve good effect in simple scenes. However, particle filter leads to incorrect estimation of posteriori probability because of particle degeneracy phenomenon; Kalman filtering is not suitable for real-time tracking of moving objects owing to high computational complexity; Camshift is usually used in a single object tracking and is not suitable for multiple objects tracking.

In the reference [7], the robustness of tracking in static background has been greatly improved, but the color characteristics of moving objects are lost and the algorithm

Y. Tan et al. (Eds.): ICSI-CCI 2015, Part III, LNCS 9142, pp. 155–163, 2015.
DOI: 10.1007/978-3-319-20469-7_18

is not suitable for dynamic background. In the reference [8], Camshift and Kalman filter is combined to track moving object, but it does not achieve good tracking effect in a dynamic background. The method in the reference [9] can track single moving object in dynamic background, but cannot track multiple objects in dynamic background. The reference [10] can achieve good effect in complex scenes, but fails to track when a moving object becomes static or re-enters current scene after going away. The algorithm in the reference [11] can track objects when a moving object re-enters current scene after going away, but it can not obtain satisfied effects in the complex scenes.

The above algorithms have some limitations in practical applications. In order to improve further tracking effect in dynamic background and multiple objects scene, this paper proposes an improved method of multiple objects tracking based on a combination of Camshift and object trajectory tracking. The algorithm detects foreground objects in dynamic background using frame difference after background correction, and then tracks multiple moving objects using a combination of object trajectory tracking and Camshift. It overcomes a limitation of the traditional Camshift with tracking only a single object. At the same time, it can continuously track multiple objects when the objects are static, re-enter the current scene or recover moving.

2 Moving Objects Detection for Dynamic Background

Moving objects detection is the foundation of tracking. It is more difficult to detect moving objects from dynamic background than from fixed background. Optical flow can relatively detect moving objects in dynamic background, but it has high algorithm complexity. The traditional algorithms based on frame difference and background subtraction cannot get accurate moving objects' region in a dynamic background. Frame difference is very sample, and it can achieve good effect as long as the background is corrected. Harris corner matching is used to estimate background movement parameters, and this method is described as follows.

2.1 To Estimate Background Movement Parameters

Assuming that the background's motion includes only translation and rotation, the movement can be estimated using the following affine coordinate transformation model with 6 parameters [12].

$$\begin{cases} x' = a_1 x + b_1 y + c_1 \\ y' = a_2 x + b_2 y + c_2 \end{cases} \tag{1}$$

In the formula 1, (x', y') is the corresponding pixel's coordinate in the current frame when (x, y) is a pixel coordinate in the previous frame. The 6 parameters *a1, a2, b1, b2, c1, c2* can be calculated by substituting three pairs of corresponding pixels' coordinate into the formula 1 [13].

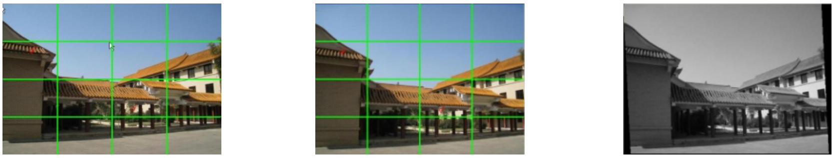
Now, the key issue is how to select the three pairs of corresponding pixels. Because Harris corner is a kind of representative feature point, it was selected as the matching points. The three pairs of Harris corners are selected according to the method in the reference [13].

2.2 To Correct the Current Frame's Background According to the Background Movement Parameters

The movement distance of background can be estimated by the formula 2 according to the background movement parameters from the above.

$$\begin{cases} \mu = x' - x \\ v = y' - y \end{cases} \tag{2}$$

According to the movement distance of background (μ, v), each pixel in the current frame can be returned to the previous frame's position. If there are no corresponding pixels in the corrected frame, the pixels' gray-scale value are filled by zero.



(a) the *k*-th frame (b) the *k+1*-th frame (c) the *k+1*-th frame corrected

Fig. 1. Correct background according to the background movement parameters

2.3 To Detect Moving Objects Using Frame Difference

After correcting background, moving objects can been detected using frame difference. Figure 2a is the result without background correction, and then Figure 2b is the result after background correction. It is shown that the moving objects detection with background correction is more accurate than one without background correction.

