IET Image Processing

Research Article

IET Journals
The Institution of Engineering and Technology

Aircraft tracking based on fully conventional network and Kalman filter

ISSN 1751-9659
Received on 11th February 2018
Revised 30th October 2018
Accepted on 3rd December 2018
E-First on 21st May 2019
doi: 10.1049/iet-ipr.2018.5022
www.ietdl.org

Jiachen Yang[1], Weirong Zhao[1], Yurong Han[1] ✉, Chunqi Ji[1], Bin Jiang[1], Zhihui Zheng[2], Houbing Song[3]

[1]School of Electrical and Information Engineering, Tianjin University, Tianjin, People's Republic of China

[2]Beijing Aerospace Automatic Control Institute, Beijing, People's Republic of China

[3]Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114, USA

✉ *E-mail: hanyurong@tju.edu.cn*

Abstract: Aircraft tracking is a significant technology for military reconnaissance, but there is no efficient algorithm to solve this particular problem. Recently, research based on deep learning for object tracking has developed rapidly, and the performance is greatly improved compared to the traditional methods, so the authors refer to relevant work and make an improvement on the previous research to improve the performance on aircraft tracking. They first learn the idea from region-based fully convolutional networks to perform detection on each frame of video. To avoid the target drift due to the failure of object detection on a certain frame, then they employ Kalman filter (KF) and extended KF together to predict the moving trajectory of the target. Beyond that, this method can confine the valid range based on the size of a target object, which increases the speed of detection. This approach can also correct the bounding box on adjacent frames. The steps are not complicated but have an excellent performance. Through the experiment, it is clear that the proposed method is reasonable and more precise.

1 Introduction

Object tracking is a significant field in the computer vision [1]. The general task of tracking is to label the target object with bounding box on per frame during the video. Different from the object detection, there is no need to consider the category to which the object belongs. In recent years, the tracking algorithm has been developed greatly, but there still exist some challenging questions, such as abrupt motion, pose changes, deformation, occlusion, background clutter, and illumination or viewpoint variation.

Over the last few years, many methods based on traditional feature extraction algorithm are proposed, to achieve good results. In [2], Li and Wang, did object tracking by means of eigenbasis space and compressive sampling. Lin *et al.* [3] employed a particle filter to analyse the problem about occlusion during the object tracking. Ahmadi *et al.* presented a strategy which made use of a multi-objective particle swarm optimisation technique [4] and had an excellent result for a small dim object. In [5], kernel-based sparse learning was adopted to design a tracker, which exploited multi-feature via the overall framework. In [6], Gall *et al.* presented a general Hough forest framework which can be applied to object tracking. There are also many classic trackers which are elaborated in [7–11]. However, owing to the features extracted from traditional methods are unable to grasp the essence of images [12], the accuracy of these methods are generally not too high.

To further understand the nature of the image, with the development of deep learning, convolutional neural network (CNN), which can extract useful features automatically, is applied to various visual fields gradually. A few tracking algorithms utilise CNN as a powerful tool to extract the object's features, then devise more efficient trackers. Nam and Han [13] proposed an innovative multi-domain learning framework. This novel network was able to derive domain-independent representations through the pre-training stage and get the domain-specific message by online learning. In [14], Ma *et al.* extracted features from three different depths of the convolution layers by correlation filter, and then applied them to acquire diverse templates, weighted fuse with three confidence maps, finally obtained the target position. In [15], Bertinetto *et al.* embed a fully-convolutional Siamese network into a simple tracking algorithm to enhance the performance and reduce the computation time. In [16], out of consideration for tracking speed, Held *et al.* contrived a tracker that utilises feed-forward neural network without online learning. These relative works can attain more effective features but usually ran for longer.

Meanwhile, tracking can also be divided into multiple subtasks, the most common project is about pedestrian detection and tracking, related researches have been done in this field in recent years include [17–19]. However, in comparison, there are a few approaches that pay attention to tracking military targets like airplanes. Aircraft is an important kind of transportation tool and military objective, therefore the effective tracking of planes has important practical significance. For the specific target object of an airplane, we propose a tracking method based on detection network, region-based fully convolutional networks (R-FCNs) [20], transplant advanced network into the framework and make an improvement to achieve better tracking results. This manner enables to utilise features extracted from CNN with faster speed. The contributions in our paper are listed in the following aspects:

(i) Excellent detection network is used as an observation model in the aircraft tracking systems. To the best of our knowledge, we are the first to employ R-FCN to the tracking framework.
(ii) Adopt a novel strategy which models the state variable of the target separately, linear and non-linear variables are analysed, respectively.
(iii) Take an innovative approach to crop a certain area founded on the scale of the target in each frame based on the previous frame's bounding box.
(iv) Adjust the scale and position of bounding box according to the value of intersection over union (IOU) and the confidence of detection.

The remaining of this paper is organised as follows. Section 2 introduces the related work associated with the proposed metric briefly. Section 3 details the concrete content of the method. In Section 4, the performance of the proposed framework and corresponding experiments are presented. Finally, we make a brief summary of the overall work in Section 5.

Fig. 1 *Different situations in aircraft tracking. From left to right, from top to bottom, corresponding to the scenes: 1 – near to viewpoint (large target); 2 – far from viewpoint (small target); 3 – side of perspective; 4 – blur; 5 – confused by background (overhead view): 6 – confused by background (upward view); 7 – illumination; 8 – occlusion*

2 Related work and motivation

2.1 Object detection network

Object detection has a close connection with object tracking, and our tracking framework is based on the detection network, so it is necessary to demonstrate the related detection algorithms. The landmark work on to detect regions deep learning network for detecting is regions with convolutional neural network (RCNN) features [21], this investigation obtained a certain number of areas where objects may exist via region proposals, input them into CNN, extracted areas' features and determined whether the corresponding region of the feature belongs to a specific class or background through the classifier, then employed features to do regression on bounding box for the sake of revising the predicted position. The algorithm has high precision but there exist too many repeated calculations during the running of the process. So the improved work, fast RCNN (Fast-RCNN) features [22], mapping the proposal of regions into features maps on the last convolution layer of CNN. This method just extracts the features from image only once, so the detection speed is promoted greatly. In view of the excellent performance, after that, Faster-RCNN [23] further breaks through the bottleneck of speed. This algorithm adds region proposal network into Fast-RCNN in order to make a finer classification of region proposals, which is generated by Fast-RCNN and conduct position correction to the bounding boxes. Compared with previous works, Faster-RCNN can achieve better performance. Moreover, R-FCN [20], which is more integrated on convolution computation, realises the fully convolutional operation of whole images. R-FCN utilises the position-sensitive score maps to overcome the dilemma on shift invariance and translational transformation. The precision and speed of this algorithm have reached the advanced level.

