IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 8, 2019, accepted September 26, 2019, date of publication September 30, 2019,
date of current version October 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2944649

Weighted Optical Flow Prediction and Attention Model for Object Tracking

WENMING CAO[1,2,3], YUHONG LI[1], AND ZHIQUAN HE[1,2,4]

[1]Shenzhen Key Laboratory of Media Security, Shenzhen University, Shenzhen 518060, China
[2]Guangdong Multimedia Information Service Engineering Technology Research Center, Shenzhen 518060, China
[3]Video Processing and Communication Lab, Department of Electrical and Computer Engineering, University of Missouri, Columbia, MO 65211, USA
[4]Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen 518060, China

Corresponding author: Zhiquan He (zhiquan@szu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61771322, Grant 61971290, and Grant 61375015, and in part by the Shenzhen Foundation fund under Grant JCYJ20160307154630057.

ABSTRACT Object tracking has been a hot computer vision topic for many years. Although great process has been made, it still has large room to improve because of the complexity of the natural scene and the multiple interference. In this work, we improve the object tracking performance in two ways. First, a sequential scoring model is proposed to integrate the optical flow information of history video frames into the feature map of current frame. Second, an attention model with optical flow information is used for further improvement by differentiating the contribution of different positions in the template to the final response map. On the other hand, the entire model are end-to-end trainable. We test the methods on OTB (Object Tracking Benchmark) and VOT (Visual Object Tracking) tracking datasets. The experimental results demonstrate that the improved tracking accuracy and robustness to occlusion, strenuous motion and vanishing objects.

INDEX TERMS Object tracking, optical flow, attention model, Siamese network.

I. INTRODUCTION

Object tracking has long been a challenging and hot research problem in computer vision, it requires knowledge and methods in different fields such as image processing, pattern recognition, artificial intelligence, deep learning, and fuzzy theory. Object tracking has broad applications in areas of visual navigation, traffic monitoring, military guidance, astronomical observation and meteorological analysis [1]. To track the target in videos, the algorithm usually consists of two models, appearance model and motion model. The appearance model can be further divided into two categories, i.e. generative model [2] and discriminant model [3]. The generative model maintains a target template by learning its features online and then search for the optimal image region that best matches the template. The corresponding region is the predicted position of the target. The discriminant model considers the tracking process as a binary classification problem which extracts the features from the target and background to train a classifier, which is used to separate the target from the image background of video frames. Many tracking algorithms have been developed to attack the problem, for example, classic tracking methods such as Meanshift [4], [5], Kalman Filter [6] and Particle Filter [7]; framework improvement algorithms such as TLD (Track By Detection) [8] and correlation filtering algorithms. Deep convolutional neural networks have achieved dramatic progress and made tremendous contributions to many important areas of computer vision and machine learning, including image classification, object detection, recognition, and semantic segmentation [9]–[15]. In object tracking, Deep convolutional neural networks have been successfully applied such as DeepSRDCF [16], CCOT [17] and ECO [18].

As one of the fundamental tracking frameworks, Correlation Filter has become a research hot topic and received extensive attention. This method generates high response values for the object and low response for the background. MOSSE filter is one of the earliest correlation based methods, which uses an adaptive training strategy and realizes real-time and robust tracking with variations in lighting, scale, pose, and non-rigid deformations [19]. After that, a series of improvement methods have been published. CSK [20] and KCF [21] were

The associate editor coordinating the review of this manuscript and approving it for publication was Guitao Cao.

proposed by Oxford University researchers, which extract the gradient histogram of the target based on ridge regression and closed-form kernel solution. Starting from KCF, Martin Danelljan studied the contribution of colors in tracking problem and proposed adaptive low-dimensional variant of color attributes to improve the tracking performance [22]. Danelljan *et al*. investigated the problem of accurate and robust scale estimation in a tracking-by-detection framework and proposed a scale adaptive tracking approach by learning separate discriminative correlation filters for translation and scale estimation [23]. In order to deal with fast object motion in tracking, the SRDCF [24] and CFLM [25] were proposed. SRDCF ignores some overflowing pixels in samples and sets the filter coefficient to 0, while CFLM uses larger scale detection image blocks and smaller filters to increase the sample ratio.

