Journal of Computers Vol. 31 No. 3, 2020, pp. 126-141
doi:10.3966/199115992020063103010

An Improved Kernel Correlation Filter Tracker Using Clock Recurrent Neural Network

Gang Wu[1,2*], Chi-she Wang[2], Yong Zhu[1], Shou-bao Su[1]

[1] School of Computer Engineering, JinLing Institute of Technology, Nanjing, 211169, China
{zdhxwg, zhudz, showbo}@jit.edu.cn

[2] Nanjing Innovation Centre of ITS, Nanjing, 211169, China
wangcs@jit.edu.cn

Received 11 November 2018; Revised 25 May 2019; Accepted 31 May 2019

Abstract. The purpose of this study is to solve the bottleneck problem of discriminative object tracker using correlation filter. On the period of learning and updating on correlation filter, errors are likely to be induced into filter, and fatal errors will finally cause tracker inefficiency. Using bidirectional clockwork recurrent neural network to construct confidence map to identify whether the moving target is blocked, a new discriminant tracking algorithm is proposed by integrating clockwork recurrent neural network and kernel correlation filtering. The proposed CKT algorithm uses confidence map to guide the state updating of the clockwork recurrent neural network and optimize the learning process of subsequent kernel related filters. The above measures assist in solving the self-learning problem of the correlation filter in the learning process. Compared with the mainstream object tracking methods on standard testing videos involving VOT2016, the tracking experiments demonstrate that the CKT algorithm respectively lists first, first, fourth and fifth rank on the tracking data A-rank, EAO, R-rank and EFO. On the complicated scenes such as object occluded, object speed drastically changing and light change, etc., the CKT algorithm has better tracking performance than the GGTV2, STAPLEp, CCOT, sKCF and SSKCF algorithms. The proposed CKT algorithm is especially suited to machine learning process where samples are continually acquired and memory storage is limited.

Keywords: clockwork recurrent neural network, confidence map, kernel correlation filtering, light change, object tracking, occlusion

1 Introduction

Using low-cost computer vision method to detect vehicles, pedestrians and specific targets, it has become an important means to construct modern intelligent transportation system in urban traffic scene. Based on computer vision, the detector tracks the moving objects in the traffic scene. Automatic data analysis is carried out on the cluster of computer servers. Thus, the status of traffic data can be generated in time, and effective monitoring information can be provided to the monitors. Object tracking is always a hot topic in the field of computer science and modern intelligent transportation [1]. In the study of intelligent traffic data in urban roads, robust, reliable and real-time tracking of specific targets is conducive to subsequent behavior recognition and analysis. Object tracking is generally divided into generative method and discriminative method. Applying generative model to describe appearance characteristics of objects, generative method minimizes the reconstruction error by searching candidate targets. The representative algorithms are sparse coding, online density estimation and principal component analysis [2] in this field. In the process of target tracking, generative method focuses on the descriptor of the target itself, ignores the background information, and drifts easily when the target is occluded. Compared with the generative tracking method, the discriminant method is more robust in the process of object tracking by the distinctive background and foreground information, and gradually plays a leading role in

* Corresponding Author

this field. The representative methods include support vector machine, correlation filter, multi instance learning, online boosting method and so on [3]. Despite the significant progress has been made in this field in recent decades, object tracking is still a challenging problem, mainly due to large appearance changes caused by occlusion, deformation, abrupt motion, illumination variation, and background clutter [4]. For example, occlusion is a common problem in object tracking. If the tracking algorithm does not have anti-occlusion mechanism, once the target is occluded partially or globally, the tracker will learn a lot of interference information. Unexpected results are likely to appear in the tracking scene, it will eventually lead to tracking drift and failure.

In general, the correlation filtering method belongs to template tracking category, which has relatively poor tracking effect on occlusion, fast deformation and motion. This paper is aiming at the bottleneck problem of discriminative target tracking using correlation filter [5]. On the period of learning and updating on correlation filter, error is likely to be induced into filter, and fatal errors will finally cause tracker inefficiency. Firstly, the proposed method divides the candidate regions of each frame into mesh blocks, and extracts features from each image block. After acquiring the block feature, the spatial association of the blocks is learned from the training process on the bidirectional clockwork recurrent neural network [6] (Abbreviated as CW-RNN). Through traversing frame in two directions, the proposed CKT algorithm calculates the confidence of each image block. The whole confidence map of the candidate region is composed of the predicted values of each image block. Due to the recursive structure of CW-RNN, the output value of each image block is affected by the related blocks, which can avoid adverse effects such as occlusion in a single direction and increase the influence power of trusted region in the overall confidence map. After obtaining the confidence map, the proposed CKT algorithm integrates CW-RNN into the training process of KCF, and weights the filters of different blocks according to the confidence map. The proposed CKT method can suppress the interference of similar objects in the background and enhance the tracking effect on the complicated tracking scene.

1.1 Related Work

In recent years, various machine learning algorithms have been applied to object tracking methods. Be different from the traditional machine learning process where a large number of sample data for off-line training need to be provided, object tracking pays more attention to extracting online training samples from the current data stream in real time. At the same time, object tracking is more focused on tracking the target effectively when the training samples are detected, read and trained for a limited number of times. In VOT2018 challenge, the mainstream trackers were based on various tracking principles: 38 trackers (53%) applied discriminative correlation filters [7], 14 trackers (18%) applied Siamese networks [8], 3 trackers (4%) applied support vector machines [9], 4 trackers (6%) were based on CNN matching [10], one tracker was based on recurrent neural network [6], 6 trackers (8%) applied mean shift [11] and 8 trackers (11%) applied optical flow [12]. Since 2015, there has been a boom in the application of deep learning [13] algorithms on target tracking, as deep learning methods gradually surpass traditional methods in the field of artificial intelligence. At present, most of the object tracking methods based on deep learning also belongs to discriminant tracking framework. The success of deep learning model is mainly due to the effective learning process based on a large number of labeled training data. However, object tracking only provides the first frame as the initial training data. Be different from object detection, deep learning of object tracking lies in the lack of timely training data on the whole tracking process. From the beginning of tracking, it is difficult to train absolutely effective deep model by end-to-end pattern. Given the limited number of training samples in online tracking, it is inferior to directly apply deep learning to tracking since the power of deep learning relies on large-scale training.

It has been found that background information is advantageous for effective tracking, which indicates that discriminative methods are more competing as demonstrated [14-17]. In particular, the correlation filter-based discriminative trackers have made significant achievements recently, and have been paid more attention by researchers. J.Henriques [18] improved the MOSSE filter by introducing kernel methods. By further handling the scale changes, three trackers based on correlation filter (Abbreviated as CFT), namely SAMF [19], DSST [20] and an improved KCF [21], have achieved state-of-art results and have beaten other attended trackers in terms of accuracy in recent competition [22]. With more CFTs are proposed recently, correlation filter-based tracking has proven its great strengths in efficiency and robustness, and has considerably accelerated the development of object tracking. Compared with the deep learning tracking methods, a central strength of the correlation filter tracking methods is that it is

extremely efficient in terms of both memory and computation. There are various aspects that can improve the robustness of CFT method. Improvements have been mainly made on representing features, handling scale variations, and applying part-based strategy over the past years.