(a) before background correction (b) after background correction

Fig. 2. Result of moving objects detection

3 An Improved Method of Multiple Object Tracking Based on a Combination of Camshift and Object Trajectory Tracking

Multiple objects tracking algorithm can track the interested objects until the objects disappear from the scenes. In practical application, the motion states of multiple objects are diversified in video scenes. For example, an object, which stopped after entering a scene, is required to maintain tracking focus in order to be tracked continuously; an object, which recovered to move after stopping, is required to be identified and be tracked continuously; an object, which has exited from the video scene and re-entered the current scene, is required to be identified, be matched with the previous motion parameters, and be tracked continuously. At the same time, it is a challenging problem to track multiple objects effectively.

In order to meet above tracking requirements, an improved method of multiple objects tracking based on a combination of Camshift and object trajectory tracking is proposed in this paper. Object trajectory tracking algorithm can achieve satisfactory effect in tracking multiple objects, but the algorithm cannot continuously track the objects which are static, re-enter the current scene or recover motion. Camshift can track single object accurately, but can be good at tracking multiple objects.

Object trajectory tracking algorithm is described in the chapter 3.1. Because Camshift is a common algorithm in moving objects tracking, it will not go into details and can be referred to the reference [3, 4].

3.1 Algorithm Principle of Object Trajectory Tracking

After moving objects were extracted from video frames, they are matched and related with the existing paths using moving trajectory tracking [14]. If there is only one object in the scene and this object is not new one, then this object is directly related with the only existing trajectory. However, if there are multiple objects in the current scene, the correspondence between the moving objects and the moving trajectories is established through distance matrix and correlation matrix in order to track continuously moving objects. The basic steps of moving trajectory tracking algorithm are as follows.

(1) Calculate the distance matrix D_E^k

Assuming that the formula 3 indicates the centroid coordinates of n moving objects at the time k:

$$M(k) = \{M_1(k), M_2(k), \dots, M_n(k)\} \tag{3}$$

where n is the total number of the moving objects detected at the time k.

The existing m trajectories are expressed as the formula 4:

$$T(k) = \{T_1(k), T_2(k), \dots, T_m(k)\} \tag{4}$$

Then, the distance matrix between the n moving objects and the m trajectories is obtained by the formula 5.

$$D_{\mathrm{E}}^{\mathrm{k}}(i,j)=\sqrt{(T_{ix}(k)-M_{jx}(k))^2+(T_{iy}(k)-M_{jy}(k))^2} \tag{5}$$

where $(T_{ix}(k), T_{iy}(k))$ is the coordinate of the trajectory T_i at the time *k*, $(M_{jx}(k), M_{jy}(k))$ is the centroid coordinates of the object M_j at the time *k*, *i*=1,... m and *j*=1,... *n*.

(2) Calculate the correlation matrix C_E^{k}

a. Initialize all the elements in C_E^{k} by 0;

b. Location the smallest element in each row and each column, and the smallest element position is respectively shown by the vector $\alpha=\{\alpha_1,..\alpha_m\}$ and $\beta=\{\beta_1,\ldots\beta_n\}$. The two vectors must be accorded with the formula 6:

$$\begin{cases} D_E^k(i,\alpha_i)=\min\limits_{1\le j\le n}(D_E^k(i,j)) \\ D_E^k(\beta_j,j)=\min\limits_{1\le i\le m}(D_E^k(i,j)) \end{cases}. \tag{6}$$

c. The corresponding position elements of the correlation matrix C_E^{k} are added by 1 according to the formula 7;

$$\begin{cases} C_E^k(i,\alpha_i)=C_E^k(i,\alpha_i)+1, i=1,\ldots,m \\ C_E^k(\beta_j,j)=C_E^k(\beta_j,j)+1, j=1,\ldots,n \end{cases} \tag{7}$$

