A Novel Probabilistic Projection Model for Multi-Camera Object Tracking

Jiaxin Lin[1,2], Chun Xiao[1], Disi Chen[2], Dalin Zhou[2], Zhaojie Ju[2], Honghai Liu[2]

1.School of Automation，Wuhan University of Technology，Wuhan，430070，China
2.Intelligent Systems and Biomedical Robotics Group, School of Computing, University of Portsmouth
tiao_ju@163.com

Abstract. Correlation Filter (CF)-based algorithms have achieved remarkable performance in the field of object tracking during past decades. They have great advantages in dense sampling and reduced computational cost due to the usage of circulant matrix. However, present monocular object tracking algorithms can hardly solve fast motion which usually causes tracking failure. In this paper, a novel probabilistic projection model for multi-camera object tracking using two Kinects is proposed. Once the object is found lost using multimodal target detection, the point projection using a probabilistic projection model is processed to get a better tracking position of the targeted object. The projection model works well in the related experiments. Furthermore, when compared with other popular methods, the proposed tracking method grounded on the projection model is demonstrated to be more effective to accommodate the fast motion and achieve better tracking performance to promote robotic autonomy.

Keywords: object tracking, correlation filter, multi-camera, multimodal target detection, projection

1 Introduction

Object tracking plays an important role in the field of computer vision. It can be used in video surveillance, human-robot interaction, augmented reality, etc. While a great many tracking algorithms have been proposed and improved constantly, challenges caused by background clutter, fast motion, occlusion and other factors still exist.

In recent years, a lot of research works focus on two kinds of trackers: correlation filter-based tracker and deep learning-based tracker. Correlation filters use circulant matrix to conduct dense sampling and improve computational efficiency by exploring matrix theory and kernel trick in the frequency domain [9, 10]. Correlation filter-based methods are steadily improved from aspects of color information [4], background information [10], unwanted boundary effects [5] and scale variation [3, 6]. Deep learning-based trackers, such as Modeling and Propagating CNNs in a Tree

* Supported by grant of the EU Seventh Framework Programme (Grant No. 611391), National Natural Science Foundation of China (Grant no. 51575412, 51575338 and 5157540).

Structure for Visual Tracking (TCNN) [12] and Multi-Domain Convolutional Neural Networks (MDNET) [17], succeed in getting high-level semantic information while the heavy computational load constrains their development.

Even though many nontrivial works have been done, there are still certain flaws. Almost all present trackers are based on monocular camera systems and recognized as single camera tracking (SCT). This indicates that we can get the information about the object from only one perspective. Once the object moves faster than the tracker can accommodate, total tracking loss may occur. In order to alleviate or further solve the problem, this paper proposes a novel probabilistic projection model for multi-camera object tracking based on two Kinects. The fast discriminative scale space tracking (fDSST) algorithm is conducted separately in two simultaneous videos. After each train-detection cycle, we use multimodal target detection to estimate whether the object is lost or not. If loss happens, the probabilistic point projection derived from the information of the other Kinect is adopted to choose a better tracking posi-tion.

The structure of this paper is as followed. Section 2 demonstrates the related work including a brief description of CF-based trackers, inter-camera tracking (ICT) and probabilistic models used in robotic area. Section 3 gives the formulation about fDSST algorithm. Section 4 gives probabilistic projection model for two Kinects. Section 5 proposes the framework of multi-camera tracking. Experiments are described in section 6 followed by a conclusion in section 7.

2 Related Work

Object tracking algorithms using correlation filter can be traced back to the article [1] published in 2010 by David S. Bolme et al. The basic idea is that the similarity of two functions can be revealed by their cross-correlation. The filter is called Minimum Output Sum of Squared Error (MOSSE) filter and its tracking speed reaches 669 frames per second (fps). It is well known that better tracking performance requires more samples. But computation of extensive samples will reduce efficiency. This problem is solved in the literature [10], where João F. Henriques et al. speed up the computing by using the matrix property of circulant matrix in the frequency domain. In case of dense sampling, it still retains a speed of more than 300 fps. Then Dual Correlation Filter (DCF) which expands the one-dimensional grayscale feature into multi-channel features and further advances the tracking results is proposed in [9]. Martin Danelljan introduces color features[4] into correlation filter on this basis. The color information is divided into 11 directions, so that the image information input of this algorithm is richer than the previous grayscale features and histogram of oriented gradients (HOG) features. The Spatially Regularized Discriminative Correlation Filters (SRDCF) algorithm is an extension of DCF. The DCF suffers the boundary problem caused by cyclic samples. Accordingly, the regularization term (originally is a con-stant multiplied by the filter) is improved in [5] and replaced by a position-dependent function multiplied by the filter. The Continuous Convolution Operator Tracker (C-COT) [6] uses the correlation filter and adopts an implicit interpolation model to pose the learning problem in the continuous spatial relation and thus obtains