The tracking method based on correlation filtering has attracted the attention of many researchers because of its fast speed and outstanding experimental result. The correlation filter trains the filter by regarding the feature regression as the Gauss distribution. In the follow-up tracking, the peak response in the predicted distribution is found to locate the target position. Fast Fourier transform is skillfully used in the operation of correlation filters to improve the processing speed greatly. At present, there are many extension tracking methods based on correlation filter, including KCF, DSST, CCOT, SRDCF, CPT, DPT and so on as demonstrated [14]. According to the analysis of the results of VOT2016-VOT2018 tracking competition [3], the tracking performance of CCOT, SCM, ASLA and other algorithms using correlation filter ranks in the front of all the methods, and these correlation filter methods are much faster than those tracking methods using deep learning algorithms in tracking speed. Using the tracking method such as correlation filter, CSK algorithm first shows the potentiality of correlation filter in object tracking field. In the CSK algorithm, templates cropped from an image can be used to produce peaks for the target. However, their responses to background patterns are also relatively high. To overcome this issue, a variety of correlation filters [7] were trained by suppressing responses to negative training samples while maintaining high response to the target. The main differences among these filters are the methods how they are constructed with the collected training samples. For example, a novel filter termed as Minimum Output Sum of Squared Error (MOSSE) [3] was developed to train correlation filters more efficiently. However, the overall performance may be limited because the MOSSE filter can be viewed as simple linear classifier. An improved KCF based on the CSK algorithm is the research work of J.F. Henriques [18], which affects the approximate dense sampling of ridge regression in many subsequent works. The detailed derivation of the whole kernel correlation filtering algorithm is given in this work. In recent years, the most popular correlation filtering tracking methods such as KCF and Struck have essentially modeled the tracking problem as an optimization problem. One shortcoming of these tracking methods is a bit too complicated on SMO optimization. The performance of Struck will be greatly improved if the appropriate optimization measures and features can be found. Another hard nut to crack is that the above algorithms do not use scale updating mechanism. In the tracking process, the filter will learn a lot of background information if the target shrinks rapidly; On the other hand, the filter will follow the local texture of the target if the target enlarges in successive frames. Both cases are likely to produce unexpected results, which will lead to tracking drift and failure. In the last few years, The tracking approaches based on DCF learns correlation filter to discriminate between the target and background appearance as demonstrated [7]. The training data is composed of observed samples of the target appearance and the surroundings. Despite their success, it is known that standard DCF tracker greatly suffer from the periodic assumption induced by circular correlation. This leads to inaccurate and insufficient training samples as well as a restricted search region. The CCOT algorithm [23] ranks first in VOT 2016, which combines spatial regularization of SRDCF [24] with adaptive sample weights, and extends the depth feature of single-layer convolution in DeepSRDCF [25] to that of multi-layer convolution. Regardless of tracking speed, CCOT can rank first on tracking accuracy and robustness in three consecutive sessions of VOT 2016- 2018. However, the disadvantage of this improved correlation filtering model is that it is becoming more and more complex in the framework, which makes the correlation filtering gradually lose its original speed advantage. The core idea of our work is to improve the tracking accuracy and robustness as much as possible without losing the tracking speed advantage of the correlation filter.

1.2 The Main Contributions of Our Work

In the process of object tracking, the state of the target also has some correlations between the previous frames and the latter frames, so it is very similar to the mechanism of the Recurrent Neural Network (RNN) in dealing with the correlations between the former and the latter frames as demonstrated [26,27]. Unlike feedforward neural networks, RNN introduces directional loops to deal with complex problems in which input variables are correlated. When there exist a certain correlations between the output of a sequence and the output in the past time, RNN will memorize the past information and use it to calculate the current output. The nodes between the RNN hidden layers are connected, and the input of the RNN

hidden layer includes not only the output of the input layer, but also the output of the previous hidden layer. Researchers have proposed various measures to improve the shortcomings of the traditional RNN model in recent years, which include Simple RNN (S-RNN) [28], Bidirectional RNN (B-RNN) [29], Gated Recurrent Unit RNN (GRU-RNN) [30], Long Short-Term Memory (LSTM) [31] and Clockwork RNN (CW-RNN) [32]. However, not all RNN models are suitable for object tracking. For example, LSTM is currently one of the most widely RNN models being used, which can better express long-term and short-term dependencies. It has been successfully applied in word vector expression, sentence validity checking, part-of-speech tagging and so on [33-34]. However, its inherent mechanism is not suitable for object tracking. Overall, the main work of our research is to optimize the traditional KCF tracking algorithm by embedding RNN into kernel correlation filter, and further improve the robustness and anti-occlusion ability of the KCF algorithm. The next section provides an overview of our work. Section 2 describes the advantages of CW-RNN for data prediction. Section 3 describes the architecture of our object tracking model using kernel correlation filtering with CW-RNN in detail. Section 4 discusses the results of experiments in section 3, and the results about mainstream object tracking methods.

2 Advantages of CW-RNN for Data Prediction

CW-RNN is an improved RNN model driven by clock frequency. CW-RNN includes input layer, hidden layer and output layer. CW-RNN divides the hidden layer into several groups, and each group processes the input according to the specified clock frequency. In the traditional RNN model, the relationship among the state variables of input layer, hidden layer and output layer exists as [22]:

$$s_t = f_s(w \cdot s_{t-1} + w_{in}\psi_t). \tag{1}$$

$$o_t = f_o(w_{out}s_t). \tag{2}$$

Where W, W_{in} and W_{out} are the hidden, input and output weight matrices respectively. ψ_t is the input vector at time step t. Vectors s_t and s_{t-1} represent the output of hidden layer at time step t and step t-1, respectively. Vector o_t is the output of step t, f_s is the activation function of the hidden layer, and f_o is the activation function of the output layer. From theoretical view, RNN can handle sequences of arbitrary length. In order to reduce the complexity of the algorithm in practical application, it is usually only set that s_t contains a number of hidden layer states instead of all the hidden layer states. Compared with the traditional RNN model, the advantage of CW-RNN lies in [32]: (1) In order to reduce the algorithm complexity, improve the network performance and accelerate the training process, the number of CW-RNN parameters can be reduced appropriately. (2) To solve the problem of long-term dependence, the hidden layer of CW-RNN works at different clock frequencies, and the hidden layer group of CW-RNN does not work simultaneously at each step, thus speeding up the training process of the network. The neurons in the CW-RNN hidden layer are divided into g groups, each group containing k neurons, each group allocating a clock cycle $T_{\text{i}} \in \{T_1, T_2, ..., T_g\}$. All neurons in the group were all connected. The circular connection between group j and group i needs to meet $T_j > T_i$. The error backward propagation of CW-RNN is similar to the traditional RNN, and the error propagates in the hidden layer group of the execution state.

The choice of the set of periods $\{T_1, T_2, ..., T_g\}$ is arbitrary. W and W_{in} are partitioned into g blocks-rows: the block-upper triangular matrix $w = [w_1, w_2, ..., w_g]^{\text{T}}$ and $w_{in} = [w_{in1}, w_{in2}, ..., w_{ing}]^{\text{T}}$. Fig. 1 shows the schematic diagram of input and output relationships in CW-RNN with 5 hidden layer groups at step t = 6.

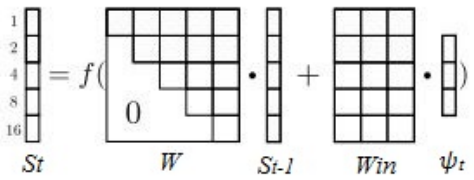


Fig. 1. Relationship between input and output

Fig. 2 shows that three RNN models of RNN, LSTM and CW-RNN are used to predict the effect of simulation data. The blue scatter line is the real data, and the green solid line is the prediction result of the data. The number of nodes in the three models is the same, and there is only one hidden layer. The mean value of the weight is set to 0. The Gauss distribution of standard deviation 0.1 is adopted to initialize. The initial state of the hidden layer is 0. The three models use random gradient descent algorithm to learn and optimize data. The three related RNN models learn the first half of the data and predict the latter part of the data. The RNN model is similar to the calculation of the average value. The prediction accuracy of the LSTM model is not as good as that of the CW-RNN model, therefore, the CKT algorithm uses CW-RNN model to process the data extracted from image features, and construct confidence map to predict whether the target is occluded or not.