(3) Establish the relationship between the moving objects and the trajectories

The elements $C_E^{\mathrm{k}}(i,j)$ have three possible values: 0, 1 or 2. If $C_E^{\mathrm{k}}(i,j)=0$, it is indicated that the *i-th* trajectory and the *j-th* moving object are not matched with each other; If $C_E^{\mathrm{k}}(i,j)=1$, it is indicated that only one of them is matched with the other; If $C_E^{\mathrm{k}}(i,j)=2$, it is indicated that the trajectory and the moving object are matched with each other. Therefore, the whole matrix C_E^{k} may have three possible cases:

a. If the *i-th* row has several nonzero elements, it is indicated that the *i-th* trajectory matches with several moving objects. Two types are divided again as follows:

- if the *i-th* row has a value of 2, for the other $C_E^{\mathrm{k}}(i,j)=1$, then the *j-th* moving object is a new object entered into the scene;
- Otherwise, if the *i-th* row has only several value of 1 and assuming $C_E^{\mathrm{k}}(i,j)=1$, it is divided into two cases:

  i) If the *j-th* column has an element with a value of 2, the *i-th* trajectory is a trajectory away from the scene;

  ii) If the *j-th* column has not an element with a value of 2, the *j-th* moving object is a new object entered into the scene.

b. If the *j-th* column has several nonzero elements, it is indicated that the *j-th* moving object matches with several trajectory. Two types are divided again as follows:

- if the *j-th* column has a value of 2, for the other $C_E^{\mathrm{k}}(i,j)=1$, then the *i-th* trajectory is a trajectory away from the scene;

- Otherwise, if the the *j-th* column has only several value of 1 and assuming $C_E^k(i,j)=1$, the *j-th* moving object is a new object entered into the scene.

c. If $C_E^k(i,j)=2$, it is indicated that the *i-th* trajectory matches strictly with the *j-th* moving object. Two types are divided again as follows:

- if $C_E^k(i,j)=1$ and $D_E^k(i,j)\le T_0$ (T_0 is the maximum displacement of a moving object between adjacent frames), the *i-th* trajectory is related with *j-th* moving object;
- Otherwise, if $C_E^k(i,j)=1$ and $D_E^k(i,j)>T_0$, it is considered that the *i-th* trajectory has a false relation with the *j-th* moving object. It is possible that the *i-th* trajectory, which is related to the object away from the previous frame, is associated with the *j-th* moving object which enters into the current frame.

3.2 Algorithm Combined Camshift with Object Trajectory Tracking

The combination of Camshift and moving trajectory tracking can achieve better tracking effect. When the objects are moving in the scene, they are tracked using the trajectory tracking method; and when the objects are motionless, have moved to the outside of current scene, or re-enter the scene, trajectory tracking algorithm will lose the focus of moving objects, but the moving objects can be tracked continuatively by searching in a limited range using Camshift algorithm and the tracking information before the focus is lost. If the moving objects is still unable to be tracked using Camshift in a limited range, they are considered to move out of the scene. In addition, when some objects recover motion and are tracked using Camshift from static state, the Camshift will be stopped, and the objects are tracked again using trajectory tracking method after retrieving the object index.

In order to facilitate analyzing, the storage structure of moving objects $Area\ \{rect, object, trace, old\}$ is defined, where $Area_{\,i}(k)$ represents the region of the *i-th* object in the *k-th* frame, *rect* represents the rectangle of moving object, *object* represents the index of moving object, *trace* represents the trajectory of moving object, and *old* represents the aging ratio. Then the steps of the combination algorithm of trajectory tracking and Camshift are described as follows:

(1) The *m* moving objects are detected using the movement parameters estimation of background;

(2) For the *m* moving objects detected, the moving regions are matched with the trajectory of moving object in the previous frame, and then the regions matched successfully are marked with an index. According to each moving object's centroid position in the current frame, the trajectory of moving object corresponding to the index is updated. At the same time, if there are the objects which enter or exit from the current frame, they are stored.

(3) For the new object entered the scene, the object region is matched and compared with all regions tracked using Camshift in order to judge whether it is a previous object re-entered the scene; if it is a previous object, Camshift tracking is aborted, it is numbered using the previous index retrieved, the moving object's record

is updated, and this algorithm goes to the step 5; if it is not a previous object but is a whole new one, this algorithm goes to the step 4.

