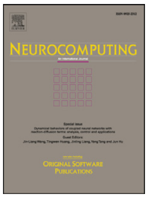
Neurocomputing 281 (2018) 78–85

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



Fusing two-stream convolutional neural networks for RGB-T object tracking

Chenglong Li[a], Xiaohao Wu[a], Nan Zhao[a], Xiaochun Cao[b], Jin Tang[a,*]

[a] School of Computer Science and Technology, Anhui University, Hefei, China
[b] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China

ARTICLE INFO

Article history:
Received 15 February 2017
Revised 22 November 2017
Accepted 28 November 2017
Available online 7 December 2017

Communicated by Dr. Y. Liu

Keywords:
Object tracking
Thermal information
Adaptive fusion
Convolutional neural network
Correlation filter

ABSTRACT

This paper investigates how to integrate the complementary information from RGB and thermal (RGB-T) sources for object tracking. We propose a novel Convolutional Neural Network (ConvNet) architecture, including a two-stream ConvNet and a FusionNet, to achieve adaptive fusion of different source data for robust RGB-T tracking. Both RGB and thermal streams extract generic semantic information of the target object. In particular, the thermal stream is pre-trained on the ImageNet dataset to encode rich semantic information, and then fine-tuned using thermal images to capture the specific properties of thermal information. For adaptive fusion of different modalities while avoiding redundant noises, the FusionNet is employed to select most discriminative feature maps from the outputs of the two-stream ConvNet, and updated online to adapt to appearance variations of the target object. Finally, the object locations are efficiently predicted by applying the multi-channel correlation filter on the fused feature maps. Extensive experiments on the recently public benchmark GTOT verify the effectiveness of the proposed approach against other state-of-the-art RGB-T trackers.

1. Introduction

Visual tracking using RGB and thermal data (called RGB-T tracking in this paper) is a fundamental problem in computer vision, and has received increasing attention due to its potentials on numerous applications, such as robotics, human–computer interaction, and self-driving systems. Given the initial ground truth, the task of RGB-T tracking is to track a particular object in sequential frames by leveraging the complementary benefits from RGB and thermal information. The one key problem of RGB-T tracking is how to leverage the useful modality information while avoiding redundant noises introduced by some individual source.

Some of existing RGB-T tracking methods [1,2] directly concatenate the features of different modalities into a vector, which usually introduces redundant information caused by some individual source. Other methods [3,4] use some weights with manual setting to achieve adaptive integration of RGB and thermal data, but the weights cannot adapt to quality variations of different modalities. Recently, Li et al. [5] introduce the weight variable for each modality, and optimize them online by assuming that they are inversely proportional to the reconstruction residues. These methods, however, may be failed when the above assumption does not hold.

Motivated by these observations and the advances of CNN in visual tracking, in this paper, we propose a novel approach that utilizes a two-stream convolutional neural network (CNN) to learn an adaptive feature representation for RGB-T tracking. Given a bounding box of the target object, we extract the deep features of both RGB and thermal modalities by the two-stream CNN due to the powerful representations for distinguishing generic objects, which is pre-trained on the large-scale ImageNet dataset [6]. The two-stream CNN consists of two sub-networks for effective and efficient tracking. One is the generic sub-network that extracts rich semantic information for representing target objects powerfully, and another is the fusion sub-network that adaptively incorporates the information from multiple modalities. In particular, we first train the thermal stream network using the ImageNet dataset to capture the rich semantic information of objects, and then fine-tune it by thermal data for adapting the thermal source. The fusion sub-network selects the discriminative feature maps from the multiple modality outputs of the two-stream networks to mitigate the effects of redundant information, and thus significantly improve the efficiency with increasing the accuracy. Moreover, we update the fusion sub-network online to adapt the object appearance variations. Finally, we adopt the multi-channel correlation filter algorithm to predict object locations. Extensive experiments

[*] Corresponding author.
E-mail address: jtang99029@foxmail.com (J. Tang).

https://doi.org/10.1016/j.neucom.2017.11.068
0925-2312/© 2017 Published by Elsevier B.V.

on the recently public benchmark GTOT verify the effectiveness of the proposed approach against other state-of-the-art trackers.

This paper makes the following major contributions for RGB-T tracking.

- It proposes a novel approach to extract deep features for effective RGB-T object tracking. Our approach is capable of utilizing the multi-modal information while avoiding the redundant noises. Extensive experiments on the recently public benchmark GTOT verify the effectiveness of the proposed approach against other state-of-the-art RGB-T trackers.
- It designs an effective CNN architecture to improve RGB-T tracking performance in terms of both accuracy and efficiency. In particular, the proposed architecture includes a two-stream sub-network and a fusion sub-network, and can select most discriminative feature maps to makes our tracker more robust and efficient.

2. Related work

In this section, we give a brief review of tracking methods closely related to this work. Comprehensive review on visual tracking methods can be found in [7,8].

RGB-T Tracking. RGB-T tracking [1,3,5,9] has attracted a lot of attentions recently as its potentials in the computer vision community. Conaire et al. [3] that can efficiently combine features for robust tracking, and instantiated the fusion of thermal infrared and visible spectrum features in this framework for automated surveillance applications, while Wu et al. [1] first concatenated the patches from RGB and thermal data into a one-dimensional vector, and then employ the l_1 tracker to achieve RGB-T tracking. A collaborative sparse representation based tracking method and a benchmark had been proposed by Li et al. [5] to jointly optimize sparse codes and the reliable weights of different modalities in an online way.