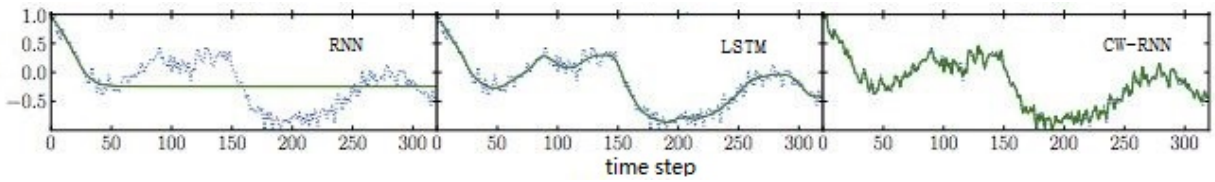


Fig. 2. Prediction effect of three RNN models

3 Object Tracking Using Kernel Correlation Filtering with CW-RNN

With the successful application of RNN in data prediction and machine learning field in recent years, and inspired by the application of correlation filter algorithm in object tracking, a new discriminant target tracking CKT algorithm is proposed in this paper. The algorithm is applied to target tracking by combining clock recurrent neural network with kernel correlation filter. The main work and contributions of this paper are: (1) A new method of constructing confidence map using bidirectional CW-RNN to discriminate the required target from the complex background is proposed. The method can timely detect whether the target is occluded on complex background. (2) An effective closed-form tracking solution is proposed, which combines the advantages of clock recurrent neural network and kernel correlation filter. The kernel correlation filter is updated by combining the new and old correlation filters, and the learning of KCF is optimized by the confidence map generated by CW-RNN.

3.1 Composition and Architecture of CKT Algorithm on Object Tracking

3.1.1 Kernel Correlation Filter Embedded in CKT Algorithm

There have already been some studies [19] to apply kernel methods in correlation filters. J.Henriques [18] proposed that correlation filter can be effectively kernelized with the introduction of ridge regression and circulant matrix [35]. The correlation filter can be treated as an online classifier. The mathematic relation between input x_i and its category attribute y_i is obtained in the training set. Supposing that the relation takes the form $f(x_i) = y_i$, training problem can be viewed as minimizing the objective function [36]:

$$\zeta(z) = \min \sum_i L(f(z, x_i), y_i) + \lambda \|z\|^2 . \tag{3}$$

Where λ is regularization parameter to prevent overfitting, and $L(\cdot)$ is loss function. The parameter $\boldsymbol{z}$ is given by Eq.4:

$$z = (X^{\mathrm{T}} X + \lambda I)^{-1} X^{\mathrm{T}} y . \tag{4}$$

Where X is a matrix whose rows are training samples. y is a vector of corresponding labels, and I is identity matrix. The computation is performed in frequency domain. X^{T} is replaced by the Hermitian transpose of X, which is $X^{\mathrm{H}} = (X^*)^{\mathrm{T}}$. In the kernel correlation filter [3], KCF improves the performance by introducing kernel function, and maps the input data x to the nonlinear feature space $\varphi(x)$. Thus $f(x_i)$ is expressed as follows [37]:

$$f(x_i) = \sum_{j=1}^{n} \alpha_i \cdot k(x_i, x_j) = \sum_{j=1}^{n} \alpha_i \cdot \langle \varphi(x_i), \varphi(x_j) \rangle . \tag{5}$$

After introducing the kernel function, the key to the solution of Eq.3 lies in the calculation by Eq.6 [28]:

$$\alpha = (k + \lambda I)^{-1} y \,. \tag{6}$$

Where $K_{i,j} = k(x_i, x_j)$. To avoid calculating the inverse matrix, the following circulant matrix X as demonstrated [37] is introduced as follows:

$$X = \begin{pmatrix} x_0 & x_1 & \cdots & x_{n-1} \\ x_{n-1} & x_0 & \cdots & x_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ x_1 & x_2 & \cdots & x_0 \end{pmatrix} = F\,diag(\hat{x})F^{\mathrm{H}}. \tag{7}$$

Where F is DFT matrix. The calculation of closed form z in Eq.4 can be transformed into Eq.8:

$$z = F\mathrm{diag}(\frac{\hat{x}}{\hat{x}^{*} \odot \hat{x} + \lambda})F^{\mathrm{H}} y. \tag{8}$$

3.1.2 Components and Framework of CKT Algorithm

Referring to the processing method of the candidate target region as demonstrated [37], the image region within three times of the target in the previous frame is selected as the candidate region. As shown in Fig. 3, candidate region is separated into $m \times n$ spatial grid sub-regions, and the HOG characteristics of d channels are extracted from each candidate target region, then the characteristic set $U \in \mathrm{R}^{h \times w \times d}$ can be obtained, where h and w respectively are the heights and widths of the spatial grids. The sub-region of each space grid is represented by a vertex, thus the candidate target area is represented by graph $\mathrm{G}\{V, \varepsilon\}$, symbol V represents the vertex set of spatial coordinate index, where $V = \{V_{ij}\}\{i = 1,\cdots,h, j = 1,\cdots,w\}$. ε represents the set of edges on adjacent vertices of the space. By traversing graph G, the input state of CW-RNN is set. To alleviate partial occlusion, G is traversed from top to bottom and from bottom to top in two directions. That is CW-RNN is used to traverse the candidate target region from the top and the bottom, and confidence graph is constructed from CW-RNN in two spaces. The confidence graph represents the probability that a subspace of the spatial grid is determined as a background or target. The cross entropy loss function E of the confidence graph is expressed as follows [32]:

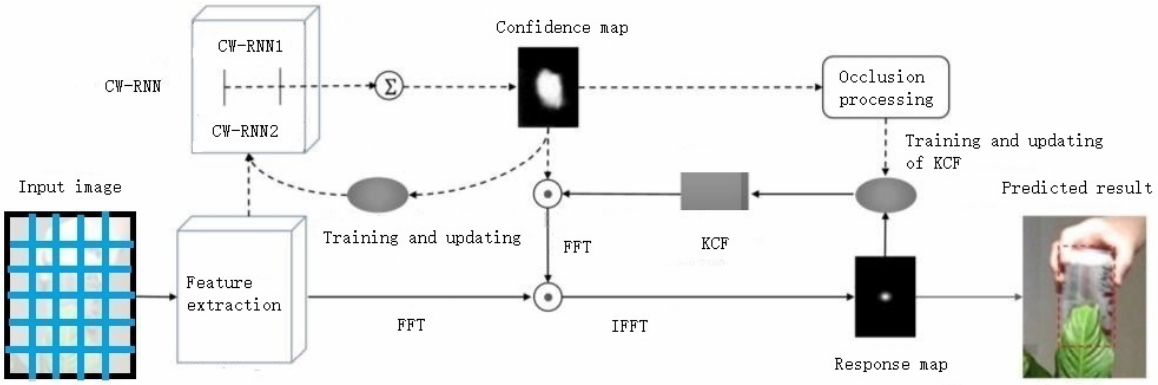


Fig. 3. Components and framework of CKT algorithm

$$E = -\sum_{(i,j)} \sum_{c \in G} y_{i,j}^{c} \ln P_r(c \,|\, v_{ij}) \,. \tag{9}$$

Where y represents the sub-region of the spatial grid as the background or target regions, $y \in C = \{0,1\}$ is the expected binary indicator. $Pr(\cdot)$ is the output probability of the model. Our model uses confidence map to guide the updating of CW-RNN and assist in adjusting the learning process of subsequent KCF.As shown in Fig. 3, the proposed CKT method based on correlation filter is to learn a group of

filters $\{f^k\}, k = 1,...,d$, each filter for one HOG feature channel in $U = \{u_1, u_2, ..., u_d\}$. The learning process of the weighted correlation filter [19] is expressed as minimizing the loss function $\zeta(f)$:

$$\zeta(f) = \left\| \sum_{k=1}^{d} u^k * f^k - y \right\|^2 + \sum_{k=1}^{d} \left\| \eta \odot f^k \right\|^2 . \quad \textbf{(10)}$$

Where symbol * is space convolution, symbol $\odot$ is product of pixel direction, f^k is convoluted with the characteristics of k channels. Based on the confidence map of CW-RNN, our method adjusts the training and updating process of KCF model by the weight η . The overall algorithm complexity of CKT algorithm is O(($h \times w$)^3+d×(h××w)^2), which integrates feature extraction, occlusion processing, CW-RNN [32] training and updating, KCF training and updating.