a continuous response function to achieve sub-pixel localization. Although C-COT won the cham-pionship in VOT2016, its performance is far from complying with real-time con-straint. Therefore, in 2017, the Efficient Convolution Operators (ECO) [3] for tracking as an accelerated version of C-COT is published. It improves C-COT from three main aspects: the dimension of the original feature channel is reduced; the generation model is improved; the update mechanism is advanced.

Presently, multi-camera tracking applications mainly concentrate on pedestrian detection, traffic monitoring, smart rooms, etc. [2,8]. In these applications, multiple targets are often tracked at the same time, and the perspectives do not or partially overlap between cameras in order to obtain a larger field of view.

The problem of coordinate conversion between two-dimensional point in camera pixel coordinate system and three-dimensional point rises. It is mentioned in the literature [13] that the traditional stereo dual-camera system requires the distance between two cameras within a certain range. Only in this way can the two simultaneously obtained images of the same scene have enough matching points that the depth information can be solved correctly and accurately. The second Kinect in this article is chosen to shoot at another angle in order to obtain more information. So, it is impossible to reconstruct the same scenario by simply using traditional binocular vision method. The problem of obtaining three-dimensional coordinates from two-dimensional images is considered in robotic grasping with a probabilistic model in [14]. In [14], two images of the same object are captured from different positions by the same camera to obtain the three-dimensional coordinates of grab point. The published error is within 4cm. In this paper, considering that Kinect can get depth information, the probabilistic model is used to obtain the three-dimensional coordinates of the target point.

3 A Brief Description of fDSST

The target sample x consists of a *d*-dimensional feature vector $x(n) \in R^d$, at each location n in a rectangular domain. We denote the feature channel $l \in \{1,...,d\}$ of x by x^l. The objective is to construct a correlation filter h consisting of one filter h^l per feature channel. This is achieved by minimizing the L^2 error of the correlation response compared to the desired correlation output *g*,

$$\varepsilon = \left\| g - \sum_{l=1}^{d} h^l \odot x^l \right\|^2 + \lambda \left\| h^l \right\|^2 . \tag{1}$$

Here, $\odot$ denotes circular correlation. The second term in (1) is a regularization with a weight parameter λ.

The filter that minimizes (1) is given by

$$H^l = \frac{\bar{G}X^l}{\sum_{k=1}^{d} \bar{X}^k X^k + \lambda}, l = 1, ..., \mathrm{d}. \tag{2}$$

Here, the capital letters denote the discrete Fourier transform (DFT) of the corresponding quantities. The bar denotes complex conjugation.

The numerator A_t^l and denominator B_t^l of the filter H_t^l with a new sample x_t are defined as follows

$$A_t^l = (1-\eta) A_{t-1}^l + \eta \bar{G} X_t^l, l = 1, ..., d \tag{3a}$$

$$B_t^l = (1-\eta) B_{t-1}^l + \eta \sum_{k=1}^{d} \bar{X}_k^t X_k^t \tag{3b}$$

Here, the scalar η is a learning rate parameter. The DFT of the correlation scores y_t is computed in the frequency domain

$$Y_t = \frac{\sum_{l=1}^{d} \bar{A}_{t-1}^l Z_t^l}{B_{t-1} + \lambda}. \tag{4}$$

The test sample z_t is extracted using the same feature representation of training samples. The estimate of the current target state is obtained by finding the maximum correlation score.

Based on multi-channel discriminative correlation filters given above, fDSST learns separate discriminative correlation filters for translation and scale estimation and reduces the feature dimension using Principal Components Analysis (PCA).