Since 2015 in-depth study of deep learning has given rise to the fast development of various fields of computer vision, including image recognition, object tracking etc. The advantage of deep learning is its strong ability to learn large spectrum of image features from low-level edges, corners and intensities to high-level semantic representations. DLT [26] is the first deep network framework to use offline training and online fine-tuning, which has achieved good results in the OTB dataset [27], but it only uses low-level features and easily causes vague features. DeepSRDCF [16] improves the tracking performance using high-level features, but the computation time increases inevitably. In [18], Martin Danelljan optimized the speed and real-time capability of Discriminative Correlation Filter (DCF) based methods by introducing efficient factorized convolution operators and a conservative model update strategy. Most of the tracking algorithms update the weights of the convolution neural network (CNN) online during the tracking, which is computationally intensive, making it difficult to use in real-time applications. Luca Bertinetto from Oxford University put forward a basic tracking framework: SiamFC [28]–[30], which trains the model offline to learn the similarity to the initial frame using available training dataset and detects the target online by mutual convolution. These methods have simple structures and strong portability.

Object tracking can be difficult when the tracking target is occluded by other unrelated objects, moving strenuously or is vanished. Tracking algorithms should be robust to these situations [31]. Template update strategy plays an important role in target tracking. Usually, the template is either fixed, updated statically or dynamically. Many methods fix the initial template frame during the tracking process, which limits its contribution in the final response. In order to solve these problems, in this work, we propose a novel method to integrate the optical flow network and the attention mechanism with SiamFC framework and achieve end-to-end training. The framework is shown in Fig. 1. The optical flow information of moving objects is an important representation of object motions, which has high phase reliability and strong robustness to illumination changes. And the attention model works in a similar way to the attention mechanism of human

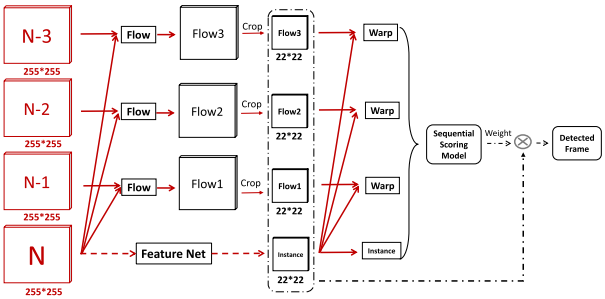

FIGURE 1. Framework of fusing optical flow frames with current frame using sequential scoring model.

brains, which can automatically focus the attention on the most informative object area. We use a sequential score model to assign weights to the past frames to indicate the correlation of the frame withe the current frame. The weights are used to combine the optical flow information of past frames with the feature map of the current frame. Besides this, we use an attention model on the template image to improve the feature map, considering the fact the many.

We have tested the proposed method on OTB [27] and VOT [32] benchmark dataset and achieved better results than methods being compared. The contribution of this paper is summarized as follows:

1) We design a novel sequential scoring model to aggregate the current and warped optical flow feature maps of past frames to increase the tracking stability.
2) We propose a novel optical flow attention model according to the moving direction of optical flow of the adjacent frame, to enhance the feature expression capacity of the template frame.
3) We compare the effect of updating template frame in different ways for tracking methods based on the Siamese framework, which demonstrates that fixed template method is the best way for this sort of framework.

The rest of the paper is organized as follows. Section II presents the related work. Section III explains the main method of our work. Experiments and results are presented in Section IV. And Section V concludes the paper.

II. RELATED WORK

A. SIAMFC TRACKER

In the past a few years, the deep learning algorithm has become a dominant method of single target tracking due to the excellent tracking accuracy. However, since most of the algorithms update the network weights online during the tracking process, the test speed becomes a major problem in real-time applications. Luca Bertinetto from Oxford University put forward a basic tracking framework, i.e., the fully convolution Siamese network, which is a similarity based method with an model trained offline with the initial frame and detects target position online while tracking. This network SiamFC is mainly trained on the ILSVRC15 video object detection

dataset [33] and has achieved good results in the OTB and VOT datasets. The Siamese network has been widely used for face recognition, key point descriptions and character matching.