In view of the excellent performance on R-FCN, we apply detection network to the overall framework and detect the aircraft target of each frame in the video. Beyond that, we adjust tactics on the original network to make this method more adaptive for tracking task. The specific strategies are elaborated in latersections.

2.2 Applications about Kalman filter (KF)

A typical instance of KF is to predict the coordinates and velocity of an object's position from a set of finite, noisy observational sequences. The main idea is to estimate the state of the object based on the observation model and state model. This algorithm is proposed in [24, 25] and has been employed in many fields in practical engineering. KF is also an important project in control theory and control system, many studies view KF as a powerful tool which is suitable to solve problems. Houtekamer and Mitchell [26] used KF for data assimilation, Obidin and Serebrovski [27] combined the filter with wavelet transform to achieve signal denoising, besides that, Li and Zhao [28] controlled the respective systems and procedure variable with this powerful tool, the authors of [29, 30] designed navigation system based on the KF.

In addition, due to the fact that the KF can predict the future state of the object according to the state model and the observation vector, a number of relevant works have been done to evaluate the target's trajectory, as in [31, 32]. Beyond that, there are many practical problems that are not completely linear, so extended KF (EKF) [33] is presented to fit the non-linear system, EKF carries out first-order Taylor expansion of the non-linear part and employs Jacobian matrix to replace the constant on KF. We mainly draw lessons from the ideas in [34, 35], use the detection network as a model of observation. For the linear–non-linear variables during the task, KF and EKF are utilised to solve the problems, respectively.

2.3 Aircraft tracking

There are several dedicated video databases used for object tracking, which are generally divided into object tracking benchmark (OTB) and visual object tracking (VOT) challenge, specifically include OTB 100 [36], OTB 50 [37], VOT2013 [38], VOT2014 [39], VOT2015 [40] and so on. These datasets involve a variety of scenes and contain multiple categories of objects. It should be noted that there exist some sequences about the airplane, but do not reflect the difficulties encountered during the process of tracking. Owing to the factors about shooting angle, rapid movement or rotation of the plane, the size or the shape of aircraft may be changed very quickly. There are also obstacles such as occluded by something, blocked by the border or confused by background. Besides that, common distortions like blur may also occur. Fig. 1 shows some different situations on aircraft tracking, most of them are hard to find on general datasets.

As Fig. 1 demonstrates, various troubles may appear in the course of aircraft tracking. If only detection network is used, it is difficult to capture the position information of the aircraft in each frame, so the possibility of losing the target will be increased. On the other side, if the KF is used alone, the whole tracking process will fail once the target is lost. Therefore, combining the detection model with effective tracker can enhance performance more efficiently. We will describe the specific approach in the next section and make partial adjustments and improvements.

3 Proposed framework

Our proposed work is based on the theory abovementioned. Fig. 2 shows a framework related to the work. As it is shown in Fig. 2, we can see the brief process of the algorithm. In a simpler term, at first, a sample frame needs to be cropped based on the bounding box of the previous frame, next put it into R-FCN to generate new bounding box, if the difference (use IOU to describe it) between the previous frame and the current frame is larger than the threshold, then directly employ the KF or EKF to process a new bounding box to produce the final box for the current frame; if the difference is smaller than the threshold, correct this box according to the previous box and then use KF to process the revised

bounding box, generating the final bounding box, which is also used for the processing of the next frame. The reason why we adopt this method is that there exist relatively strong correlation between adjacent frames, so these two adjacent frames' bounding box are close to each other, if the distance is relatively far away, meaning that there exists a deviation during detection, therefore, it is necessary to correct bounding box under these circumstances. Within that process, detection network plays a role in preliminary localisation, KF and EKF are used for state estimation by modelling the linear and non-linear parts of the target's motion parameters, respectively. At the same time, tailoring and correction strategies are adopted to improve the precision and speed.

3.1 Observation model R-FCN

In order to establish a proper observation model for the tracking system, we adopt the excellent detection network which is called region-based fully convolutional networks (R-FCNs. R-FCN consists of shared convolution structure, this method incorporates the translation transformation characteristic into FCN and sets up position-sensitive score maps that encode the information about relative spatial location, then attaches region of interest (ROI) pooling layer on the top of FCN to command score maps' message. ROI is divided into $n \times n$ bins, in addition, the pooled response on (i, j) bin in Cth class is formulated as

$$r_c(i, j|\theta) = \sum_{(x, y) \in \text{bin}(i, j)} z_{i, j, c}(x + x_0, y + y_0|\theta) \tag{1}$$

$z_{i, j, c}$ is one of $n^2(C + 1)$ scores maps. These scores maps vote for ROI to generate $(C + 1)$-dimensional vectors $r_c(\theta) = \sum_{i, j} r_c(i, j|\theta)$. The softmax response corresponding to the per class term is calculated as

$$S_c(\theta) = \frac{\mathrm{e}^{r_c(\theta)}}{\sum_{c' = 0}^{C} \mathrm{e}^{r_{c'}(\theta)}} \tag{2}$$

Equation (2) is used to estimate the cross-entropy loss.

The similar method is applied on bounding box regression, but the discrepancy is the additional $4n^2$ convolution layer is added for boundary regression. The position-sensitive ROI pooling is implemented on $4n_2$ maps. Through the average voting and aggregation, the four-dimensional vector can be produced on per ROI. The vector is expressed as $\boldsymbol{v} = (v_x, v_y, v_w, v_h)$, where v_x, v_y mean the X, Y-coordinates of the centre point, v_w, v_h represent the width and the height of the boxes, respectively. Then the vector parameterises bounding boxes for subsequent calculation.