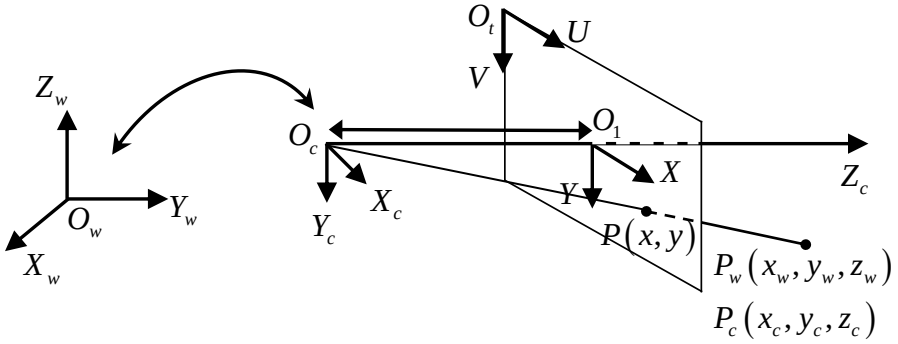

Fig. 1. Linear model of a camera

4 Probabilistic Projection Model between Two *Kinect*s

4.1 Projection Theory and Coordinate System Conversion

An ideal imaging model for a camera is normally linear for both simplicity and efficiency. The principle of linear imaging is depicted in Figure 1.

The target point $P_w\left(P_c\right)$, the point P on the image plane and the optical center of the camera O_c are collinear. An imaginary world coordinate system $O_w - X_w Y_w Z_w$ is used to describe the position information of objects in space. $O_c - X_c Y_c Z_c$ denotes the camera coordinate system. The optical center of the camera O_c is defined as the origin of the coordinate system. The distance from the optical center to the image plane is f, the focal length of the camera. The pixel coordinate system is expressed as $O_t - UV$. The upper left corner of the image is the origin and the basic unit of this coordinate system is pixel. Each image pixel is actually a small rectangle and its physical size is recorded as dx, dy. $O_1 - XY$ is the physical coordinate system which takes the center point of the image as the origin and millimeter as the basic unit. The coordinates of O_1 in the $O_t - UV$ coordinate system is $\left(u_0, v_0\right)$. It is assumed that the coordinates of any spatial point in the world coordinate system and camera coordinate system can be respectively expressed as $\left(x_w, y_w, z_w\right)$ and $\left(x_c, y_c, z_c\right)$. The conversion between the coordinates of the same spatial point in the world coordinate system and the pixel coordinate system of the image can be derived [18]:

$$\begin{aligned} z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} &= \begin{bmatrix} f/dx & 0 & u_0 & 0 \\ 0 & f/dy & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} f/dx & 0 & u_0 & 0 \\ 0 & f/dy & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \end{aligned} \tag{5}$$

For camera0, R_0, T_0 respectively represent the rotation matrix and translation vector of the camera coordinate system relative to the space coordinate system. And for camera1, R_1, T_1 represent the same parameters. Hence, we can get:

$$\begin{bmatrix} x_{c1} \\ y_{c1} \\ z_{c1} \end{bmatrix} = R_0 \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + T_0 \qquad \begin{bmatrix} x_{c2} \\ y_{c2} \\ z_{c2} \end{bmatrix} = R_1 \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + T_1 \tag{6}$$

Therefore, the rotation matrix and translation vector between two camera coordinate systems can be obtained by:

$$\begin{aligned} R_c &= R_0^{-1} R_1 \\ T_c &= R_0^{-1} \left(T_0 - T_1 \right) \end{aligned} \tag{7}$$

4.2 Probabilistic Projection Model for Two *Kinect* System

From section 4.1 and section 4.2, the coordinates (u, v) in the pixel coordinate system of the image can be easily obtained if the coordinates (x_w, y_w, z_w) in the world coordinate system of the correspondent spatial point are known. Since two *Kinect*s are used as image sensors in this paper, RGB images and depth information can be attained at the same time which enables us to get the 3-D coordinates of certain point in the 2-D image. In this way, the coordinates of this 3-D point can be projected to the coordinates in the pixel coordinate system of any other calibrated camera.

Here, the problem is how to get proper and accurate information of the coordinates in the world coordinate system. Since the rectification between RGB and depth images from a *Kinect* usually leads to some non-numeric points in the rectified depth images, we apply a novel probabilistic model for the depth offered by depth image.