The tracking problem is in essence a similarity learning problem with respect to the object in initial frame. SiamFC compares the similarity between template image in initial frame and candidate image x in current frame by learning the matching equation $f(z, x) = g(\varphi(z), \varphi(x))$, where $\varphi()$ is a feature extraction network. The target position is determined by the highest scores. To make SiamFC to run fast, the feature network of SiamFC is based on AlexNet [34] with the fully connected layer being removed. The template image z and candidate image x pass through the feature network to get the corresponding feature maps of size $6 \times 6 \times 128$ and 22×128 respectively. The two sets of feature maps are then convolved to produce a response map which indicates the similarity between the template and the candidate image.

B. OPTICAL FLOW FOR OBJECT TRACKING

Recently, optical flow of moving objects has been widely used in computer vision field, for example, FlowNet [35] and TVNet [36]. With the help of pyramid network, FlowNet is the first network that utilize deep CNNS to predict the motion information. of moving objects. In frame prediction, in order to generate the middle or next frame, [37] proposed spatio-temporal video autoencoder which takes a video frame as input and estimate optical flow based on LSTM memory state and the current observation. In the field of pose estimation, using optical flow across the several frames as temporal context, [38] predicted the movement and gesture successfully in different wild datasets. Applying flow information in FlowNet, DFF [39] and FGFA [40] eliminated the effect of obscure objects due to fast motion and shape deformation in the task of video detection. Last but not least, network-based flow information has been used in [41] and achieved convincing results. However, these sorely utilize the flow feature off-the-shelf and are not trained end-to-end.

C. ATTENTION MODEL

Ignoring irrelevant visual information by intelligently exploring the visual field, attention mechanism is defined as the active direction of the mind to an object and successfully applied in the machine translation and other natural language processing tasks. For example, Wang eta. utilized a residual attention network that consists of bottom-up top-down feedforward structure to perform the image classification task [42]. With a sptial attention deep net combined with partial PSO (Particle Swarm Optimization), [43] Spatial Attention Deep Net with Partial PSO for Hierarchical Hybrid Hand Pose Estimation achieved better accuracy in human pose prediction. A video captioning model named Gaze Encoding Attention Network (GEAN) was proposed to leverage gaze tracking information to provide the spatial and temporal attention for sentence generation [44]. RASNet [45] put forward a general attention that learns from offline datasets and residual attention model based on hourglass network [46].

III. MAIN WORK

A. TVNET

Based on FlowNet [35] and FlowNet2.0 [47], TVNet was an end-to-end trainable neural network, able to learn optical-flow-like features from data. TVNet subsumed the optical flow solver and imitated and unfolded the iterative process of TV-L1 method [48] so that it can be used directly without any extra learning. The basic optical flow equation of the TV-L1 is written as:

$$\min_{u(x),x\in\Omega} \sum_{x\in\Omega} (|\nabla u_1(x)| + |\nabla u_2(x)|) + \lambda|\rho(u(x))| \quad (1)$$

where the $|\nabla u_1(x)| + |\nabla u_2(x)|$ is the smooth condition and $\rho(u(x))$ means the assumption of brightness constancy. To solve this equation, it is transformed to convex optimization problem by introducing an auxiliary variable v. The new optimization problem becomes

$$\min_{\{u,v\}} \sum_{x\in\Omega} (|\nabla u_1| + |\nabla u_2|) + \frac{1}{2\theta}|u - v|^2 + \lambda|\rho(v)| \quad (2)$$

In order to obtain the minimum value of the Eqn. 2, the value of u and v should be nearly equal. So we can optimize the Eqn. 2 by fixing the value of u and v alternatively. When fixing the value of v, the Eqn. 2 becomes:

$$\min_{\{u,v\}} \sum_{x\in\Omega} (|\nabla u_1| + |\nabla u_2|) + \frac{1}{2\theta}|u - v|^2 \quad (3)$$

When fixing the value of u, the Eqn. 2 is:

$$\min_{\{u,v\}} \sum_{x\in\Omega} \frac{1}{2\theta}|u - v|^2 + \lambda|\rho(v)| \quad (4)$$

For more details about solving the above optimization problems, please refer to [48].