In terms of accuracy and speed, R-FCN achieves the state-of-the-art performance, but it is still inappropriate to apply R-FCN to tracking task directly. Unlike object detection, there is a strong correlation among neighbouring frames in tracking videos. In other words, the object's location is pretty close to each other at adjacent frames. When locating the target on one frame, the object in the next frame will appear at a nearby position. On a side note, the time consume on network detection is positively related to the size of the detection area, so tailoring the frames properly is beneficial for accelerating the speed of detection.

Based on the abovementioned idea, we crop the region in light of the previous frame. Indicating top left, top right, bottom left and bottom right vertex coordinates of the box as B_{tl}, B_{tr}, B_{bl} and B_{bt}, respectively, the relationships between them are listed as follows:

$$\begin{cases} B_{\text{tl}(x)} = v_x - v_w/2, B_{\text{tl}(y)} = v_y + v_h/2 \\ B_{\text{tr}(x)} = v_x + v_w/2, B_{\text{tr}(y)} = v_y + v_h/2 \\ B_{\text{bl}(x)} = v_x - v_w/2, B_{\text{bl}(y)} = v_y - v_h/2 \\ B_{\text{br}(x)} = v_x + v_w/2, B_{\text{br}(y)} = v_y - v_h/2 \end{cases} \tag{3}$$

Suppose the object is detected rightly in a frame, then take the current target for the centre, regard identical central as a basic point, then cut out a certain range, feed the cropped images into the detection network. The width and height about the new detection region are computed as the counterpart of the preceding bounding box, which is enlarged by a factor. Denote this factor as t we draw the relationship the between object's coordinate on consecutive frames. The schematic diagram is displayed in Fig. 3.

It is noticeable that the value of t is a variable, t gradually decreases with the area of the detection box increases. The reason for the practice is that a large object is easy to detect, which does not require much background information; by contrast, the small target needs more false samples to be recognised. More specifically, the form of formula is $t = N/wh$, where N is a constant, w, h represent the width and height of bounding box, respectively. In this way, the practice also prevents tailoring strategies having less effect when aircrafts are larger.

3.2 State estimation model

Supposing that each frame is detected correctly, the goal of tracking the target seems to be achieved, but it is only in the ideal case. Obviously, detection networks are prone to errors because of various obstacles which include blur, occlusion, confusion or rotation. Meanwhile, although the whole trajectory of the airplane is irregular, the motion state can be modelled theoretically among adjoining frames. Inspired by the work in [34], KF is chosen to build dynamic estimation model. Considering that it is dispensable to know the moving speed of an object in the tracking procedure, we take the position of the target's centre point $(\boldsymbol{u}, \boldsymbol{v}, \dot{\boldsymbol{u}}, \dot{\boldsymbol{v}})$ (dot indicates the derivative form), proportion and aspect ratio on a bounding box $(\boldsymbol{s}, \boldsymbol{r}, \dot{\boldsymbol{s}}, \dot{\boldsymbol{r}})$ as a status descriptive variable, construct state vector $[\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{s}, \boldsymbol{r}, \dot{\boldsymbol{u}}, \dot{\boldsymbol{v}}, \dot{\boldsymbol{s}}, \dot{\boldsymbol{r}}]$, which is used for depicting motion trail.

The $\boldsymbol{u}, \boldsymbol{v}, \dot{\boldsymbol{u}}, \dot{\boldsymbol{v}}$ can be approximated by a linear constant velocity model. In a previous work, s and r are deemed as constants, but these are apparently unreasonable. What is quite different from other objects, the airplane's shape is possibly varied greatly during the tracking, which leads to the area and aspect ratio of bounding boxes change as well. Based on the mathematical relationship, s and r are not suitable for a linear model. However, KF is only confined to the linear dynamic system, non-linear variables cannot be properly predicted through filters. To make up for this deficiency, EKF is employed to fit the non-linear part. For the linear part of the state, constructing a sub-vector $[\boldsymbol{u}, \boldsymbol{v}, \dot{\boldsymbol{u}}, \dot{\boldsymbol{v}}]$ which is separated from the state vector above.

On the basis of relevant theory, the following equations can be used to describe the moving target:

$$\begin{cases} x_k = \boldsymbol{A} x_{k-1} + B u_k + w_k \\ z_k = \boldsymbol{H} x_k + v_k \end{cases} \tag{4}$$

where $\boldsymbol{x}$ denotes the state vector of the system, z denotes the measured value. $\boldsymbol{A}$ is known as the state transition matrix. B and u constitute the control parts, but they may be neglected in such a non-controllable system. $\boldsymbol{H}$ represents the transformational matrix which maps variables to measurement value, and w, v signify the noises during the process of status updating and measuring, respectively. The first formula is called the state equation and the second is called the observation equation, subscript k represents the corresponding value at the kth moment. In the corresponding theory [24], assuming the noises abovementioned obey the Gauss distribution. Therefore, the distribution can be formulated as

$$\begin{cases} p(w) \sim N(0, \boldsymbol{Q}) \\ p(v) \sim N(0, \boldsymbol{R}) \end{cases} \tag{5}$$

$\boldsymbol{Q}$ and $\boldsymbol{R}$ indicate the covariance matrices in the multivariate model.

First, calculate the predicted value $\hat{x}_k^-$, error covariance matrix $\boldsymbol{P}_k^-$ between the predictive value and truth value by the following formula:

$$\begin{cases} \hat{x}_k^- = \boldsymbol{A} \hat{x}_{k-1} + B \hat{u}_k \\ \hat{P}_k^- = \boldsymbol{A} P_{k-1} \boldsymbol{A}^{\mathrm{T}} + \boldsymbol{Q} \end{cases} \tag{6}$$