3.2 Execution Details and Parameter Settings of CKT Algorithm

Feature extraction. The CKT algorithm uses HOG features to extract features for tracking task. The HOG features are collected in the candidate target region and a series of 4×4 pixels of the spatial grid are extracted for quantization processing.

Occlusion processing. The confidence rate is defined as the accumulation of probability values in the target region. If the confidence rate of the current frame is lower than the average value on the previous frames, it is considered that the target in the current frame has been occluded. The threshold of confidence rate is set to 0.8 on experience values in the object tracking period. A confidence map from CW-RNN [32] is used to predict whether occlusion exists. When the target is predicted to be a high probability occlusion, the KCF model will not be updated for the time being.

Training and updating of CW-RNN. Due to the training samples are insufficient, learning rate of 0.02 and the initial 5 frames are used to train CW-RNN on the initial stage of object tracking. The CW-RNN is updated with a fixed interval of five frames in subsequent tracking period. In order to avoid over-fitting of CW-RNN, a small learning rate of 0.001 is used to fine-tune the CW-RNN after the initial five frames.

Training and updating of KCF. KCF is initialized in the first frame. As indicated by Eq.11, integrating the new and old filter methods, the state of KCF is updated in the subsequent frame, and the KCF learning factor θ is set to the empirical value of 0.03.

$$\overline{f} = \theta f_{new} + (1-\theta)\overline{f} . \quad \textbf{(11)}$$

4 Experiments and Analysis

Our experimental hardware platform is composed of CPU Intel Xeon E5V4-3.5GHZ, 32GB RDIMM storage, and Nvidia GPU K80 graphics card. Based on the VOT2016 standard dataset [38], the proposed CKT algorithm and other mainstream tracking algorithms are used to test on the object tracking experiment. The main tracking challenges lie in the rapid movement of targets, camera motion, illumination intensity change, occlusion, et al.

As shown in Fig. 4, tracking results of the nineteen algorithms on the VOT2016 standard sequences are visible. The color box respectively represents the position predicted by corresponding algorithm. In order to further compare the proposed method with current mainstream tracking methods, we do further test to acquire quantitative tracking results on different scenes about VOT2016 dataset. These complex scenes mainly involve the situations of camera motion, illumination intensity change, motion direction change, occlusion and scale change. The tracking experiments are divided into two groups: baseline experiment and unsupervised experiment. In terms of object tracking evaluation indicators, the center location errors of the tracked target are defined as the distance between the central locations and the manually labeled ground truth. The location error with respect to object center is applied for quantitative evaluations. Given the tracked bounding box ROI_T and the ground truth bounding box ROI_G, the tracking score [39-42] is defined by:

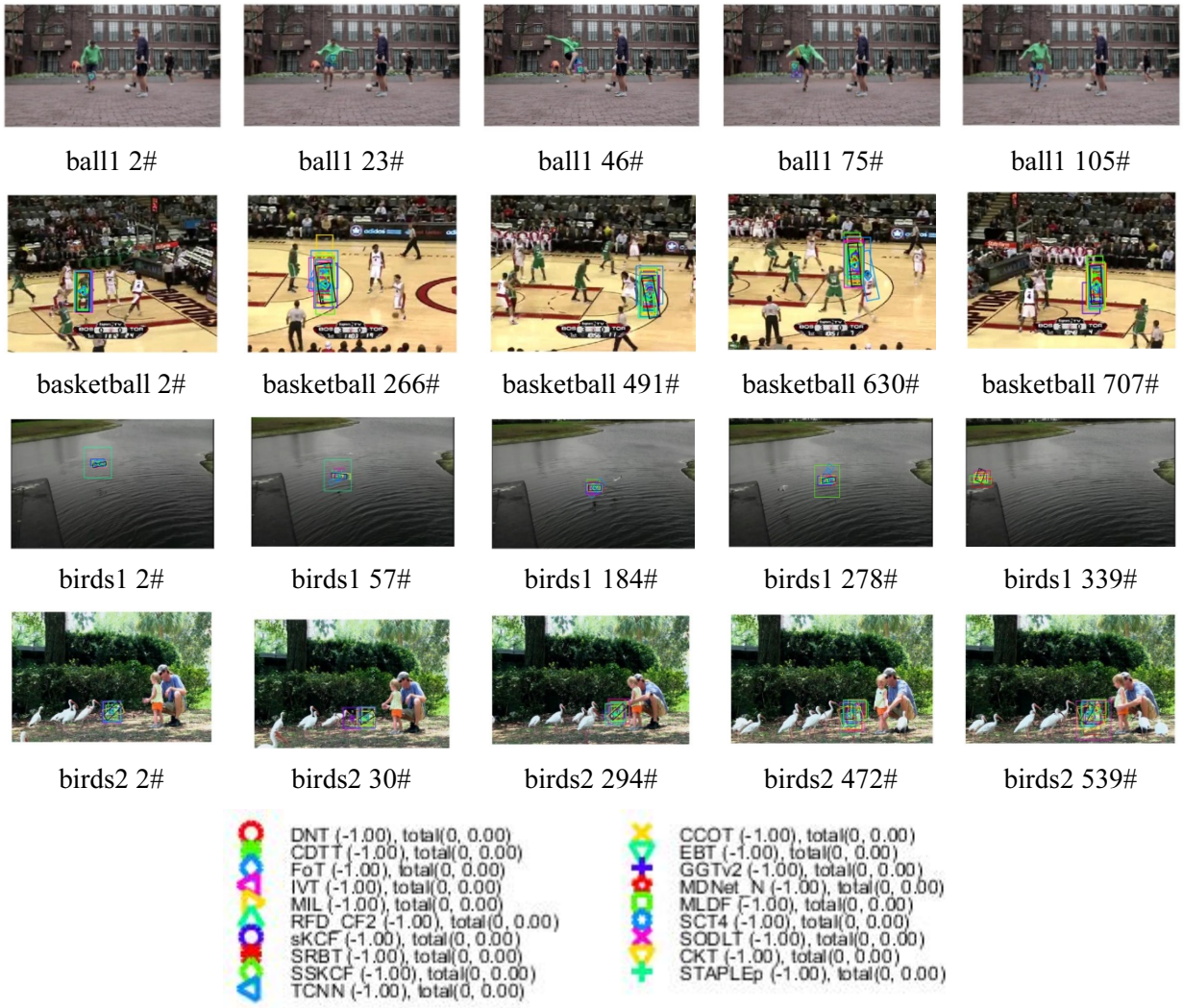


Fig. 4. Tracking results of nineteen tracking algorithms

$$score = \frac{area(ROI_T \cap ROI_G)}{area(ROI_T \cup ROI_G)}. \tag{12}$$

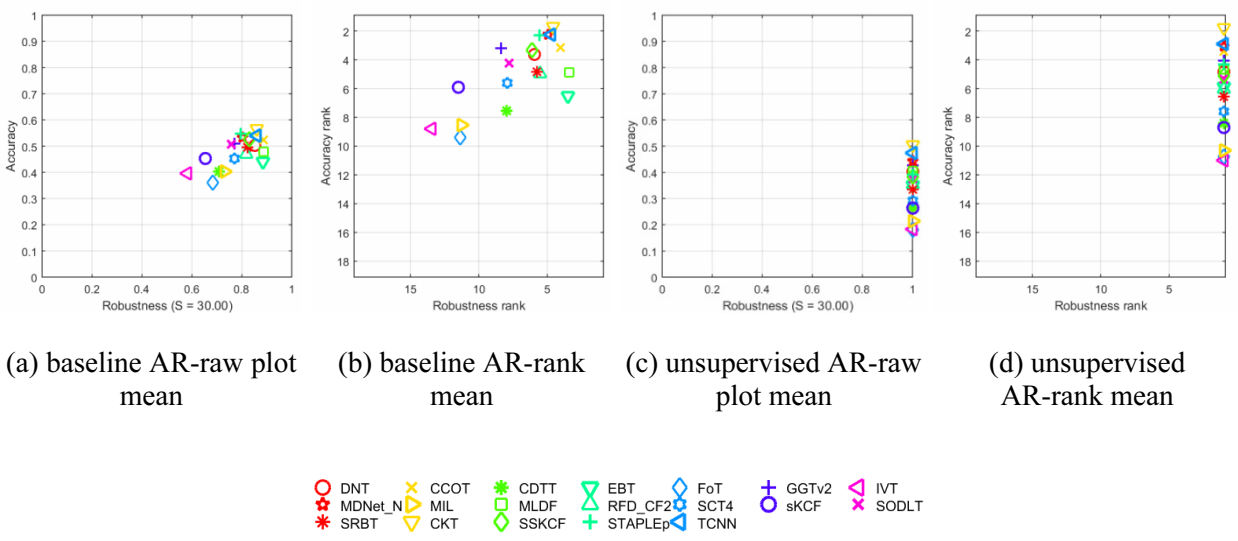
When the tracking score is above 0.5, the tracking result is considered as success at the frame. When a failure is detected, the tracker is reinitialized after five frames. In the evaluation criteria of image tracking, accuracy and robustness are the two most uncorrelated evaluating indicators. The accuracy is the average overlap (AO) between the predicted and ground truth bounding boxes during successful tracking periods. Average overlap can be computed directly without intermediate success plots, giving the measure a clear interpretation. Due to variable lengths on practical image sequences, the matter of increased bias and variance of AO measure is objective existence. Compared with the no-reset AO such as the OTB, the VOT reset-based AO drastically reduces the bias and the corresponding variance. Robustness measures the times the tracker loses target during the whole tracking periods. Robustness is calculated according to the following formula.

$$R_s = e^{-SM}. \tag{13}$$

Where M is the average number of tracking failures, and $M = F_0 / N$. Parameter N is the length of a specific sequence, and F_0 is the total number of tracking failures. The parameter S is a manually selected parameter. The meaning is the number of frames that are expected to be tracked continuously. The coefficient S is set to 30 in our experiment. Accuracy-robustness ranking (AR-rank) plots [32] were proposed to visualize the tracking results. A high average rank means that a tracker was well-performing

in accuracy as well as robustness. In VOT2016 evaluation criterion, the AR raw plots [32] were constructed to show the absolute average performance. Performance is evaluated in all of these nineteen approaches by overlaps between the predicted bounding boxes with the ground truth bounding boxes. A-rank and R-rank are ultimately computed with all the image sequences.

Fig. 5 shows the AR-raw and AR-rank plots generated by testing sequences on the nineteen tracking method. The nineteen tracking algorithms include the proposed CKT, the current mainstream TCNN [32], DNT [32] and MDNET_N [43] algorithms, as well as the traditional IVT [32] and MIL [32] algorithms as a reference. The nineteen trackers originate from various classes. The MLDF [44], TCNN, and DNT are derived from CNNs [45-46]. The sKCF [47], GGTV2 [48], SSKCF [49] and CKT are variations of kernel correlation filters. CCOT [50] and STAPLEp [51] belong to correlation filtering trackers using different features. The EBT [52] is structured SVM tracker, Fot [53] is block tracker, while IVT, MIL, SRBT [54] and CDTT [32] belong to the appearance model tracker. As shown in Fig. 5(a) and Fig. 5(c), the tracking results are summarized in AR-raw plots on the baseline and unsupervised experiment. The baseline and unsupervised AR-raw plots are constructed by concatenating the tracking results from all 60 testing sequences. The normalized AR-rank plot is obtained by averaging the rank lists. The ranking converts the accuracy and robustness to the same scale. In AR-rank plots shown in Fig. 5(b) and Fig. 5(d), the tracking method with smaller accuracy and robustness has relatively better tracking performance. The baseline experiment is illustrated by Fig. 5(b) As a reference, the tracking accuracy and robustness of IVT and MIL algorithms are ranked at the lower left side of Fig. 5(b) The tracking accuracy and robustness of the proposed CKT algorithm respectively rank first, fourth in all nineteen methods. The unsupervised experiment is illustrated by Fig. 5(d) Fig. 5(d) shows that the tracking accuracy of the CKT algorithm ranks first in the nineteen methods on the equivalent condition of robustness. The AR-rank data of nineteen tracking algorithms are collected in Table 1, and the red, blue and green data respectively represent the first, second and third places in the list. From the tracking data in Table 1, it can be seen that the proposed CKT algorithm ranks the first of nineteen tracking methods and the comprehensive evaluation of A-rank is 1.73. The comprehensive evaluation of R-rank for CKT algorithm is 2.80, which ranks fourth in all nineteen tracking methods.



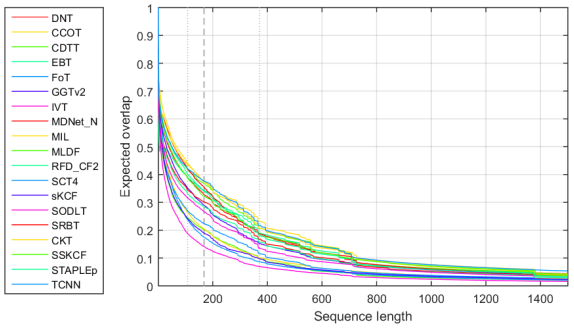
(a) baseline AR-raw plot mean (b) baseline AR-rank mean (c) unsupervised AR-raw plot mean (d) unsupervised AR-rank mean

Fig. 5. The AR-raw and AR-rank plots generated by all sequences

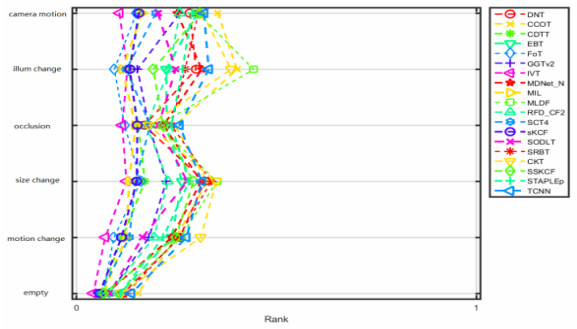
Fig. 6 shows expected overlap (EO) curves [32] on the nineteen tracking algorithms. Fig. 7 shows the baseline experimental expected overlap plots on the different scenes such as camera motion, illumination change, motion change, occlusion, et al. As shown in Fig. 7, the right-most tracker is the top performing according to expected average overlap values. Due to the data of the AR plots and the AR-rank plots cannot directly reflect the advantages of these tracking algorithms, expected average overlap (EAO) [32] is introduced to combine the raw values of tracking failures and accuracies in a principled pattern.