Deep convolutional neural networks based trackers. Recently, Convolutional Neural Networks(CNN), with their strong capabilities of learning feature representations, have achieved superior performance in visual tracking. Different from the traditional methods with hand-crafted features [10–18], deep networks can directly learn features from raw data without resorting to manual tweaking. A typical convolutional neural network is a variant of feed-forward neural network and training with a large-scale dataset like ImageNet [19]. Hence, numerous approaches have since been proposed to utilized deep features for computer visual problem. Fan et al. [17] used fully convolutional network for human tracking, which took the whole frame as input and predicted the foreground heat map by one-pass forward propagation. Meanwhile Li et al. [20] proposed a deep tracking framework using a candidate pool of multiple CNNs. Besides, a deep antoencoder is first pretrained off-line and then fine-tuned for binary classification in online tracking in [21].

Correlation filters based trackers. Correlation filters have gained attention in the area of visual tracking due to its highly computational efficiency and competitive performance on benchmark tracking datasets [22–24]. The correlation filters approximate the dense sampling scheme by a circulant matrix and its regression model can be computed in the Fourier domain [25]. Danelljan et al. [26] introduced the Spatially Regularized Discriminant Correlation Filter (SRDCF) to allow the expansion of the training and search regions without increasing the effective filter size. Henriques et al. [25] formulated kernelized correlation filters(KCF) using circulant matrices, and efficiently incorporated multi-channel features in a Fourier domain. The work in [27] achieved significant improvement by investigating the use of multiple color features in the discriminant correlation filter framework.

Two-stream CNN. Recently, two-stream CNN is proposed for video analysis, such as action recognition and video classification. Simonyan et al. [28] achieved most competitive performance by training two ConvNet on spatial and temporal streams for action recognition in videos, and Feichtenhofer et al. [29], fused spatial and temporal cues at several levels of granularity in feature abstraction. Hao et al. [30] in-depth study on various implementation choices of deep learning based video classification, which examined the performance of the spatial and temporal streams separately and jointly with different network architectures under various sets of parameters.

3. RGB-T tracking via two-stream CNN

In this section, we describe the details of the proposed method. Fig. 1 shows an overview of the proposed tracking framework. First, we use the generic two-stream sub-networks to extract object feature representations for RGB and thermal sources, and then employ the fusion sub-network to select the most discriminative feature maps with removing redundant noise features. Finally, the object location is predicted by adopting the multi-channel correlation filter on the selected features.

3.1. Two-stream CNN

Our two-stream CNN architecture consists of two components, i.e., generic sub-networks and fusion sub-network.

3.1.1. Generic sub-network

Most of trackers utilize hand-crafted features to describe the target, which are incapable of capturing the rich semantic information of the target and easily lead to tracking failure in challenging conditions. Therefore, we employ the convolutional neural networks (CNN) to extract object features.

For RGB source, image content includes the appearance information, such as shape and color, while reflects the thermal distribution of scenes and objects for thermal source. Therefore, we utilize two-stream CNN to extract the specific features for different modalities, where one CNN is applied to process the RGB stream and the other CNN is used to handle the thermal stream.

The illustration of these two streams is shown in Fig. 1. The CNN of both two streams have the similar structure as a general deep CNN for image classification, VGG-16 [31] in this paper. They directly take individual object patches as network inputs followed by 13 convolutional layers and 4 pooling layers. Herein, we remove the fully connected layers for tracking purpose.

For capturing different properties from RGB and thermal data, we adopt different strategies for training CNN. For RGB stream, we train the CNN on the large-scale dataset, ImageNet [6] to encode the rich semantic information. For thermal stream, we fine-tune the Thermal-network with classification tasks to extract feature representations, and this way is widely used in visual tracking [21,32].

3.1.2. Fusion sub-network

As discussed in [5], the multi-modal features usually include some redundant noises, which affect the performance of RGB-T tracking. Therefore, we propose a fusion sub-network to remove the useless features for accuracy and efficiency.

Our fusion network consists of two separate convolutional layers, and Rectified Linear Units (ReLU) is chosen as the non-linearity for these layers. The first convolutional layer has convolutional kernel of size 9 × 9 and outputs 512 feature maps as the input to the next convolutional layer. The other convolutional layer has convolutional kernel of size 5 × 5 and outputs the heat map of the input patches.