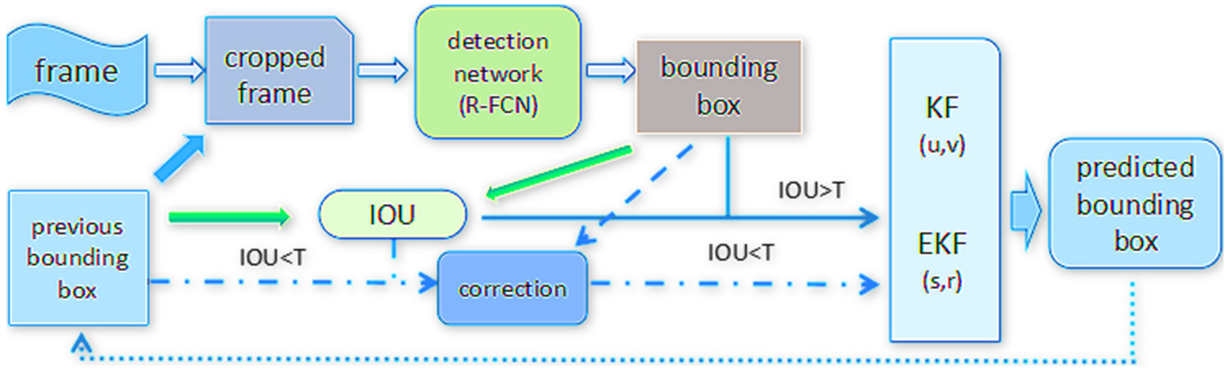

Fig. 2 *Framework of the proposed method. T means threshold, it is a constant set by ourself. The four characters in brackets represent the relevant parameter of a bounding box, where u,v denote the abscissa and ordinate of centre point, respectively, s, r indicate the scale and aspect ratio of the bounding box, respectively*

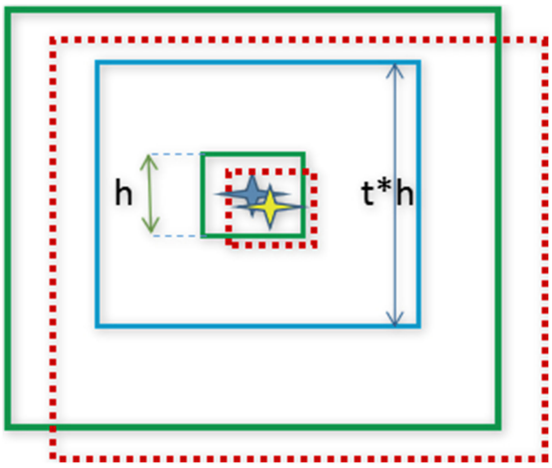

Fig. 3 *Schematic diagram about tailoring strategy. The large and small green solid rectangular boxes represent boundary and bounding boxes on the previous frame, large and small red dashed rectangular boxes indicate the boundary and bounding boxes on the current frame, blur and yellow star denote the target on the previous and the current frames, respectively, blue box signifies the cropped area on the current frame*

Kalman gain K can be computed through the above two equations, the estimation is derived as

$$\begin{cases} K_k = \boldsymbol{P}_k^- \boldsymbol{H}^{\mathrm{T}} \left(\boldsymbol{H} \boldsymbol{P}_K^- \boldsymbol{H}^{\mathrm{T}} + R \right)^{-1} \\ \hat{x}_k = \hat{x}_k^- + K_k (z_k - \boldsymbol{H} \hat{x}_k) \end{cases} \tag{7}$$

Finally, the error covariance matrix is calculated between the estimated value and truth value, preparing for the next recursion on the basis of the formula:

$$\boldsymbol{P}_k = (I - K_k \boldsymbol{H}) \boldsymbol{P}_k^- \tag{8}$$

$\hat{x}_k$ is the kth predicted results, $\boldsymbol{P}_k$ is a covariance matrix of this result. They all participate in the new prediction as an initial value in the next cycle of prior estimation. By the way of iterative algorithm, the correction of the track is gradually completed.

In the analysing of the non-linear problem, the sub-vector $[\boldsymbol{s}, \boldsymbol{r}, \dot{\boldsymbol{s}}, \dot{\boldsymbol{r}}]$ is created in the same way. What is different from before is that state matrix $\boldsymbol{A}$ and mapping matrix $\boldsymbol{H}$ are no longer constant matrices, they are expressed in the following formula:

$$\begin{cases} \boldsymbol{F}_k = \dfrac{\partial f}{\partial x}\Big|_{\hat{x}_{k-1}, u_k} \\ \boldsymbol{H}_k = \dfrac{\partial h}{\partial x}\Big|_{\hat{x}_k} \end{cases} \tag{9}$$

where $\boldsymbol{F}_k$, $\boldsymbol{H}_k$ are the counterparts of $\boldsymbol{A}$, $\boldsymbol{H}$ in EKF, respectively, they are derived from the Jacobian matrix. Adding the non-linear part to a linear system, the state characteristics of the target will be depicted. In contrast to the simple model, this approach is clearly more reasonable.

3.3 Correction of tracking

Several problems are unsolved in the foregoing chapters. On the one hand, target drift may arise in continuous detection, subsequent detection fails when detection error appears in the frame. On the other hand, there also exist diversities between the estimation model and observation model, so it is necessary to combine two schemes effectively. To address these issues, we take IOU as the standard of optimisation.

$\mathrm{IOU_d}$ and $\mathrm{det_a}$, $\mathrm{det_b}$ are introduced to measure the reliability of positioning results. $\mathrm{IOU_d}$ symbolises the overlapping rate between two bounding boxes before and after. If $\mathrm{IOU_d} < T$ (T means threshold), implying that the detection of the frame is out of order, and then $\mathrm{det_a}$, $\mathrm{det_b}$ are designed for weighing the confidence values. These two symbols signify the credibility on detection between fore-and-aft frames, namely, larger value determines higher precision, so the final correction box should get more close to the corresponding detection box. The schematic diagram is illustrated in Fig. 4. Several points need to be explained in Fig. 4. The width of the former box, latter box, corrected box are separately represented as $w_\mathrm{b}, w_\mathrm{a}, w_\mathrm{c}$, respectively, the same mark method is used for height h as well. Similarly, $(x_\mathrm{b}, y_\mathrm{b}), (x_\mathrm{a}, y_\mathrm{a}), (x_\mathrm{c}, y_\mathrm{c})$ corresponding to horizontal and vertical coordinates of the centre point on the respective bounding box. According to the geometric relationship, $(w_\mathrm{c}, h_\mathrm{c}, x_\mathrm{c}, y_\mathrm{c})$ may be derived through the following formulas:

$$\begin{cases} w_\mathrm{c} = w_\mathrm{b} \dfrac{\mathrm{det_b}}{\mathrm{det_b} + \mathrm{det_a}} + w_\mathrm{a} \dfrac{\mathrm{det_a}}{\mathrm{det_b} + \mathrm{det_a}} \\ h_\mathrm{c} = h_\mathrm{b} \dfrac{\mathrm{det_b}}{\mathrm{det_b} + \mathrm{det_a}} + h_\mathrm{a} \dfrac{\mathrm{det_a}}{\mathrm{det_b} + \mathrm{det_a}} \\ \dfrac{|x_\mathrm{c} - x_\mathrm{b}|}{|x_\mathrm{c} - x_\mathrm{a}|} = \dfrac{\mathrm{det_a}}{\mathrm{det_b}} \\ \dfrac{|y_\mathrm{c} - y_\mathrm{b}|}{|y_c - y_d|} = \dfrac{\mathrm{det_a}}{\mathrm{det}_b} \end{cases} \tag{10}$$

In accordance with the above formula, we can obtain the unknown quantities after correction. Eventually, the Hungarian [41] algorithm is applied to get the optimal solution about the observation and estimation model.

Under the situation, where $\mathrm{IOU_d} > T$, the calculation becomes simple. Just employ the Hungarian algorithm as before to obtain the optimal solution. Besides that, in the event that one of IOU_t (The overlap between detection and predicted bounding box is defined IOU_t.) is $< T_{\mathrm{IOU}}$ (can be self-defined). This result means detection error is mainly caused by the model's self-limitation. For this reason, so we abandon this tracker to avoid failed tracking. Once the tracker is terminated, doing the tracking again on the next

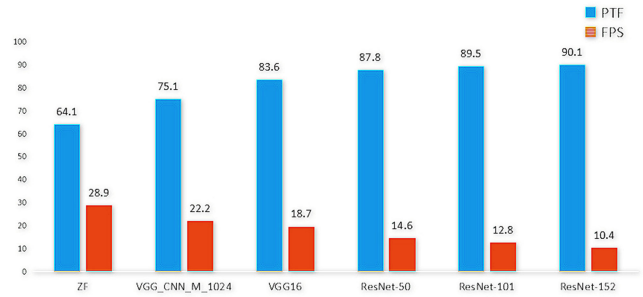

Fig. 4 *Schematic diagram of correction, where blue and green solid rectangular boxes represent the bounding box on before and after frames, respectively, red dashed rectangular box indicates the bounding box after correction, the points with the same colour represent the centre point correspondingly*

Table 1 Performance on different detection networks

Detection network	Recall	Precision	PTF	FPS
RCNN	41.7	60.2	66.7	5.2
FRCN	49.8	68.5	76.8	12.7
FrRCNN	55.4	76.1	87.3	21.4
R-FCN	**69.9**	**81.2**	**95.9**	**29.6**

Bold values indicate the results are better than others.

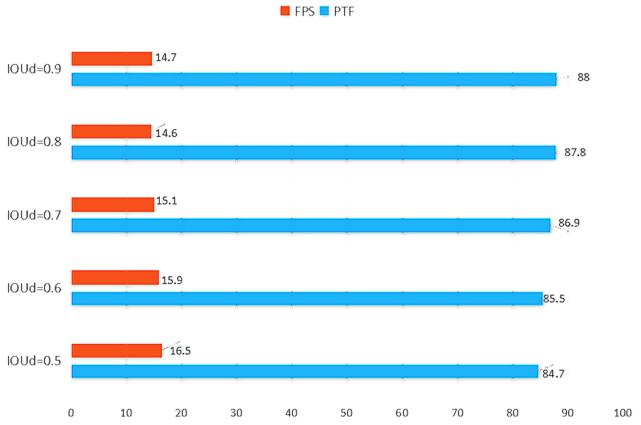

Fig. 5 *Performance under different recognition networks*

frame. When the object is occluded by an obstacle, only the occlusion will be detected in high probability. In such a situation, the detection result should be removed and only rely on the prediction model.

4 Experiments and analysis

4.1 Experimental database

Universal detection and tracking database cannot deal with problems on aircraft tracking well. In order to meet the needs of the task, we collect samples from 30 different videos which contain plane flying in the air. The duration of the videos are not exactly the same. To ensure that the number of samples is large enough, we extract frames from videos at a certain proportion. Moreover, videos should be sampled at regular intervals to avoid getting too similar between samples. First, we choose 2500 frames randomly at a certain proportion to avoid getting too similar between samples. The reason why we select frames as training set is that these data need to be put in a detection network. As for the test set, we choose other 30 videos clips, what needs to be explained is that the object of the test is continuous frames. Only in this way tailoring and correction can be carried out according to the relationship between adjacent frames. So the main difference between training and testing data is not only that they are coming from various time clips, but also the training set consists of discontinuous video frames, while test set is constituted of continuous frames. Besides that, GTX 1080 is used in the following experiments.

4.2 Comparison with different detection network

To verify the effect on the observation model in a tracking task, we compared the accuracy and speed with different detection networks. A series of algorithms are chosen, which include RCNN, Fast-RCNN (abbreviated as FRCN), Faster-RCNN (abbreviated as FrRCNN) and R-FCN. We evaluate the performance of two aspects. On the one hand, conduct object tracking for the whole video, take proportion of true tracked frame (PTF) and frames per second (FPS) as evaluation indexes; on the other hand, divide the entire video into images and carry on the detection for each image, view the recall rate and accuracy as criteria. The strategy can also reflect the role of the estimated model through comparison. In this experiment, Simonyan and Zisserman (VGG16) [42] is selected as a recognition network. The results are tabulated in Table 1.

From Table 1, we can see that the estimated model can improve the detection performance through precision and PTF. In terms of accuracy and speed, R-FCN achieves the best results among congeneric algorithms, which is also proved in other works. Furthermore, by comparing the values in the table, we can draw the conclusion that detection performance has a significant influence on tracking, demonstrating the effectiveness of the observation model.