Table 1. The AR-rank data of nineteen tracking algorithms

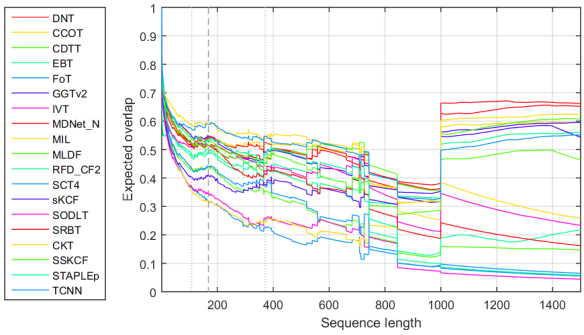
Tracking algorithms	baseline experiment		unsupervised experiment		Comprehensive evaluation	
	A-rank	R-rank	A-rank	R-rank	A-rank	R-rank
DNT	3.65	5.92	4.83	**1.00**	4.24	3.46
CCOT	3.17	4.02	3.42	**1.00**	3.29	2.51
CDTT	7.55	7.95	8.42	**1.00**	7.98	4.47
EBT	6.50	**3.48**	5.87	**1.00**	6.18	**2.24**
FoT	9.40	11.35	10.73	**1.00**	10.07	6.17
GGTv2	3.22	8.37	4.08	**1.00**	3.65	4.68
IVT	8.82	13.43	10.98	**1.00**	9.90	7.22
MDNet_N	**2.25**	4.95	3.07	**1.00**	2.66	2.98
MIL	8.53	11.28	10.32	**1.00**	9.43	6.14
MLDF	4.87	**3.42**	4.68	**1.00**	4.78	**2.21**
RFD_CF2	4.95	5.52	6.03	**1.00**	5.49	3.26
SCT4	5.62	7.90	7.58	**1.00**	6.60	4.45
sKCF	5.92	11.48	8.72	**1.00**	7.32	6.24
SODLT	4.22	7.78	5.35	**1.00**	4.78	4.39
SRBT	4.85	5.73	6.57	**1.00**	5.71	3.37
CKT	**1.68**	4.60	**1.78**	**1.00**	**1.73**	2.80
SSKCF	3.32	6.10	5.08	**1.00**	4.20	3.55
STAPLEp	2.32	5.58	4.33	**1.00**	3.33	3.29
TCNN	**2.25**	4.70	**2.88**	**1.00**	**2.57**	2.85



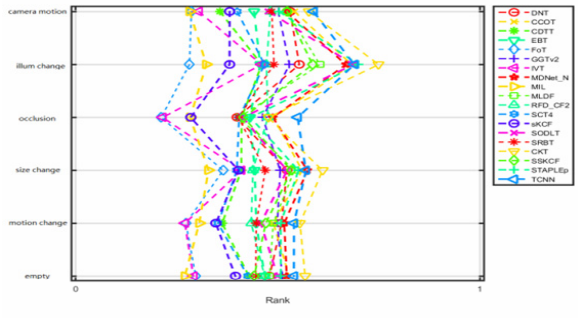
EO curves on baseline experiments



EO ordering of baseline experiments



EO curves on unsupervised experiments



EO ordering of unsupervised experiments

Fig. 6. Expected overlap curves on the nineteen tracking algorithms

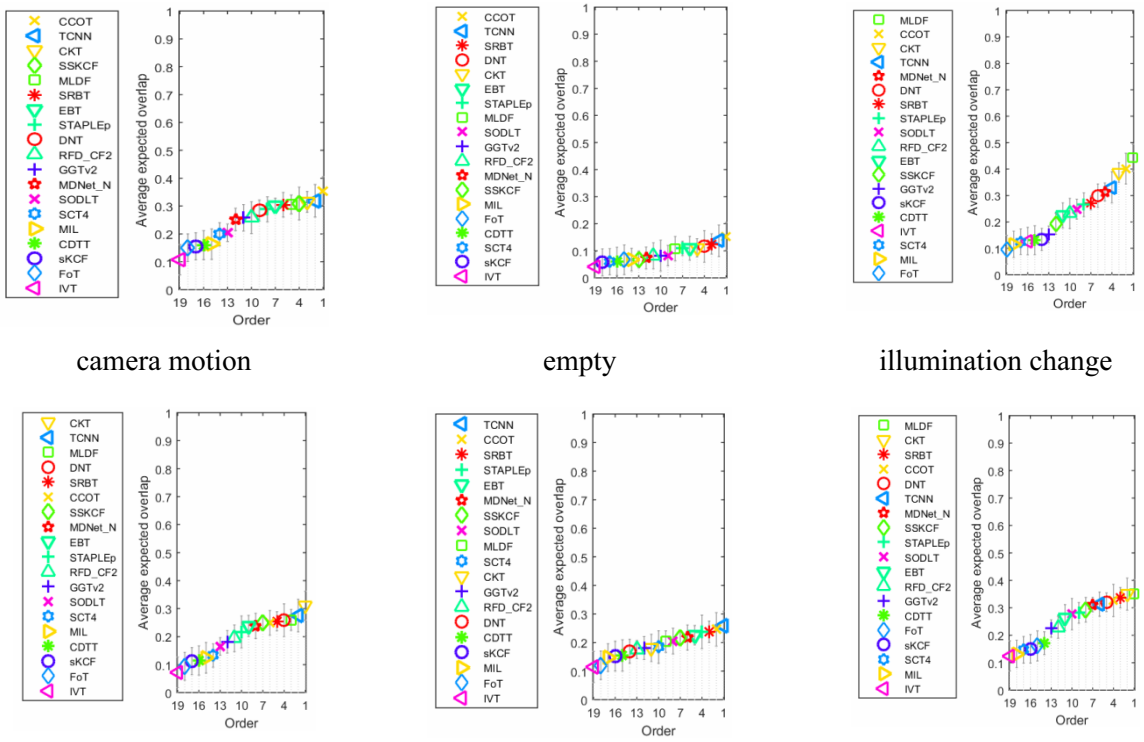
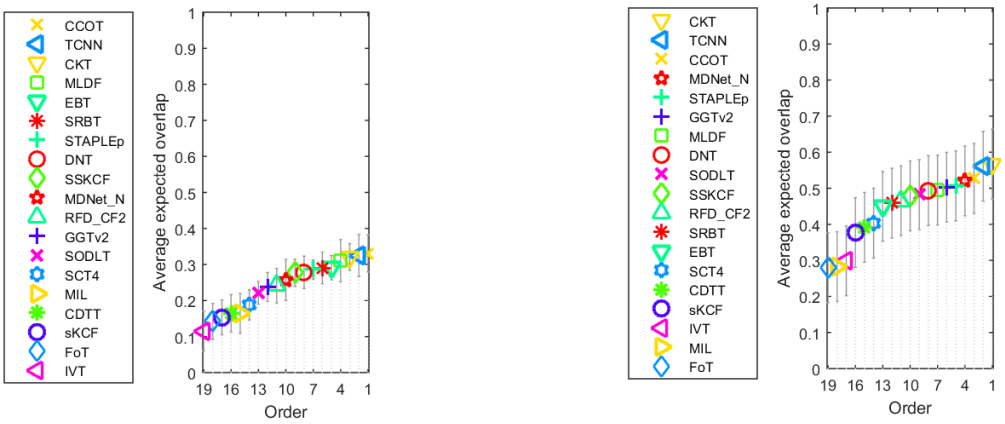


Fig. 7. Baseline experimental expected overlap plots on different scenes

Compared with the original EO, the EAO measures the expected no-reset overlap of a tracker running on a short-term sequence. Fig. 8 is the expected average overlap curves of the nineteen tracking algorithms. Table 2 shows the raw values for the EAO scores.