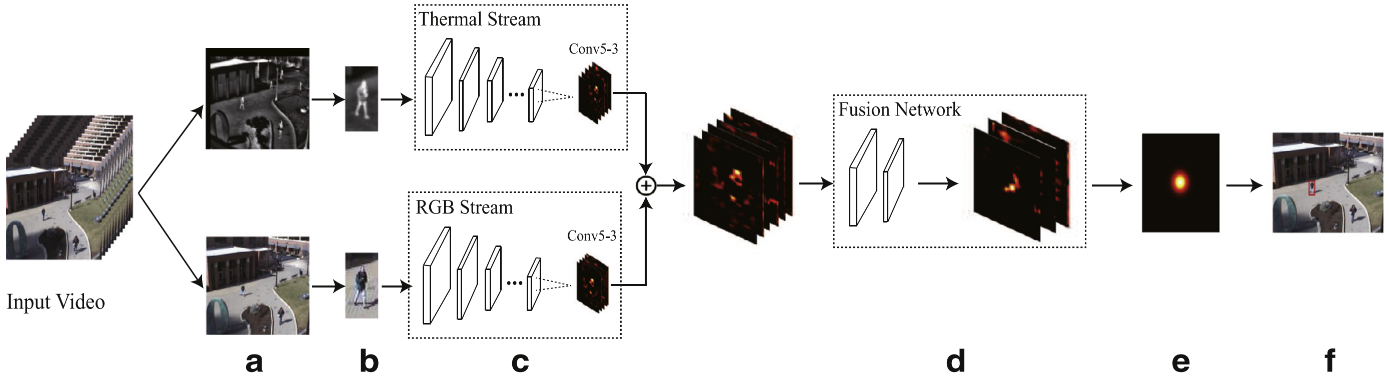


Fig. 1. Pipeline of our method. (a) Input frames. (b) Target patch. (c) Generic network. (d)Feature map selection. (e) Correlation filter. (f) Tracking results.

Table 1
The running time of the compared trackers and the bold fonts of results indicate the best performance.

	Ours	CSR [5]	Struck [12]	SCM [15]	RPT [32]	MEEM [36]
Code Type	MATLAB	MATLAB&C++	C++	MATLAB&C++	MATLAB&C++	MATLAB&C++
FPS	**15**	1.6	10.8	0.3	2.6	4.9

The fusion network initialized in first frame by minimizing the following square loss function:

$$\mathcal{L} = \mathcal{L}_{\mathcal{R}} + \mathcal{L}_{\mathcal{T}} \tag{1}$$

$$\mathcal{L}_{\mathcal{R}} = \| \hat{X} - X_{\mathcal{R}} \|^2 \tag{2}$$

$$\mathcal{L}_{\mathcal{T}} = \| \hat{X} - X_{\mathcal{T}} \|^2 \tag{3}$$

where the subscript $\mathcal{R}$ and $\mathcal{T}$ indicates the two different modalities of RGB and Thermal, respectively. $\hat{X}$ represents the heat map generated by the fusion network, and X is the target heat map initialized using the ground truth bounding box in the first frame. We utilize the feature maps extracted by the generic network to train the fusion network.

3.1.3. Feature map selection

We denote the feature maps of the outputs of generic subnetworks as $\mathbf{F} \in R^{d \times n}$, where each feature map is reshaped into a d-dimensional vector and n denotes the number of feature maps. We define the significance of the each feature map as the changes of the loss function caused by the impact of the $\mathbf{f}_i$, the ith column of $\mathbf{F}$. According to [33,34], the changes of the loss function caused by the perturbation of the feature map $\Delta\mathbf{F}$ can be computed by a two-order Taylor expansion as follows:

$$\Delta\mathcal{L}_{\mathcal{R}} = \sum_i g_i^{\mathcal{R}} \Delta\mathbf{f}_i + \frac{1}{2}\sum_i h_{ii}^{\mathcal{R}} (\Delta\mathbf{f}_i)^2 + \frac{1}{2}\sum_{i \neq j} h_{ij}^{\mathcal{R}} \Delta\mathbf{f}_i \Delta\mathbf{f}_j \tag{4}$$

$$\Delta\mathcal{L}_{\mathcal{T}} = \sum_i g_i^{\mathcal{T}} \Delta\mathbf{f}_i + \frac{1}{2}\sum_i h_{ii}^{\mathcal{T}} (\Delta\mathbf{f}_i)^2 + \frac{1}{2}\sum_{i \neq j} h_{ij}^{\mathcal{T}} \Delta\mathbf{f}_i \Delta\mathbf{f}_j \tag{5}$$

$$g_i^{\mathcal{R}} = \frac{\partial \mathcal{L}_{\mathcal{R}}}{\partial \mathbf{f}_i}, h_{ij}^{\mathcal{R}} = \frac{\partial^2 \mathcal{L}_{\mathcal{R}}}{\partial \mathbf{f}_i \partial \mathbf{f}_j} \tag{6}$$

$$g_i^{\mathcal{T}} = \frac{\partial \mathcal{L}_{\mathcal{T}}}{\partial \mathbf{f}_i}, h_{ij}^{\mathcal{T}} = \frac{\partial^2 \mathcal{L}_{\mathcal{T}}}{\partial \mathbf{f}_i \partial \mathbf{f}_j} \tag{7}$$

where g_i and h_{ij} represents the first and second order derivatives of the loss function with respect to the input feature maps, respectively. Hence, we can further define the change of the loss function as

$$\Delta\mathcal{L} = \Delta\mathcal{L}_{\mathcal{R}} + \Delta\mathcal{L}_{\mathcal{T}} \tag{8}$$