4.3 Comparison with different recognition network

Due to the great significance of recognition network for accuracy and speed on detection, experiments are executed on R-FCN with various recognition networks such as (Zeiler and Fergus) [43], VGG (Visual Geometry Group) CNN M1024, VGG16, ResNet (Residual Networks)-50, ResNet-101 and ResNet-152 [44]. Results are shown in Fig. 5. In Fig. 5, the histogram distinctly demonstrates the property of networks. The depth of networks is increasing from left to right, and it is clear that the accuracy of tracking is enhanced gradually as the depth increases, while speed varies in the opposite direction. In practical applications, the type of network should be chosen according to the demands; in this paper, for consideration of both speed and accuracy, we selected VGG16 and ResNet-50 in the detection part.

4.4 Displayed tracking results and comparison with other methods

To demonstrate the experiment more visually, we select several illustrations which are used for displaying. Since the difference between adjacent frames is not obvious, the selected frames have a certain time interval. Partial tracking results are exhibited in Fig. 6. The sample graphs are easy to prove that the effect of tracking is excellent. In order to show its performance more precisely, we compare the proposed method with other tracking algorithms about TLD [45], TGPR [46], DLT [47], Struck [48] and CF2 [14], the results are tabulated in Table 2.

In Table 2, TLD shows an advantage in the aspect of speed but its precision is poor among the group. CF2 is competitive in accuracy and proposed work is able to achieve a better result; furthermore, the proposed method outperforms CF2 in tracking speed. Therefore, the performance of the proposed algorithm is better through synthetical consideration.

4.5 Effect on different strategies

In this section, we make a comparison among the diverse schemes. They are separated into several categories: (1) only use detection network; (2) employ the estimation model without cropped strategy; (3) only adopt cropped strategy; (4) adopt cropped and estimated strategy; (5) take cropped and corrected strategy; (6) adopt average cropped strategy; (7) proposed algorithm. To make clear about the differences and similarities between the various strategies, the whole frame is divided into three parts: crop, estimation, and correction (Fig. 7). Make a tick for the involved

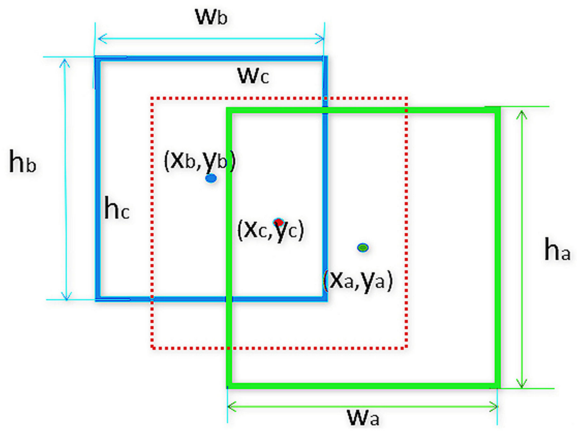

Fig. 6 *Partial displayed tracking results on the proposed algorithm*

Table 2 Performance on different tracking algorithms

Metrics	PTF (IOU>0.8)	PTF (IOU>0.9)	FPS
TLD	62.1	57.0	28.3
TGPR	73.1	69.5	0.79
DLT	55.9	52.9	11.4
struck	67.6	65.4	13.2
CF2	87.2	85.9	14.0
proposed (Res-Net50)	**95.9**	**95.2**	**29.6**

Bold values indicate the results are better than others.

Boldface represents the best indicators in group.

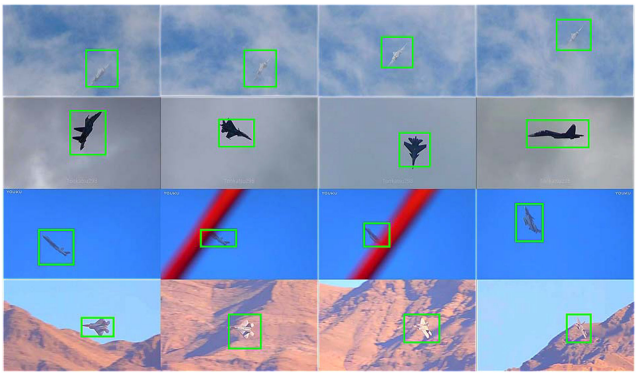

Fig. 7 *Performance under different correction criteria*

Table 3 Comparison of content on diverse schemes

™	Crop	Estimation	Correction
(1) None	×	×	×
(2) estimation	×	✓	×
(3) Crop	×	✓	×
(4) crop + estimation	✓	✓	×
(5) crop + correction	✓	×	✓
(6) average strategy	✓	✓	✓
(7) proposed	✓	✓	✓

'None' represents the scheme without estimation model and cropping strategy.

part and make a cross of contents that are not included. The relational table is shown in Table 3.

In order to make the data more specific, the indicators of each practice are tabulated in Table 4. What should be noted is that recognition network is unified as ResNet-50, and the IOU in the table is distinct from IOU_d, IOU_t described earlier. IOU represents the overlapping ratio between truth value and prediction box. The average strategy in the table represents the approach which takes the average position between the previous and current bounding box.

Table 4 Final performance on diverse schemes

Scheme	PTF (IOU>0.8)	PTF (IOU>0.9)	FPS
(1) none	87.2	85.3	25.8
(2) estimation	88.5	86.1	23.2
(3) crop	87.9	86.1	31.6
(4) crop + estimation	90.5	89.2	28.6
(5) crop + correction	92.3	91.6	30.4
(6) average strategy	94.3	93.0	29.9
(7) proposed	**95.9**	**95.2**	**29.6**

Bold values indicate the results are better than others.

IOU represents the over-lapping ratio between truth value and prediction box

Table 5 Influence of predicted model

Predicted model	PTF (IOU>0.8)	PTF (IOU>0.9)	FPS
(1) KF	94.8	94.0	29.8
(2) EKF	91.3	89.5	28.9
(3) KF + EKF	95.9	95.2	29.6

By comparing between groups, there are many points worth analysing. Compared with (1) and (3), and (2) and (4), we find that cropped strategy is able to speed up effectively, and the accuracy is improved slightly. This is due to the reduction of the tracking range, which helps to decrease the computational time and increase the stability of the location. In contrast with (1) and (2), and (3) and (4), the estimation model is proved to be beneficial for precision. Meantime, this part of the operation also increases the consumption of time. Doing a comparison of (3) and (5), it is clear that the accuracy of the model is promoted after calibration. In addition, the speed is not decreased remarkably during the correction process. By comparing (6) and (7), it is obvious the proposed rectification method achieves better performance, meaning that the veracity is enhanced at almost the same speed.