(a) EAO of baseline experiments (b) EAO of unsupervised experiments

Fig. 8. EAO data for the nineteen tracking algorithms

Table 2. The EAO and EFO data of nineteen tracking algorithms

Tracking algorithms	baseline experiment	unsupervised experiment	Comprehensive evaluation of EAO and EFO	
	EAO	EAO	EAO	EFO
DNT	0.2783	0.4936	0.3859	1.127
CCOT	**0.3310**	0.5276	0.4293	0.507
CDTT	0.1644	0.3916	0.2780	13.398
EBT	0.2913	0.4499	0.3706	3.011
FoT	0.1420	0.2793	0.2106	**105.714**
GGTv2	0.2377	0.5028	0.3703	0.357
IVT	0.1147	0.2991	0.2069	14.880
MDNet_N	0.2572	0.5204	0.3888	0.534
MIL	0.1645	0.2825	0.2235	7.678
MLDF	0.3106	0.4950	0.4028	1.483
RFD_CF2	0.2415	0.4654	0.3535	0.896
SCT4	0.1879	0.4036	0.2957	11.131
sKCF	0.1533	0.3771	0.2652	**91.061**
SODLT	0.2213	0.4826	0.3520	0.576
SRBT	0.2904	0.4587	0.3745	3.688
CKT	0.3207	**0.5670**	**0.4439**	28.472
SSKCF	0.2771	0.4791	0.3781	29.153
STAPLEp	0.2862	0.5066	0.3964	44.765
TCNN	**0.3249**	**0.5610**	**0.4429**	0.507

An important measure called the equivalent filter operations (EFO) [32] is introduced to partially accounts for the speed of computer used for tracker analysis. To put EFO units into perspective, a C++ implementation of a NCC tracker provided in the VOT toolkit runs with average 140 frames per second, the execution speed of NCC tracker equals to 200 EFO units. Table 2 describes the EFO scores for the all nineteen trackers in this paper. The 30 pixel×× 30 pixel window filtering is performed on 600 pixel×× 600 pixel image, and then each frame is processed using a tracking algorithm. Dividing the evaluation time of the image into the filtering operation time, we calculate the normalized performance parameter EFO of the tracking algorithm. The EAO and EFO data of nineteen tracking algorithms are collected in Table 2. The red, blue and green data respectively represent the first, second and third places in the list. From the data in Table 2, it can be seen that the proposed CKT algorithm ranks the first of nineteen tracking methods and the comprehensive evaluation of EAO is 0.4439. According to the EAO measure, the top performing tracker was CKT, followed by the TCNN. In contrast to EAO data, the CKT method is significantly superior to the similar correlation filtering method such as GGTv2 (0.3703), sKCF (0.2652), CCOT (0.4293), STAPLEp (0.3964) and SSKCF (0.3781). The EFO of CKT algorithm is 28.472, which ranks fifth in nineteen tracking methods.

5 Conclusions

Object tracking plays an important role in constructing modern intelligent transportation system. Aiming at the unsolved difficult issues existing in the process of object tracking using correlation filter, a new discriminant object tracking CKT algorithm is proposed by introducing a clock loop neural network into kernel correlation filtering in this paper. The proposed method uses bi-directional CW-RNN to mine reliable parts of target in the object tracking process. By modeling CW-RNN on two-dimensional plane, the tracking drift problem caused by accumulation of prediction errors is preferably solved. It is also an improvement of discriminant object tracking method using KCF. The object tracking experiments are based on VOT evaluation criteria and data sets. The object tracking evaluation contains sixty image sequences in which targets are denoted by rotated bounding boxes. These image sequences involve complex tracking situations including camera motion, illumination intensity change, motion direction change, occlusion and scale change. 19 trackers including our method have been evaluated on these image sequences. A large percentage of trackers have been published at mainstream computer vision conferences and top journals, including CVPR, ICCV, TPAMI and TIP. The proposed CKT tracker performs very well in accuracy as well as robustness, which are reflected in the A-rank, R-rank and EAO

data. The proposed CKT tracker uses the bi-directional CW-RNN to construct confidence map to identify whether the target is occluded, so as to effectively suppress the negative impact of complicated background information on the tracking period. Our method uses confidence map to guide the state updating of clock loop neural network and optimize the subsequent learning process of kernel correlation filter. The tracking experiments on sixty standard VOT2016 test sequences show that A-rank, R-rank, EAO and EFO of the proposed CKT algorithm are 1.73, 2.80, 0.4439 and 28.472, respectively, and the tracking data of the proposed CKT algorithm respectively list first, fourth, first and fifth of the 19 tracking methods. As a correlation filtering method, the proposed CKT algorithm is significantly superior to GGTV2 [48], sKCF [47], CCOT [50], STAPLEp [51] and SSKCF [49] in tracking performance. In image tracking process, the proposed CKT algorithm improves the performances of traditional kernel correlation filter. The CKT algorithm adapts to the scale change of the target in the real-time tracking process and effectively reduces the error accumulation of traditional KCF tracking algorithm. A confidence map is constructed by using the bi-directional CW-RNN to guide the state updating of the clock-loop neural network. The proposed algorithm can significantly improve the tracking performance of the kernel correlation filtering. At the same time, it shows that CW-RNN is effective in mining associated relations on adjacent mesh blocks and restricting kernel correlation filters. In the further research of object tracking, we will try to build more effective background model for the image region to further improve the on-line tracking performance of the algorithm.

Acknowledgements

The research work of this paper originates from related projects of Nanjing Innovation Centre of ITS. This research was funded by Nanjing Innovation Centre of ITS. The projects involve the following items such as The National Natural Science Foundation of China (No. 61375121 and No. 61305011), Nanjing economic and Information Committee project (Transport big data public service platform), Nanjing science and Technology Committee project (No. 201704002). On behalf of the research group, I would like to thank Professor Wang, Professor Zhu and Professor Su for their strong support for this study. At the same time, I would like to thank all my colleagues in the research group.

References

[1] H. Li, Y. Li, F. Porikli, Deeptrack: Learning discriminative feature representations online for robust visual tracking, IEEE Transactions on Image Processing 25(4)(2015) 1834-1848.

[2] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[3] M. George, B.R. Jose, J. Mathew, Performance evaluation of KCF based trackers using VOT dataset, Procedia Computer Science 125(2018) 560-567.

[4] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, P.H. Torr, Staple: Complementary learners for real-time tracking, in: Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[5] H. Li, H. Wu, S.-J. Lin, X.-N. Luo, Coupling deep correlation filter and online discriminative learning for visual object tracking, Journal of Computational and Applied Mathematics 329(2018) 191-201.

[6] E. Gundogdu, A.A. Alatan, Good features to correlate for visual tracking, IEEE Transactions on Image Processing 27(5)(2018) 2526-2540.

[7] A. Lukezic, T. Vojir, L.C. Zajc, J. Matas, M. Kristan, Discriminative correlation filter with channel and spatial reliability, in: Proc. 2017 IEEE Conf. on Computer Vision and Pattern Recognition, 2017.

[8] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, W. Hu, Distractor-aware siamese networks for visual object tracking, in: Proc. 2018 European Conference on Computer Vision, 2018.

[9] X. Wang, Y.-L. Xing, An online support vector machine for the open-ended environment, Expert Systems with Applications 120(2019) 72-86.

[10] Y. Song, Q.V. Hu, L. He, P-CNN: Enhancing text matching with positional convolutional neural network, Knowledge-Based Systems 169(2019) 67-79.

[11] O. Sliti, H. Hamam, H. Amiri, CLBP for scale and orientation adaptive mean shift tracking, Journal of King Saud University - Computer and Information Sciences 30(3)(2018) 416-429.

[12] Z.-G. Tu, W. Xie, D.-J. Zhang, R. Poppe, R. C. Veltkamp, B.-X. Li, J.S. Yuan, A survey of variational and CNN-based optical flow techniques, Signal Processing: Image Communication 72(2019) 9-24.

[13] M. Yan, M.-L. Li, H.-W. He, J.-k. Peng, Deep learning for vehicle speed prediction, Energy Procedia 152(2018) 618-623.