The Hessian matrix is enormous when the number of feature maps is large and thus its computation is too time consuming. Therefore, we can use some simplifying approximations. The diagonal approximation assumes that the total loss caused by the changes of feature map is the sum of the individual loss caused by changes of each feature maps. In other words, the last terms of Eqs (4) and (5) are discarded [33,34]. In addition, we define the significance of $\mathbf{f}_i$ as the changes of loss function after setting $\mathbf{f}_i$ to 0, *i.e.*, $\Delta\mathbf{f}_i = 0 - \mathbf{f}_i$. Hence, the significance of $\mathbf{f}_i$ can be computed as following equation:

$$s_i = g_i^{\mathcal{R}} \Delta\mathbf{f}_i + \frac{1}{2} h_{ii}^{\mathcal{R}} (\Delta\mathbf{f}_i)^2 + g_i^{\mathcal{T}} \Delta\mathbf{f}_i + \frac{1}{2} h_{ii}^{\mathcal{T}} (\Delta\mathbf{f}_i)^2 \tag{9}$$

$$S_k = \sum_{x,y} s(x, y, k) \tag{10}$$

where the significance of the kth feature map are further defined as the summation of significance of all its elements, *i.e.*, S_k; where $s(x, y, k)$ is the significance of the element indexed by location (x, y) on the kth feature map. Then, the significance of each feature map computed by Eq. (10) and all the feature maps are sorted in the descending order by their significance. The top K feature maps are selected, denoting as $\mathbf{Z}$ ($\mathbf{Z} \in R^{d \times K}$).

3.2. RGB-T object tracking

3.2.1. Object localization

Our method is built on KCF tracker [25], promising performance among the recent top-performing trackers in terms of both accuracy and efficiency. The key of KCF tracker is that the augmentation of negative samples are employed to enhance the discriminative ability of the track-by-detector scheme while exploring the structure of the circulant matrix for the high efficiency.

We integrate the features into the multi-channel correlation filter to localize the target. Denoting $\mathbf{z}$ as one feature map with the size $W \times H$, where W, H indicates the width and the height of the target image patch, respectively. All the circular shifts of $\mathbf{z}$ along the W and H dimensions that we consider as the training samples, and each shifted sample $\mathbf{z}_{w,\,h}$ have a label $y(w,\ h)$. Then, the correlation filter $\mathbf{w}$ is learned by solving the following minimization problem,

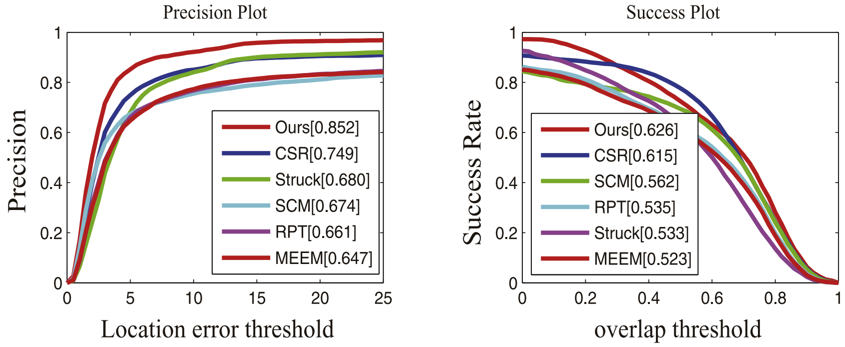


Fig. 2. The precision and success plots of our method against other baseline trackers.

Table 2
Precision scores on different attributes: deformation(DEF), fast motion(FM), low illumination(LI), large scale variation(LSV), occlusion(OCC), small object(SO), thermal crossover(TC). The red color indicate the best performance, the green fonts indicate the second best one, and the third best results in blue colors.

	DEF	FM	LI	LSV	OCC	SO	TC
Ours	0.921	0.764	0.885	0.835	0.780	0.895	0.825
CSR [5]	0.677	0.761	0.731	0.800	0.791	0.802	0.816
Struck [12]	0.756	0.639	0.740	0.660	0.674	0.745	0.680
SCM [15]	0.596	0.691	0.669	0.795	0.717	0.721	0.688
RPT [32]	0.671	0.588	0.638	0.691	0.679	0.649	0.654
MEEM [36]	0.688	0.686	0.665	0.628	0.684	0.793	0.696

Table 3
Precision score/success score(PS/SS) of the proposed method and the deep learning baselines.

	Ours	Deep-DSST [39]	Deep-KCF [25]
Precision Score	0.852	0.77	0.75
Success Score	0.626	0.56	0.47

$$\mathbf{w}^{*} = \arg\min_{\mathbf{w}} \sum_{w,h} \| \langle \mathbf{w}, \phi(\mathbf{z}_{w,h}) \rangle - y(w,h) \|^2 + \lambda \| \mathbf{w} \|_2^2 \qquad (11)$$

where ϕ denotes the mapping to a kernel space and λ is a parameter($\lambda \geq 0$) to control the impact of the regularization term. $(w, h) \in \{0, 1, ..., W-1\} \times \{0, 1, ..., H-1\}$. The inner product is induced by a linear kernel in the Hilbert space, i.e.,$\langle \mathbf{w}, \phi(\mathbf{z}_{w,h}) \rangle = \sum_{i=1}^{K} \mathbf{w}_{w,h,i}^{\top} \phi(\mathbf{z}_{w,h,i})$. And, $y(w,h) = e^{-\frac{(w-W/2)^2+(h-H/2)^2}{2\delta^2}}$, where the δ denote as the kernel width of Gaussian function. To find an approximate solution of Eq. (11), inspired by Henriques et al. [25], Ma et al. [35], we solve Eq. (11) in each individual feature channel using fast Fourier transformation(FFT). The learned filter in the frequency domain on the ith channel can be solved as following,