4.6 Accuracy of tracking model under different correction criteria

In the previous description, IOU_d is used as a standard to determine whether the correction is necessary, so we should pick out a suitable threshold to get better results. On one side, the very small value may tolerate major mistakes, which will cause interruption during the tracking process; on the other side, too large value make the calibration harsh, ignore variability between the inter-frame and introduce the redundant computation. Therefore, the range of values is restricted from 0.5 to 0.9. The results are illustrated in Fig. 6. From Fig. 6, we can find out the general regularity. With the increase of IOU_d, the precision increases gradually while the speed decreases. When the value reaches 0.8, the accuracy is almost not changed, but the cost of time has increased somewhat. The final result is rational because the observation model has a relatively fixed precision. It shows that when IOU_d exceeds a certain threshold, the number of frames, which needs to be corrected does not increase any more. By comprehensive consideration, 0.8 is an appropriate value which is selected in the experiment.

4.7 Influence of predicted model

In the preceding sections, we adapt KF and EKF to in the experiments. Now we only employ one of the filters to explore what changes have been made in the performance. The results are listed in Table 5.

From the table, we can see if employ one of them, the final performance will drop but the operation time is almost the same. In addition, the performance of KF is better than that of EKF. This may indicate that the linear model plays a more important role than non-linear model, and EKF is more complicated than KF.

5 Conclusion

In this paper, we propose a particular tracking algorithm for airplane based on R-FCN and KF. The excellent detection network, R-FCN, is used to provide the location message of a plane for tracking model. To reduce the detection time, we crop a certain region in each frame based on the position of previous frame's bounding box; beyond that, we also change the scale of the detected region according to the target's size. We regard the R-FCN as observation model, combine the model with the estimation model, KF, to modify the prediction of the trace. When the detection on adjacent frames varies remarkably, in order to increase the certainty of detection, the bounding box at next frame is adjusted through the *det* about before and after results. This method is proved effective in the experimental part. There is room for improvement in future work. The proposed algorithm is prone to target transition in the case of multiple aircraft tracking; besides, the detection network has not been improved fundamentally. We will explore this direction on further research.

6 Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (grant no. 61871283), Foundation of Pre-Research on Equipment of China (grant no. 61403120103) and Joint Foundation of Pre-Research on Equipment from Education Department of China (grant no. 6141A02022336). The Key Research Program of Shanxi Province, China (grant no. 2018ZDXM-GY-036).

7 References

[1] Jiang, B., Yang, J., Lv, Z., *et al.*: 'Wearable assistant vision system for visually impaired users based on binocular sensors', *IEEE Internet Things J.*, 2018, **6**, p. 99

[2] Li, J., Wang, J.: 'Adaptive object tracking algorithm based on eigenbasis space and compressive sampling', *IET Image Process.*, 2012, **6**, (8), pp. 1170–1180

[3] Lin, S.D., Lin, J.J., Chuang, C.Y.: 'Particle filter with occlusion handling for visual tracking', *IET Image Process.*, 2015, **9**, (11), pp. 959–968

[4] Ahmadi, K., Salari, E.: 'Small dim object tracking using a multi objective particle', *IET Image Process.*, 2015, **9**, (9), pp. 820–826

[5] Kang, B., Zhu, W.P., Liang, D.: 'Robust multi-feature visual tracking via multi-task kernel-based sparse learning', *IET Image Process.*, 2017, **11**, (12), pp. 1172–1178

[6] Gall, Y., Razavi, G., Lempitsky Gall, J., *et al.*: 'Hough forests for object detection, tracking, and action recognition', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2011, **33**, (11), pp. 2188–2202

[7] Yang, F., Lu, H., Zhang, W., *et al.*: 'Visual tracking via bag of features', *IET Image Process.*, 2012, **6**, (2), pp. 115–128

[8] Xiang, Y., Alahi, A., Savarese, S.: 'Learning to track: online multi-object tracking by decision making'. Proc. of the IEEE Int. Conf. on Computer Vision, Santiago, Chile, 2015, pp. 4705–4713

[9] Choi, W.: 'Near-online multi-target tracking with aggregated local flow descriptor'. Proc. of the IEEE Int. Conf. on Computer Vision, Santiago, Chile, 2015, pp. 3029–3037

[10] Yoon, J.H.., Yang, M.H.., Lim, J., *et al.*: 'Bayesian multi-object tracking using motion context from multiple objects'. 2015 IEEE Winter Conf. on Applications of Computer Vision (WACV), Honolulu, USA, 2015, pp. 33–40

[11] Razavi, S.F., Sajedi, H., Shiri, M.E.: 'Integration of colour and uniform interlaced derivative patterns for object tracking', *IET Image Process.*, 2016, **10**, (5), pp. 381–390

[12] Yang, J., Jiang, B., Li, B., *et al.*: 'A fast image retrieval method designed for network big data', *IEEE Trans. Ind. Inf.*, 2017, **13**, (5), pp. 2350–2359

[13] Nam, H., Han, B.: 'Learning multi-domain convolutional neural networks for visual tracking'. IEEE Conf. Computer Vision and Pattern Recognition, Las Vegas, USA, 2016, pp. 4293–4302

[14] Ma, C., Huang, J.B., Yang, X., *et al.*: 'Hierarchical convolutional features for visual tracking'. IEEE Conf. on Computer Vision, Santiago, Chile, 2015, pp. 3074–3082

[15] Bertinetto, L., Valmadre, J., Henriques, J.F., *et al.*: 'Fully-convolutional Siamese networks for object tracking'. European Conf. on Computer Vision, Amsterdam, Netherlands, 2016, pp. 850–865

[16] Held, D., Thrun, S., Savarese, S.: 'Learning to track at 100 fps with deepregression networks'. European Conf. on Computer Vision, Amsterdam, Netherlands, 2016, pp. 749–765

[17] Li, F., Zhang, R., You, F.: 'Fast pedestrian detection and dynamic tracking for intelligent vehicles within V2V cooperative environment', *IET Image Process.*, 2017, **11**, (10), pp. 833–840