The central idea of the TVNet algorithm is to transform the iterative process into a superposition of neural networks based on the TV-L1 solution. On the one hand, if the number of iterations is fixed in one loop, the iterative process in the TV-L1 algorithm can be expanded into a fixed-size feed-forward network. On the other hand, each iterative process is continuous, which ensures that gradients can be back-propagated through the layers, and thus the system is end-to-end trainable.

B. AGGREGATION USING OPTICAL FLOW

In our methods, the past frames are fused with the current frame to locate the moving object. For the frame fusion model, the most critical operation is to warp the candidate video frame into the target one. For an optical flow network, the warp operation refers to merging optical flow information obtained by the features of adjacent history frames through the optical flow network into the current frame. We follow the work of [41] to define the warp operation:

$$f_{j\to i} = W\left(f_j, M_{i\to j}\right) = W\left(f_j, F\left(I_i, I_j\right)\right) \quad (5)$$

where I_i is a video frame i, f_j is the feature map of frame j, and $F(I_i, I_j)$ is an optical flow network, such as TV-L1, which projects a location p in frame i to the location $p+\delta p$ in current frame i. So, $f_{j\to i}$ denotes the feature maps warped from previous frame j to current frame i. The warping operation is implemented by the bilinear functions applied on all the locations for each channel in the feature maps. The warping in certain channel is formulated as:

$$f^m_{j\to i}(p) = \sum_q K(q, p+\delta p) f^m_j(q) \tag{6}$$

where the $p = (p_x, p_y)$ is two-dimensional locations, $\delta p = F(I_i, I_j)(p)$ represents optical flow estimation value for each coordinate point, and m indicates a channel in the feature map, $q = (q_x, q_y)$ enumerates all locations in the feature map. From Eqn. 6, we can have

$$\frac{\partial f^m_{j\to i}(p)}{\partial f^m_j(q)} = K(q, p+\delta p) \tag{7}$$

From the Eqn. 7, we know that the backward propagation method can be applied in the certain network with the optical flow feature, which means the model can be trained end-to-end. The warp operation integrates the information of the previous frame into the current frame in the form of optical flow information, providing various information about the target object, such as different angles, illumination intensity and degree of deformation.

C. SEQUENTIAL SCORING MODEL

In order to effectively fuse the optical flow information of past frames with the current feature map, the weights for history frames are needed to indicate the importance of aggregated frames at each location. For this purpose, we put forward a sequential scoring model to obtain the weights. We adopt the idea of squeeze-and-excitation network [49] to design the sequential scoring model, which learns the weights according to the loss function through the network. The sequential scoring model is divided into two network blocks for extracting and expanding respectively, as shown in Fig. 2. The extracting network consists of the global average (Eqn. 8) and the global maximum pooling operation (Eqn. 9).

$$G_{S-GA}(q_T) = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} q_T(q_x, q_y) \tag{8}$$

where the W and H means the width and the height of the feature map and the q_T indicates the candidate history frames.

$$G_{S-GM}(q_T) = \mathrm{Max}\left(q_T(q_x, q_y)\right) \tag{9}$$

The above two pooling processes output two T-dimensions vectors respectively so that we can obtain two sets of weights through the shared expanded network block. The expanding network block is defined as follow:

$$W_{ex}(G_{S-GA}, G_{S-GM}) = \sigma(C_2\delta(C_1(G_{S-GA}, G_{S-GM}))) \tag{10}$$