$$\mathbf{w}^{i} = \frac{\mathbf{y} \odot \phi(\bar{\mathbf{z}}^{i})}{\sum_{i=1}^{K} \mathbf{z}^{i} \odot \phi(\bar{\mathbf{z}}^{i}) + \lambda} \qquad (12)$$

where $\mathbf{y}$ is the Fourier transformation form of $\mathbf{y} = \{y(w,h) | (w,h) \in \{0, 1, ..., W-1\} \times \{0, 1, ..., H-1\}\}$, and the bar denotes complex conjugation and $\odot$ denotes element-wise product. To locate the target at the next frame, and the correlation response map is computed based on the selected feature maps $\mathbf{x}$ in the new frame and the multi-channel correlation filter:

$$R = \mathscr{F}^{-1}\{\sum_{i=1}^{K} \mathbf{w}^{i} \odot \phi(\bar{\mathbf{x}}^{i})\} \qquad (13)$$

where $\mathscr{F}^{-1}$ denotes the inverse FFT transform. Thus, the target location can be estimated by searching for the position of maximum value of the correlation response map R.

3.2.2. Online update

We further propose an online strategy to update the fusion network in order to adapt the object appearance variations. In this paper, we update the fusion network using the results of tracking in the current frame by the following:

$$\mathcal{L} = \delta \| \mathcal{W} \|^2 + \| \hat{X}(x,y) - X_{\mathcal{R}}(x,y) \|^2 + \| \hat{X}(x,y) - X_{\mathcal{T}}(x,y) \|^2 \qquad (14)$$

where (x, y) are the spatial coordinates; $X(x, y)$ is the target patch in the current frame; $\hat{X}(x,y)$ is the heat map generated by the selection network according to the result of tracking in the current frame; δ is the trade off parameter for weight decay; and $\mathcal{W}$ is the weight parameter of the convolutional layers.

4. Experiments

In this section, we give the details of our experimental implementation and discuss the results of tracking performance evaluation.

4.1. Evaluation settings

Experimental setup. We implemented the proposed tracker in MATLAB based on the wrapper of Caffe framework [37], run on a TITAN X GPU with 12G RAM, and the proposed tracker performs at about 15 FPS (frames per second). We also present the runtime comparison of our tracker against other methods in Table 1.

Table 4
Precision score/success score(PS/SS) of the proposed method with different version.

	Ours	Ours-RGB	Ours-Thermal	Ours-W	Ours-NoUp
Precision Score	0.852	0.824	0.812	0.814	0.836
Success Score	0.626	0.596	0.602	0.593	0.614

Table 5
Success scores on different attributes: deformation(DEF), fast motion(FM), low illumination(LI), large scale variation(LSV), occlusion(OCC), small object(SO), thermal crossover(TC). The red color indicate the best performance, the green fonts indicate the second best one, and the third best results in blue colors.

	DEF	FM	LI	LSV	OCC	SO	TC
Ours	0.734	0.517	0.642	0.646	0.556	0.591	0.595
CSR [5]	0.561	0.653	0.605	0.646	0.633	0.608	0.643
Struck [12]	0.604	0.517	0.553	0.496	0.515	0.527	0.510
SCM [15]	0.499	0.589	0.561	0.516	0.570	0.531	0.555
RPT [32]	0.576	0.463	0.521	0.517	0.510	0.478	0.509
MEEM [36]	0.577	0.523	0.518	0.462	0.530	0.496	0.546

The state of the target (i.e., size and location) in the first frame is given by the ground truth. The number of feature maps selected by the proposed feature selection method is set to $K = 600$. For convolutional features extraction, we crop an image patch with 2.5 times the size of the target bounding box and then resize it into 224×224 pixels for the generic network, and we use the feature maps of conv4-3 layer. The Gaussian kernel is used in the correlation filter.

Fine-tune sub-network. For many deep learning-based tracking algorithms [35,38], they train CNN using large datasets in the classification task, such as ImageNet, to extract feature representations. Therefore, we employ the similar way to fine-tune the thermal-stream network as follows. First, we collect a set of thermal images, each of which contains one kind of object, and annotate them with class labels. Second, we employ some operations, such as noise addition and rotation, to perform data augmentation. Finally, we use these annotated images to fine-tune the thermal-stream network.

GTOT benchmark. To evaluate the effectiveness of the proposed method, we conduct experiments on the GTOT benchmark [5]. The dataset consists of 50 aligned RGB-T video sequences with ground truth object locations, and the sequences are categorized with 7 attributes based on different challenging factors, including occlusion, large scale variation, fast motion, low illumination, thermal crossover, small object and deformation. To better analyze the strength and weakness of the proposed method, we compare with five RGB-T trackers implemented in GTOT, including CSR [5], Struck [12], SCM [15], RPT [32], and MEEM [36].

Evaluation metrics. For quantitative comparison, we employ the precision score and success score used in GTOT [5].

Precision score. The precision score demonstrates the percentage of frames where the distance between the predicted target location and the ground truth is within a given threshold. Since many targets in the benchmark [5] are small and all the trackers are ranked according to the precision scores at the threshold of 5 pixels.