[18] Zeng, X., Ouyang, W., Wang, X.: 'Multi-stage contextual deep learning for pedestrian detection'. IEEE Int. Conf. on Computer Vision, Sydney, Australia, 2013, pp. 121–128

[19] Sun, L., Liu, G., Liu, Y.: 'Multiple pedestrians tracking algorithm by incorporating histogram of oriented gradient detections', *IET Image Process.*, 2013, **7**, (7), pp. 653–659

[20] Dai, J., Li, Y., He, K., *et al.*: 'R-FCN: object detection via region-based fully convolutional networks'. Advances in Neural Information Processing Systems, Barcelona, Spain, 2016, pp. 379–387

[21] Girshick, R., Donahue, J., Darrell, T., *et al.*: 'Rich feature hierarchies for accurate object detection and semantic segmentation'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Columbus, USA, 2014, pp. 580–587

[22] Girshick, R.: 'Fast R-CNN'. Proc. of the IEEE Int. Conf. on Computer Vision, Santiago, Chile, 2015, pp. 1440–1448

[23] Ren, S., He, K., Girshick, R., *et al.*: 'Faster R-CNN: towards real-time object detection with region proposal networks', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017, **39**, (6), pp. 1137–1149

[24] Kalman, R.E.: 'A new approach to linear filtering and prediction problems', *J. Basic Eng.*, 1960, **82**, (1), pp. 35–45

[25] Kalman, R.E., Bucy, R.S.: 'New results in linear filtering and prediction theory', *J. Basic Eng.*, 1961, **83**, (1), pp. 95–108

[26] Houtekamer, P.L., Mitchell, H.L.: 'Data assimilation using an ensemble Kalman filter technique', *Mon. Weather Rev.*, 1998, **126**, (3), pp. 796–811

[27] Obidin, M., Serebrovski, A.: 'Signal denoising with the use of the wavelet transform and the Kalman filter', *J. Commun. Technol. Electron.*, 2014, **59**, (12), p. 1440

[28] Li, Z., Zhao, X.: 'Kalman filter based optimal controllers in free space optics communication', *J. Opt. Soc. Korea.*, 2016, **20**, (3), pp. 368–380

[29] Pashev, G.: 'Kalman filter for measuring the frequency of high-stability oscillators using signals from satellite navigation systems', *Meas. Tech.*, 2014, **56**, (12), pp. 1397–1400

[30] Zhang, Y., Dang, Y., Li, N., *et al.*: 'A integrated navigation algorithm based on distributed Kalman filter'. 2015 IEEE Int. Conf. on Information and Automation, Lijiang, Yunnan, China, 2015, pp. 2132–2135

[31] Gao, S., Liu, Y., Wang, J., *et al.*: 'The joint adaptive Kalman filter (JAKF) for vehicle motion state estimation', *Sensors*, 2016, **16**, (7), p. 1103

[32] Bhattacharya, S., Mukhopadhyay, S.: 'Frequency-weighted prefiltering for Kalman filter based target tracking'. TENCON 2003. Conf. on Convergent Technologies for the Asia-Pacific Region, Bangalore, India, 2003, vol. 2, pp. 836–840

[33] Wiklander, J.: 'Performance comparison of the extended Kalman filter and the recursive prediction error method', Department of Electrical Engineering, 2003

[34] Bewley, A., Ge, Z., Ott, L., *et al.*: 'Simple online and realtime tracking'. 2016 IEEE Int. Conf. on Image Processing (ICIP), Phoenix, USA, 2016, pp. 3464–3468

[35] Azab, M.M., Shedeed, H.A., Hussein, A.S.: 'New technique for online object tracking-by-detection in video', *IET Image Process.*, 2014, **8**, (12), pp. 794–803

[36] Wu, Y., Lim, J., Yang, M.H: 'Object tracking benchmark', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2015, **37**, (9), pp. 1834–1848

[37] Wu, Y., Lim, J., Yang, M.H: 'Online object tracking: a benchmark'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Portland, Oregon, 2013, pp. 2411–2418

[38] Kristan, M., Pflugfelder, R., Matas, J., *et al.*: 'The visual object tracking VOT2013 challenge results'. IEEE Int. Conf. on Computer Vision Workshops, Sydney, Australia, 2014, pp. 98–111

[39] Kristan, M., Pflugfelder, R., Leonardis, A., *et al.*: 'The visual object tracking VOT2014 challenge results', *Lect. Notes Comput. Sci.*, 2016, **8926**, pp. 191–217

[40] Kristan, M., Matas, J., Leonardis, A., *et al.*: 'The visual object tracking VOT2015 challenge results'. Proc. of the IEEE Int. Conf. on Computer Vision Workshops, Santiago, Chile, 2015, pp. 1–23

[41] Mills-Tettey, A., Stent, A., Dias, M.B.: 'The dynamic Hungarian algorithm for the assignment problem with changing costs', Carnegie Mellon University., 2007

[42] Simonyan, K., Zisserman, A.: 'Very deep convolutional networks for large-scale image recognition', arXiv preprint, arXiv:14091556, 2014

[43] Zeiler, M.D., Fergus, R.: 'Visualizing and understanding convolutional networks'. European Conf. on Computer Vision, Zürich, Switzerland, 2014, pp. 818–833

[44] He, K., Zhang, X., Ren, S., *et al.*: 'Deep residual learning for image recognition'. The IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)., Las Vegas, USA, 2016

[45] Kalal, Z., Mikolajczyk, K., Matas, J.: 'Tracking-learning-detection', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2012, **34**, (7), pp. 1409–1422

[46] Gao, J., Ling, H., Hu, W., *et al.*: 'Transfer learning based visual tracking with Gaussian processes regression'. European Conf. on Computer Vision, 2014, pp. 188–203

[47] Wang, N., Yeung, D.Y.: 'Learning a deep compact image representation for visual tracking'. Advances in Neural Information Processing Systems, 2013, pp. 809–817

[48] Hare, S., Golodetz, S., Saffari, A., *et al.*: 'Struck: structured output tracking with kernels', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2016, **38**, (10), pp. 2096–2109