(4) For a whole new object, the corresponding parameters are added to the structure record of moving objects according the next formula.

$$\begin{cases} Area_n(0)\text{-}>object = n \\ Area_n(0)\text{-}>rect = \mathrm{Re}ct_n(0) \end{cases}. \tag{8}$$

where, n represents the total number of the moving object in the current frame. If there is an object that has not been processed yet, this algorithm goes to the step 3 to continue processing. If the objects have processed, this algorithm goes to the step 5.

(5) For the moving objects that are still not matched in the previous frame, in order to avoid the individual moving objects disappear temporarily, the objects are added to the list of the moving objects area, and the aging value $Area_i(n)\text{-}>old$ is increased by 1. If the aging value does not exceed a beforehand threshold, the object is considered to loss temporarily tracking focus in the scene, and this algorithm goes to the step 6. If the aging value exceeds a beforehand threshold, the object is considered to leave from the scene and is cleared from the objects' record.

(6) For the object that loss temporarily tracking focus in the current scene, its motion region parameters will be transferred to Camshift; then, the moving object's color probability distribution is calculated, the object's feature template is obtained, and the matched object is searched in the nearby area; if the matched object is obtained, the matched object is regarded as static state in the current frame; at this time, the information of motion area will be transmitted to Camshift, and the static object is locked and tracked using Camshift in subsequent frames until it regains moving. If the matched object is not found, the object is regarded to leave from the scene, and the motion parameters are temporarily stored in order to prevent re-enter the scene in the subsequent frames.

(7) If there is an object that is not matched and is not processed, this algorithm goes to the step 5.

(8) All trajectories of moving object are updated.

4 Experiments and Analysis

In the programming environment with VC++ and OpenCV, our improved algorithm is verified using our videos shot outdoors, in which there are multiple objects and complex moving background with a video resolution of 640 x 480. Experimental results show that multiple objects in the scene, including moving objects, static objects, renewing moving objects, the objects re-entered the scene and the objects left from the scene, are tracked robustly and accurately.

Figure 3 shows the tracking effect using Camshift. We find that Camshift tracks only a single object and is inaccurate in the video.

Figure 4 shows the tracking effect using trajectory tracking. We find that a static object is lost to track in the scene, at the same time, and an object re-entered the scene is defined as a new object to continue tracking, which is not reasonable.

Figure 5 shows the tracking effect using our improved method. We can see that our method has more satisfactory tracking effect compared with alone Camshift and trajectory tracking algorithms, can track multiple objects accurately and can correctly track a static object and an object re-entered the scene.

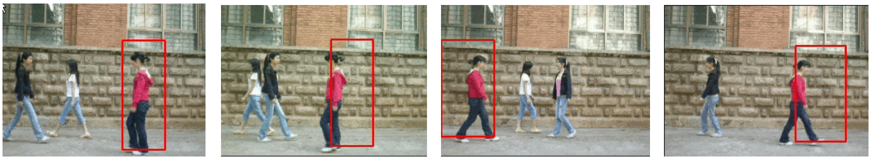


Fig. 3. The tracking effect using Camshift

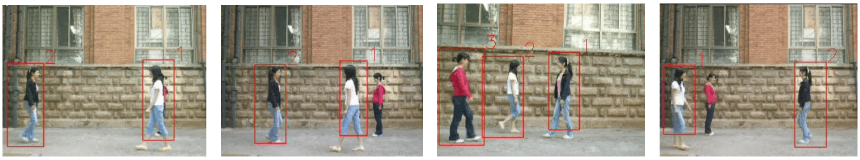


Fig. 4. The tracking effect using trajectory tracking [14]