Success score. The success rate is another effective evaluation metric, and it computed as the percentage of frames in a sequence where the intersection-over-union overlap with the ground truth bounding box is larger than a threshold $\tau \in [0, 1]$. We define the r_o and the r_g as the output bounding box and the ground truth bounding box, respectively. Hence, the overlap score define as $S = \frac{|r_o \bigcap r_g|}{|r_o \bigcup r_g|}$, where $\bigcap$ and $\bigcup$ indicates the intersection and union operations, the $|\cdot|$ denote the number of pixels in the region. In this work, we employ the area under curve of the success curve to represent the success score, which the curve is plotted by the success rate.

4.2. Comparison results

The precision plots and success plots of the compared tracking approaches on the GTOT dataset [5] are presented in Fig. 2. From Fig. 2 we can see that the proposed tracking method outperforms all the other tracking approaches in terms of precision score and success score. In particular, the proposed method achieves the precision score of 85.2%, outperforming CSR [5], the second best RGB-T tracker, by 10.3%.

We also evaluate the proposed tracker with other deep learning based methods, including Deep-DSST [39] and Deep-KCF [25], and Table 3 presents the comparison results. Overall, our tackers achieves a superior performance than these deep learning baselines in precision score and success score. It demonstrates the effectiveness of the proposed approach.

To comprehensively evaluate the robustness of the proposed approach, we further evaluate all the tracking methods on sequences with 7 attributes, as shown in Table 2 and Table 5, and we find that the precision score of our method is higher than other methods in many challenges. In particular, the precision score reaches 92.1% and 89.5% when the proposed method handle in the scenes with deformation and small object, respectively. Furthermore, we also present the tracking performance from several challenging scenarios in Fig. 3. Fig. 3 shows that these state-of-the-art trackers are unable to handle these complicated scenarios well. For example, most of compared trackers lose the target when the background is cluttered, as shown in the sequence *occBike* and *Otcbvs*. In contrast, the proposed approach can localize the targets more accurately. Although our method perform well in several scenarios, however, for the video sequences with the fast motion challenge, our method can not handle well. And we also note that our method has failure in handle the occlusion challenge. It is accounted to the weakness of our search strategy, and we will consider the robust motion or search models to leverage more temporal and spatial information in future work.

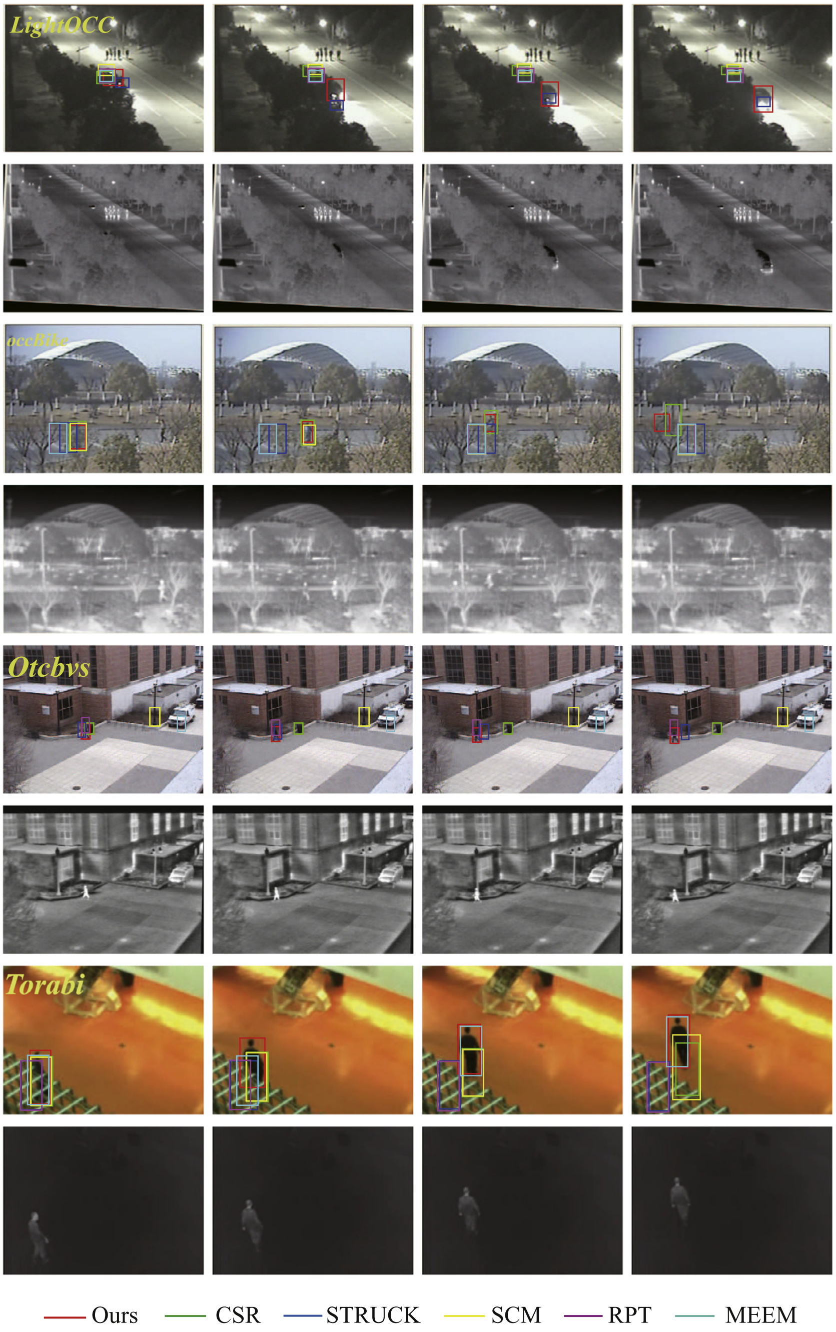


Fig. 3. Some frames of our method and five other methods, CSR [5], Struck [12], SCM [15], RPT [32], MEEM [36], on three challenging videos.