It is assumed that, even though error exists in the rectified RGB and depth images, the coordinates of depth value closer to the chosen RGB point (u, v) are more likely to be the true depth value of the point. Then the two-dimensional Gaussian distribution is used to represent the possibility of the depth value at certain positions to be the true depth value, which can be written in:

$$P\left(z(\hat{u},\hat{v}) = 1 \mid D\right) = \left(2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}\right)^{-1} \exp\left[-\frac{1}{2\left(1-\rho^2\right)}\left(\frac{(\hat{u}-u)^2}{\sigma_1^2} - \frac{2\rho(\hat{u}-u)(\hat{v}-v)}{\sigma_1\sigma_2} + \frac{(\hat{v}-v)^2}{\sigma_2^2}\right)\right], \tag{8}$$

where D denotes the depth image, $D(u, \mathrm{v})$ denotes the depth value directly get from the coordinates (u, v) in the depth image D, $\hat{u}, \hat{\mathrm{v}}$ are random variables obeying two-dimensional normal distribution and can be expressed as:

$$\hat{u}, \hat{\mathrm{v}} \sim \mathrm{N}\left(u, v, \sigma_1^2, \sigma_2^2, \rho\right). \tag{9}$$

$z(\hat{u},\hat{\mathrm{v}})=1$ denotes that $D(\hat{u},\hat{\mathrm{v}})$ is the true depth value of the point (u,v) in RGB image, and $z(\hat{u},\hat{\mathrm{v}})=0$ otherwise. Then the expectation of the true depth value of the point (u,v) in RGB image can be calculated in:

$$E(D(u,v))=\sum_{\hat{u},\hat{\mathrm{v}}} D(\hat{u},\hat{\mathrm{v}})P\left(z(\hat{u},\hat{\mathrm{v}})=1\,|\,D\right). \tag{10}$$

Noted that, for certain coordinates $(\hat{u},\hat{\mathrm{v}})$, if the value $D(\hat{u},\hat{\mathrm{v}})$ equals 0 (out of *Kinect* detection range) or is non-numeric, we specially define

$$P\left(z(\hat{u},\hat{\mathrm{v}})=1\,|\,D\right)=0. \tag{11}$$

And this will result in

$$\sum_{\hat{u},\hat{\mathrm{v}}} P\left(z(\hat{u},\hat{\mathrm{v}})=1\,|\,D\right)<1 \tag{12}$$

and the calculated depth will be smaller than the actual value.

We remedy this problem by changing the expectation equation like

$$E(D(u,v))=\left(\sum_{\hat{u},\hat{\mathrm{v}}} D(\hat{u},\hat{\mathrm{v}})P\left(z(\hat{u},\hat{\mathrm{v}})=1\,|\,D\right)\right)\Big/\sum_{\hat{u},\hat{\mathrm{v}}} P\left(z(\hat{u},\hat{\mathrm{v}})=1\,|\,D\right) \tag{13}$$

5 Multi-Camera Object Tracking Framework

Our multi-camera object tracking method runs fDSST separately on each *Kinect*. The difference between traditional tracker and ours is that the multimodal target detection [15] is introduced as the criterion for judging whether the target is lost or not and when the projection process is needed. For example, when the camera0 finds that the ratios between multiple peaks to the highest peak of the correlation scores $y_{t,\mathrm{trans}}$ are greater than a predefined threshold θ, the loss of the targeted object is inferred. Then, the coordinates $proj_t$ projected from the other camera (camera1) to the pixel coordinate system of camera0 is calculated. The algorithm will re-detect the response $yproj_{t,\mathrm{trans}}$ at that point, compare its maximum with that of $y_{t,\mathrm{trans}}$ and choose the higher one for the translation vector calculation. The original fDSST algorithm is followed subsequently to estimate the scale and update translation filter model and scale filter model.

6 Experimental Results

6.1 Experiments for Projection

Based on the theories in Section 4, MATLAB stereo calibration app is used to get inner parameters and inter parameters between two *Kinect*s. To ensure the effectiveness and robustness of our projection method, experiment is conducted on 10 pairs of random matching points. Firstly, the actual coordinates of each pair of points are labelled manually. Then, the Euclidean distance between the actual coordinates and the calculated coordinates is used as error criterion. The results are compared with that of traditional projection methods. All specific data is shown in table 1 and is in pixels.