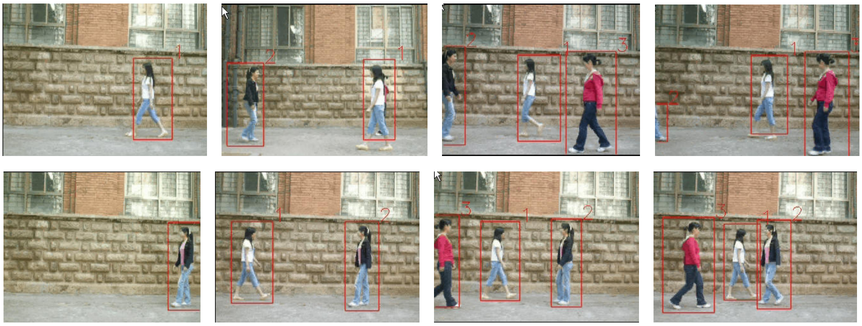


Fig. 5. The tracking effect using our improved method

From the above experimental results, we can see, Camshift and trajectory tracking algorithm based on combination of paper made reference [14] more accurate, more ideal tracking effect, which not only realizes the Camshift and accurate tracking of multiple targets, but also can effectively to a stationary target, in the scene again into the target, and then moving target and exit the scene target effectively tracking, meet the requirement of the practical application of the tracking.

5 Conclusion

In the paper, through analyzing the advantages and disadvantages of Camshift and object trajectory tracking, we discover that the two methods just can well overcome each other's shortcomings. Therefore, this paper combines effectively the two methods so as to realize multiple objects tracking in complex and dynamic scenes.

The experiments show that the improved algorithm can achieve better result in detecting and tracking moving objects for dynamic background.

Acknowledgments. This work is supported by the Natural Science Foundation of China (61163024, 61262067), the Application and Foundation Project of Yunnan Province (2011FB019, 2013FB010) and the Young Teachers Training Program of Yunnan University.

References

1. Ristic, B., Arulampalam, S., Gordon, N.: Beyond the Kalman Filter: Particle Filters for Tracking Applications. Artech House, Boston (2004)
2. Moussakhani, B., Flam, J.T., Ramstad, T.A., Balasingham, I.: On change detection in a Kalman filter based tracking problem. Signal Processing **105**, 268–276 (2014)
3. Dorin, C., Visvanathan, R., Peter, M.: Kernel-based object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(5), 564–577 (2003)
4. Hong-zhi, Z., Jin-huan, Z., Hui, Y., Shi-lin, H.: Object Tracking Algorithm Based on CamShift. Computer Engineering and Design **27**(11), 108–110 (2006)
5. Nummiaro, K., Koller-Meier, E., Van Gool, L.: Object tracking with an adaptive color-based particle filter. In: Van Gool, L. (ed.) DAGM 2002. LNCS, vol. 2449, pp. 353–360. Springer, Heidelberg (2002)
6. Rui, T., Zhang, Q., Zhou, Y., Xing, J.C.: Object tracking using particle filter in the wavelet subspace. Neurocomputing **119**(7), 125–130 (2013)
7. Nouar, O.D., Ali, G., Raphael, C.: Improved object tracking with Camshift algorithm. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 657-660. IEEE, Piscataway (2006)
8. Kai, S., Shi-rong, L.: Combined algorithm with modified Camshift and Kalman Filter for multi-object tracking [J]. Information and Control **38**(1), 11–16 (2009)
9. Yuan, G.-w., Gao, Y., Dan, X.: A Moving Objects Tracking Method Based on a Combination of Local Binary Pattern Texture and Hue. Procedia Engineering, Elsevier **15**, 3964–3968 (2011)
10. Xin-jun, S., Jin-bo, C., Fa-wei., X.: Multiple Targets Tracking in Traffic Image Sequence. Journal of Changshu Institute Technology (Nature Sciences) **23**(4), 82–86 (2009)
11. Xue, L., Fa-liang, C., Hua-jie, W.: An Object Tracking Method Based on Improved Camshift Algorithm. Microcomputer Information **23**(21), 304–306 (2007)
12. Zhang, L.Y., Zhang, G.L.: Background motion model parameters estimation based on extended Kanade-Lucas tracker. Computer Applications **25**(8), 1946–1947 (2005)
13. Yan, X.L., Liang, B., Zeng, G.H.: Object tracking method based on block motion estimation. Journal of Image and Graphics **13**(10), 1869–1872 (2008)
14. Xin-jun, S., Jin-bo, C., Fa-wei, X.: Multiple Targets Tracking in Traffic Image Sequence. Journal of Changshu Institute Technology (Nature Sciences) **23**(4), 82–86 (2009)