4.3. Component analysis

To facilitate better analysis the effectiveness of the proposed trackers, we conducted several additional comparative experiments, including: i) Ours-RGB, the RGB images only as the input and the framework without fusion sub-network. ii) Ours-Thermal, the Thermal images only as the input and the framework without fusion sub-network. iii) Ours-W, that removes the fusion sub-network in our framework. It directly concatenate the features of different modalities into a vector, followed by the correlation filter to perform tracking.iv) Ours-NoUp, the framework without update strategy v) The different value of *K*.

The evaluation results are shown in Table 4 and Table 6, the tables shows that the precision scores and success scores on the benchmark [5]. We can draw the following conclusions from Table 4 and Table 6: 1) By using the feature map selection, the proposed tracker achieves the performance gain of 3.8% in precision over Ours-W, justifying the importance of the fusion sub-network. 2) Considering both effects of two modalities is better than that of the single modality. 3) The proposed update strategy can adapt the object appearance variations, and the strategy can improve the

Table 6
Precision score/success score(PS/SS) of the proposed method with different *K*.

	Ours(K=600)	K=1024	K=800	K=400
Precision score	0.852	0.814	0.836	0.828
Success score	0.626	0.593	0.610	0.609

tracking performance. 4) The selection of K affects the tracking performance, and it is significant to select appropriate K to improve the tracking accuracy.

5. Conclusion

In this paper, we have proposed an effective approach for RGB-T tracking in CNN framework. The proposed method employs the CNN to extract object feature representations, and we jointly consider the features of different modalities. A fusion network is then designed to fuse the feature maps of different modalities and select the discriminative features to remove modality noises and reduce the computation redundancy for robust tracking. To validate the effectiveness of our proposed tracker, we perform comprehensive experiments on public online benchmark and our tracking approach outperforms several state-of-the-art RGB-T trackers. In future work, we will consider the robust motion or search models that leverage more temporal and spatial information, and integrate more modalities, such as depth and near-infrared, into our framework to perform more robust tracking.

Acknowledgement

This work was supported by the National Natural Science Foundation of China under Grant 61702002, Grant 61472002 and Grant 61602006, in part by the Natural Science Foundation of Anhui Higher Education Institution of China under Grants KJ2017A017 and KJ2015A110, and in part by the Co-Innovation Center for Information Supply & Assurance Technology, Anhui University under Grant Y01002449.

References

[1] Y. Wu, E. Blasch, G. Chen, L. Bai, H. Ling, Multiple source data fusion via sparse representation for robust visual tracking, in: Proceedings of the IEEE International Conference on Information Fusion, 2011.

[2] H.P. Liu, F.C. Sun, Fusion tracking in color and infrared images using joint sparse representation, Sci. China Inf. Sci. 55 (3) (2012) 590–599.

[3] C.Ó. Conaire, N.E. O'Connor, A. Smeaton, Thermo-visual feature fusion for object tracking using multiple spatiogram trackers, Mach. Vis. Appl. 19 (5–6) (2008) 483–494.

[4] C.O. Conaire, N. O'Connor, E. Cooke, A.F. Smeaton, Comparison of fusion methods for thermo-visual surveillance tracking, in: Proceedings of the IEEE International Conference on Information Fusion, 2006.

[5] C. Li, H. Cheng, S. Hu, X. Liu, J. Tang, L. Lin, Learning collaborative sparse representation for grayscale-thermal tracking, IEEE Trans. Image Process. 25 (12) (2016) 5743–5756.

[6] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2012.

[7] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, A.V.D. Hengel, A survey of appearance models in visual object tracking, ACM Trans. Intell. Syst. Technol. (TIST) 4 (4) (2013) 58.

[8] A. Yilmaz, O. Javed, M. Shah, Object tracking: a survey, ACM Comput. Surv. 38 (4) (2006) 13.

[9] C. Li, S. Hu, S. Gao, J. Tang, Real-time grayscale-thermal tracking via laplacian sparse representation, in: Proceedings of the IEEE International Conference on Multimedia Modeling, 2016.

[10] S. Avidan, Support vector tracking, IEEE Trans. Pattern Anal. Mach. Intell. 26 (8) (2004) 1064–1072.

[11] Z. Kalal, J. Matas, K. Mikolajczyk, Pn learning: bootstrapping binary classifiers by structural constraints, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010.

[12] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M.-M. Cheng, S.L. Hicks, P.H. Torr, Struck: structured output tracking with kernels, IEEE Trans. Pattern Anal. Mach. Intell. 38 (10) (2016) 2096–2109.

[13] X. Jia, H. Lu, M.-H. Yang, Visual tracking via adaptive structural local sparse appearance model, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012.

[14] T.B. Dinh, N. Vo, G. Medioni, Context tracker: exploring supporters and distracters in unconstrained environments, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2011.

[15] W. Zhong, H. Lu, M.-H. Yang, Robust object tracking via sparsity-based collaborative model, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012.