Coordinates get manually		Traditional Projection results	Distance(error)	Probabilistic Projection results	Distance(error) of Our Mathod
Camera0	Camera1				
(136,246)	(385,377)	NaN	NaN	(136,241)	4.53
(147,225)	(409,418)	(139,218)	9.92	(136,218)	12.35
(144,221)	(407,428)	(130,213)	15.69	(131,213)	14.84
(142,211)	(407,438)	(126,209)	15.57	(153,206)	12.35
(405,304)	(130,295)	(400,308)	6.36	(400,307)	5.70
(121,251)	(385,379)	(122,141)	10.11	(123,241)	10.31
(188,271)	(517,345)	NaN	NaN	(184,258)	12.98
(448,238)	(191,356)	(443,241)	6.10	(443,241)	6.10
(402,305)	(133,294)	(402,307)	2.55	(402,307)	2.55
(304,416)	(115,220)	(299,416)	5.02	(299,416)	5.02
RMS of distance (excluding data contains NaN)			9.99		9.57
RMS of distance (all data)			NaN		9.60

Table 1. Projection Results of Traditional Method and Ours

From the table above, there are two non-numeric results represented by *NaN* after using traditional projection method. This is directly due to the non-numeric depth value at the point in Camera0. As for our probabilistic projection method, the problem of invalid depth value caused by rectification of *Kinect* is completely avoided. The projection result for every point is achieved successfully. In order to compare results quantitively, the root mean square (RMS) of error, excluding the invalid results, of traditional projection method is 9.99 pixels while the correspondent one of our probabilistic method is 9.57 pixels. This proves that our method has a slight improvement in accuracy and is much more robust compared to the traditional projection method.

Since the distance between projection results and true positions of our method maintains within 15 pixels, it can certainly meet the tracking requirement according to the common size of target which is bigger than 40 *40 pixels. With such acceptable error range, the projection result will be within the target window.

6.2 Benchmark Building and Evaluation

To validate the effectiveness of our proposed tracking method, a dataset including 4 pairs of video sequences is built by two *Kinect*s sensors. Each pair of sequence has two RGB videos and two sets of depth information. All target positions at each frame in RGB images are annotated using rectangles. The manual annotations are treated as the ground-truth to evaluate the methods performance. Each of the recorded video sequences faces several challenges such as occlusion, out-of-plane rotation, in-plane rotation, out-of-view, background clutter, deformation and scale variation. Some of the frames are illustrated in Fig.2.

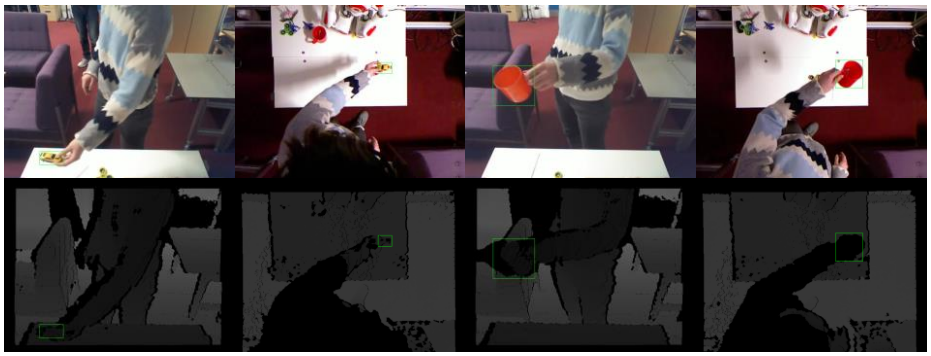

Fig. 2. Demonstration of video sequences in our datasets

The data were captured by two *Kinect* v1.0 cameras. The resolution of output RGB image is 480 *640 pixels and the depth information is rectified to the same size as the RGB image. The frame rate is 25 fps.

All tracking parameters are set the same as in [7]. As for multimodal target detection, the parameter is set as [15].

Since this is a real-time tracking system, the comparison confined within the results of several algorithms with high speed like CN [4], KCF [9], CA-SAMF [11], CA -DCF [11] and fDSST [7].