[14] Z. Chen, Z. Hong, D. Tao, An experimental survey on correlation filter-based tracking, Computer Science 53(6025)(2015) 68-83.

[15] Y. Yin, D. Xu, X. Wang, M. Bai, Online state-based structured SVM combined with incremental PCA for robust visual tracking, IEEE Transactions on Cybernetics 45(9)(2015) 1988-2000.

[16] B. Babenko, M.-H. Yang, S. Belongie, Robust object tracking with online multiple instance learning, IEEE Transactions on Pattern Analysis and Machine Intelligence 33(8)(2011) 1619-1632.

[17] L. Fiaschi, F. Diego, K. Gregor, M. Schiegg, U. Koethe, M. Zlatic, F.A. Hamprecht, Tracking indistinguishable translucent objects over time using weakly super-vised structured learning, in: Proc. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[18] J. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE Transactions on Pattern Analysis and Machine Intelligence 37(3)(2015) 583-596.

[19] Y. Li , J. Zhu, A scale adaptive kernel correlation filter tracker with feature integration, in: Proc. 2014 Computer Vision-ECCV 2014 Workshops, 2014.

[20] M. Danelljan, G. Häger, F.S. Khan, M. Felsberg, Accurate scale estimation for robust visual tracking, in: Proc. 2014 British Machine Vision Conference, 2014.

[21] G.S. Walia, R. Kapoor, Recent advances on multicue object tracking: a survey, Artificial Intelligence Review 46(1)(2016) 1-39.

[22] B. Shuai, Z. Zuo, G. Wang, Quaddirectional 2d-recurrent neural networks for image labeling, IEEE Signal Processing Letters 22(11)(2015) 1990-1994.

[23] M. Danelljan, A. Robinson, F.S. Khan, M. Felsberg, Beyond correlation filters: Learning continuous convolution operators for visual tracking, in: Proc. 2016 European Conference on Computer Vision, 2016.

[24] M. Danelljan, G. Hager, F. Shahbaz Khan, M. Felsberg, Learning spatially regularized correlation filters for visual tracking, in: Proc. 2015 IEEE International Conference on Computer Vision, 2015.

[25] M. Danelljan, G. Hager, F.S. Khan, M. Felsberg, Convolutional features for correlation filter based visual tracking, in: Proc. 2015 IEEE International Conference on Computer Vision Workshops, 2015.

[26] Q. Wang, F. Chen, W. Xu, M.-H. Yang, Object tracking with joint optimization of representation and classification, IEEE Transactions on Circuits and Systems for Video Technology 25(4)(2015) 638-650.

[27] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, Exploiting the circulant structure of tracking-by-detection with kernels, in: Proc. 2012 European Conference on Computer Vision, 2012.

[28] S. Li, W.-C. Zhang, A.B. Chan, Maximum-margin structured learning with deep networks for 3D human pose estimation, International Journal of Computer Vision 122(1)(2017) 149-168.

[29] B. Fan, L. Xie, S. Yang, A deep bidirectional LSTM approach for video-realistic talking head, Multimedia Tools and Applications 75(9)(2016) 5287-5309.

[30] Y.-Z. Zhang, R. Yamaguchi, S. Imoto, Sequence-specific bias correction for RNA-seq data using recurrent neural networks, BMC Genomics 18(1)(2017) 1-10.

[31] L. Chen, Y.-H. He, L. Fan, Let the robot tell: describe car image with natural language via LSTM, Pattern Recognition Letters 98(15)(2017) 75-82.

[32] J. Koutnik, K. Greff, F. Gomez, J. Schmidhuber, A clockwork RNN, in: Proc. 2014 The 31st International Conference on Machine Learning, 2014.

[33] J. Sun, S. Zhang, L. Zhang, Object tracking with spatial context model, IEEE Transactions on Signal Processing Letters 23(5)(2016) 727-731.

[34] M. Grabner, H. Grabner, H. Bischof, Learning features for tracking, in: Proc. 2007 IEEE Conference on Computer Vision and Pattern Recognition, 2007.

[35] S. Salti, A. Cavallaro, L.D. Stefano, Adaptive appearance modeling for video tracking: survey and evaluation, IEEE Transactions on Image Processing 21(10)(2012) 4334 - 4348.

[36] C. Ma, X. Yang, C. Zhang, M.-H. Yang, Long-term correlation tracking, in: Proc. 2015 IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[37] L. Zhang, P.N. Suganthan, Robust visual tracking via co-trained kernelized correlation filters, Pattern Recognition 69(2017) 82-93.

[38] M. Kristan, A. Leonardis, J. Matas et al., The visual object tracking VOT2016 challenge results, in: Proc. 2016 The IEEE International Conference on Computer Vision Workshops on visual object tracking challenge, 2016.

[39] M. Danelljan, F.S. Khan, M. Felsberg, J.v.d. Weijer, Adaptive color attributes for real-time visual tracking, in: Proc. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[40] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, P.H.S. Torr, Staple: complementary learners for real-time tracking, Computer Vision & Pattern Recognition 38(2)(2016) 1401-1409.

[41] G. Zhu, F. Porikli, H. Li, Beyond local search: Tracking objects everywhere with instance-specific proposals, in: Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[42] X. Jingjing, R. Stolkin, A. Leonardis, Single target tracking using adaptive clustered decision trees and dynamic multi-level appearance models, in: Proc. 2015 IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[43] H. Nam, B. Han, Learning multi-domain convolutional neural networks for visual tracking, in: Proc. 2015 CoRR, 2015.

[44] L. Wang, W. Ouyang, X. Wang, H. Lu, Visual tracking with fully convolutional networks, in: Proc. 2015 IEEE International Conference on Computer Vision, 2015.

[45] P. Zhang, T. Zhuo, W. Huang, K.-L. Chen, M. Kankanhalli, Online object tracking based on CNN with spatial-temporal saliency guided sampling, Neurocomputing 25(7)(2017) 115-127.

[46] H. Nam, B. Han, Learning multi-domain convolutional neural networks for visual tracking, in: Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[47] A.S. Montero, J. Lang, R. Laganiere, Scalable kernel correlation filter with sparse feature integration, in: Proc. 2015 The IEEE International Conference on Computer Vision, 2015.

[48] D. Du , H. Qi , L. Wen , Q. Tian , Q. Huang, Geometric hypergraph learning for visual tracking, IEEE Transactions on Cybernetics 99 (2016) 1-14.

[49] J. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE Transactions on Pattern Analysis and Machine Intelligence 37(3)(2015) 583-596.

[50] K. Zhang, L. Zhang, Q. Liu, D. Zhang, M.-H. Yang, Fast visual tracking via dense spatio-temporal context learning, in: Proc. 2014 Computer Vision-ECCV, 2014.

[51] L. Cehovin, A. Leonardis, M. Kristan, Visual object tracking performance measures revisited, IEEE Transactions on Image Processing 25(3)(2016) 1261-1274.

[52] G. Zhu, F. Porikli, H. Li, Beyond local search: tracking objects everywhere with instance-specific proposals, 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[53] M. Danelljan, G. Häger, F.S. Khan, M. Felsberg, Adaptive decontamination of the training set: a unified formulation for discriminative visual tracking, in: Proc. 2016 IEEE Conference on Computer Vision & Pattern Recognition, 2016.

[54] L. Wang, W. Ouyang, X. Wang, H. Lu, Visual tracking with fully convolutional networks, in: Proc. 2015 IEEE International Conference on Computer Vision, 2015.

Appendix: JOC-Author Benefit

Author Contributions: conceptualization, methodology, software and writing, Gang Wu; investigation and resources, Chi-she Wang; supervision, Yong Zhu; project administration, Shou-bao Su.
Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study in the analyses and interpretation of data.