[16] K. Zhang, L. Zhang, M.-H. Yang, Fast compressive tracking, IEEE Trans. Pattern Anal. Mach. Intell. 36 (10) (2014) 2002–2015.

[17] J. Gao, H. Ling, W. Hu, J. Xing, Transfer learning based visual tracking with gaussian processes regression, in: Proceedings of the IEEE European Conference on Computer Vision, 2014.

[18] Y.X.S.D. Liu Shuang, Y. Zhang, J.J. Zhang, Robust facial landmark detection and tracking across poses and expressions for in-the-wild monocular video, Comput. Vis. Media 3 (1) (2017) 33–47.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009.

[20] H. Li, Y. Li, F. Porikli, Deeptrack: learning discriminative feature representations online for robust visual tracking, IEEE Trans. Image Process. 25 (4) (2016) 1834–1848.

[21] N. Wang, D.-Y. Yeung, Learning a deep compact image representation for visual tracking, in: Proceedings of Advances in Neural Information Processing Systems, 2013.

[22] Y. Wu, J. Lim, M.-H. Yang, Online object tracking: a benchmark, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013.

[23] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernandez, T. Vojir, G. Hager, G. Nebehay, R. Pflugfelder, The visual object tracking vot2015 challenge results, in: Proceedings of the IEEE International Conference on Computer Vision, 2015.

[24] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, D. Tao, Multi-store tracker (muster): a cognitive psychology inspired approach to object tracking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[25] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE Trans. Pattern Anal. Mach. Intell. 37 (3) (2015) 583–596.

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

[27] M. Danelljan, F. Shahbaz Khan, M. Felsberg, J. Van de Weijer, Adaptive color attributes for real-time visual tracking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[28] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, in: Proceedings of the Advances in Neural Information Processing Systems, 2014.

[29] C. Feichtenhofer, A. Pinz, A. Zisserman, Convolutional two-stream network fusion for video action recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[30] H. Ye, Z. Wu, R.-W. Zhao, X. Wang, Y.-G. Jiang, X. Xue, Evaluating two-stream cnn for video classification, in: Proceedings of the IEEE International Conference on Multimedia Retrieval, 2015.

[31] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proceedings of the International Conference on Learning Representations, 2015.

[32] Y. Li, J. Zhu, S.C. Hoi, Reliable patch trackers: robust visual tracking by exploiting reliable patches, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.

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

[34] S.R.Y. LeCun, J.S. Denker, L.D. Jackel, Optimal brain damage, in: Proceedings of the Advances in Neural Information Processing Systems, 1989.

[35] C. Ma, J.-B. Huang, X. Yang, M.-H. Yang, Hierarchical convolutional features for visual tracking, in: Proceedings of the IEEE International Conference on Computer Vision, 2015.

[36] J. Zhang, S. Ma, S. Sclaroff, Meem: robust tracking via multiple experts using entropy minimization, in: Proceedings of the IEEE European Conference on Computer Vision, 2014.

[37] J. Yangqing, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional architecture for fast feature embedding, in: Proceedings of the 22nd ACM international conference on Multimedia, 2014.

[38] Y. Qi, S. Zhang, L. Qin, H. Yao, Q. Huang, J. Lim, M.-H. Yang, Hedged deep tracking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

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

Chenglong Li received the M.S. and Ph.D. degrees from the School of Computer Science and Technology, Anhui University, Hefei, China, in 2013 and 2016 respectively. From 2014 and 2015, he was a visiting student with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. He is currently a lecturer at School of Computer Science and Technology, Anhui University, and also a postdoctoral research fellow at the Center for Research on Intelligent Perception and Computing (CRIPAC), National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), China. He was a recipient of the ACM Doctoral Dissertation Award in 2016.

Xiaohao Wu received the B.S. degree in Anhui University, Hefei, China. He is currently pursuing a M.S. degree in Anhui University. From Sep. 2016 to Feb. 2017, he worked as a visiting student in the State Key Laboratory of Information Security, Institute of information Engineering, Chinese Academy of Sciences (CAS), China. His current research interests include computer vision, deep learning and machine learning.

Nan Zhao received the B.S. degree from the School of Computer Science and Technology, Anhui University, Hefei, China, in 2015. He is currently pursuing the M.S. degree in the School of Computer Science and Technology from Anhui University. His current research interest is computer vision with a focus on visual tracking and multi-model object tracking.

XiaoChun Cao received the B.E. and M.E. degrees in computer science from Beihang University, Beijing, China, and the Ph.D. degree in computer science from the University of Central Florida, Orlando, FL, USA. He has been a Professor with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China, since 2012. He spent about three years with ObjectVideo Inc., as a Research Scientist. From 2008 to 2012, he was a Professor with Tianjin University, Tianjin, China. He has authored and co-authored over 120 journal and conference papers. He is a fellow of the IET. He is on the Editorial Board of the IEEE TRANSACTIONS OF IMAGE PROCESSING. His current research interests include computer vision, pattern recognition, and machine learning.

Jin Tang received the B.Eng. Degree in automation and the Ph.D. degree in computer science from Anhui University, Hefei, China, in 1999 and 2007, respectively. He is currently a Professor with the School of Computer Science and Technology, Anhui University. His current research interests include computer vision, pattern recognition, and machine learning.