Two tracking evaluation metrics, success rate and precision rate proposed in [16] are employed with one-pass evaluation (OPE). The success rate measures the intersection over union (IoU) between the ground-truth and tracking results. Area Under the Curve (AUC) of success plots is used to rank the trackers. The precision rate is defined as the distance error between the estimated target center and the ground-truth. Trackers are ranked in terms of the distance error at a threshold of 20 pixels.

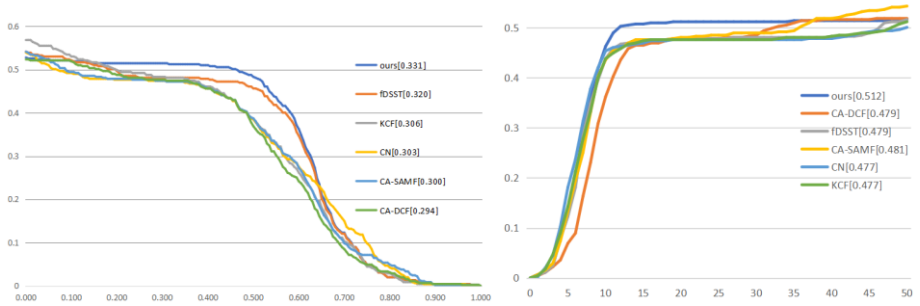

Fig. 3. The success rate plot and the precision plot

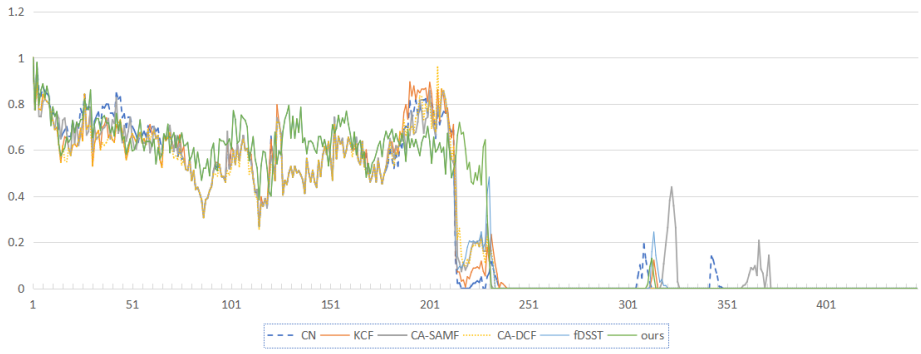

Fig. 4. Overlap curve of the sequence 'toy car' from camera0

6.3 Empirical results

Because of the challenging characteristics of our locally captured videos, all trackers fail to track the object without intervention. Some failure instances are caused by occlusion while others are mostly affected by fast motion. It is promising to see that the proposed method can alleviate these problems to some extent, reflected by the success rate plot and the precision plot shown in Fig.3. Compared with all the other trackers, our tracker ranks the first place in both evaluation plots.

As for precision plot, our method outperforms other algorithms from very beginning. Among all algorithms for comparison, KCF is proposed earliest and is often used as a baseline. CN tracker puts the color information into consideration. When all trackers can catch up with the object, this tracker can easily figure out background and choose a better center of the target window. The CA-SAMF and CA-DCF use the context information as negative samples to cope with the challenge of cluttered background. While they may not get the most accurate position of the target because of the negative effect of certain context, they can get better tracking performance than KCF tracker overall. Our method is based on the fDSST algorithm. The fDSST is also a correlation filter tracker that especially joints a scale correlation filter with a translation filter. When the scale of the object changes, our method can change the window

size for better adaptation under the incorporated background information. As a result, our method achieves better performance than the rest.

In the plot of success rate, our method outperforms significantly at the overlap rate of 0.5. This means that our method catches up with the object in more frames than other algorithms. This result also owes to the advantages of fDSST algorithms, since the scale space filter can improve the overlap effectively. And it is intuitive that the projection strategy assists in catching up with a fast moving object.

In Fig.4, the overlap curve of the sequence 'toy car' from camera0 is demonstrated. Near the 203rd frame, there is an obvious drop of overlap for all other algorithms, while an overlap of about 0.5 remains for the following 40 frames by using our method. The video is finally checked manually and we find that there is a frame where the object moves so fast and changes its appearance so rapidly at the same time, that none algorithm succeeds to catch up with it. The projected coordinates used in our method successfully update the model after the multimodal target detection works and finish the tracking task.

7 Conclusion and Discussion

In this paper, a novel probabilistic projection model for multi-camera object tracking based on two *Kinect*s is proposed. The multimodal target detection is combined with the multi-camera object tracking method and further fuses information from two *Kinect*s by projecting points between them. The probabilistic projection model achieves satisfactory experimental results with robustness and it can be further used in robotic vision and control to promote autonomy of robots or robotic systems. Experimental results also demonstrate that our method outperforms other algorithms and helps alleviate the problem of tracking during fast motion. However, there is still improvement to be done such as fusing the information from two *Kinect*s to predict out-of-view in the future work.

References

1. Bolme, D. S., Beveridge, J. R., Draper, B. A., & Lui, Y. M. (2010, June). Visual object tracking using adaptive correlation filters. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on (pp. 2544-2550). IEEE.
2. Chen, W., Cao, L., Chen, X., & Huang, K. (2014, October). A novel solution for multi-camera object tracking. In Image Processing (ICIP), 2014 IEEE International Conference on (pp. 2329-2333). IEEE.
3. Danelljan, M., Bhat, G., Khan, F. S., & Felsberg, M. (2017, July). ECO: Efficient Convolution Operators for Tracking. In CVPR (Vol. 1, No. 2, p. 7).
4. Danelljan, M., Shahbaz Khan, F., Felsberg, M., & Van de Weijer, J. (2014). Adaptive color attributes for real-time visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1090-1097).

5. Danelljan, M., Hager, G., Shahbaz Khan, F., & Felsberg, M. (2015). Learning spatially regularized correlation filters for visual tracking. In Proceedings of the IEEE International Conference on Computer Vision (pp. 4310-4318).
6. Danelljan, M., Robinson, A., Khan, F. S., & Felsberg, M. (2016, October). Beyond correlation filters: Learning continuous convolution operators for visual tracking. In European Conference on Computer Vision (pp. 472-488). Springer, Cham.
7. Danelljan, M., Häger, G., Khan, F. S., & Felsberg, M. (2017). Discriminative scale space tracking. IEEE transactions on pattern analysis and machine intelligence, 39(8), 1561-1575.
8. Ge Dongyuan, Yao Xifan, & Li Kainan. (2010). Calibration of Binocular Stereo Vision System. Mechanical Design and Manufacturing, 6(6), 1-2.
9. Henriques, J. F., Caseiro, R., Martins, P., & Batista, J. (2015). High-speed tracking with kernelized correlation filters. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(3), 583-596.
10. Henriques, J. F., Caseiro, R., Martins, P., & Batista, J. (2012, October). Exploiting the circulant structure of tracking-by-detection with kernels. In European conference on computer vision (pp. 702-715). Springer, Berlin, Heidelberg.
11. Mueller, M., Smith, N., & Ghanem, B. (2017, July). Context-aware correlation filter tracking. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Vol. 2, No. 3, p. 6).
12. Nam, H., Baek, M., & Han, B. (2016). Modeling and propagating cnns in a tree structure for visual tracking. arXiv preprint arXiv:1608.07242.
13. Saxena, A. (2009). Monocular depth perception and robotic grasping of novel objects. STANFORD UNIV CA DEPT OF COMPUTER SCIENCE.
14. Saxena, A., Driemeyer, J., & Ng, A. Y. (2008). Robotic grasping of novel objects using vision. The International Journal of Robotics Research, 27(2), 157-173.
15. Wang, M., Liu, Y., & Huang, Z. (2017, July). Large margin object tracking with circulant feature maps. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA (pp. 21-26).
16. Wu, Y., Lim, J., & Yang, M. H. (2015). Object tracking benchmark. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(9), 1834-1848.
17. Zhang, Z., Xie, Y., Xing, F., McGough, M., & Yang, L. (2017). Mdnet: A semantically and visually interpretable medical image diagnosis network. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 6428-6436).
18. Zhang, Z. (2000). A flexible new technique for camera calibration. IEEE Transactions on pattern analysis and machine intelligence, 22.