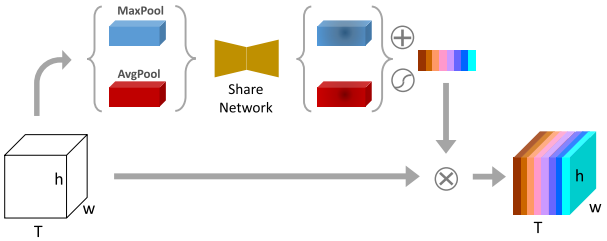

FIGURE 2. System diagram of sequential scoring model.

where the C_1 is the fully-connected network with output dimension is T/r, where r is a scaling parameter that can reduces or increases the number of candidate frames according to a certain ratio, hence decreasing the computational cost, δ is the ReLU activation layer, C_2 is another fully-connected network with the output dimension the same as the input dimension, and finally σ is the sigmoid function. The final output W_{ex} has the dimension of $T \times h \times w$, which contains the weight for each position in the input candidate frames. As we can see W_{ex} is learned from the fully-connected and activation layers, which can be trained end-to-end. The input to the sequential score model shown in Fig. 2 is the candidate frames.

D. AGGREGATION BETWEEN DIFFERENT FRAMES

With the sequential score model, we can obtain the weight matrix W_{ex}, the warped optical flow frames are aggregated by the weighted summation as

$$\bar{f}_i = \sum_{i=t-T}^{t} w_{j\to i} f_{j\to i} \tag{11}$$

where the $w_{j\to i}$ is the weight of each spatial locations and features of the optical flow frame, and $f_{j\to i}$ is the warped optical flow frame. The final $\bar{f}_i$ is the detection frame used for objection detection, which fuses the information of last T frames. Fig. 1 shows an example when $T = 3$. The instance frame and the last three frames are passed through the TVNet to calculate the optical flow, i.e. Flow1, Flow2 and Flow3 respectively. The optical flow frames are cropped to size of 22×22 to match the size of the feature map from the current frame. Then, the warp operation is performed on current feature map and each of the optical flow frame. After this, sequential scoring model takes in all the information to get the weight matrix W_{ex} for each of the three frames. In the end, the final detection frame is obtained by the weighted sum of the warped optical flow frames as defined in Eqn. 11. The overall algorithm SiamFlow is shown in **Algorithm** 1.

E. OPTICAL FLOW ATTENTION MODEL

When performing the convolution operation between the template frame and detection frame, the contribution of different areas in the template frame is the same, which limits the discriminating power of the algorithm. To solve this problem, an attention model is often used to pay more "attention" to

Algorithm 1 SiamFlow

Basic knowledge:
crop(...): cropping images for network input
$W(\ldots)$: the optical flow result merges with the current frame
$\varepsilon(\ldots)$: the mapped feature map
G_{S-GA}: global average pooling; G_{S-GM} : Global maximum pooling
$C(\ldots)$: the convolution of a template frame with the current frame
$P(\ldots)$: find the peak point in the response graph
Input: $I_1, \ldots, I_n, b_1$
Output: $b_{2,\ldots}, b_n$

$$D_1, T_1 \leftarrow \text{crop}\left(I_1^o, b_1\right)$$
$$d_1, t_1 \leftarrow N_{\text{feature}}\left(D_1, T_1\right)$$
for $i \leftarrow 2, \ldots, n$ **do**
  for $j = \max(1, i-k)$ to $(i+k)$ **do**
    $$f_{j\to i} = W\left(f_j, Flow\left(d_i, d_j\right)\right)$$
    $$f^e_{j\to i}, f^e_i = \varepsilon\left(f_{j\to i}, f_i\right)$$
    $$w_{j\to i} = G_{S-GM}\left(G_{S-GA}\left(f^e_{j\to i}, f^e_i\right)\right)$$
  end for
  $$\overline{d_i} = \sum_{i=t-T}^{t} w_{j\to i} f_{j\to i}$$
  $$r_i = C\left(t_i, \overline{d_i}\right)$$
  $$b_i = P\left(r_i\right)$$
end for

FIGURE 3. Effect of using attention model. Left figure: No attention model used.

the informative areas in the image. Fig. 3 shows the effect of using attention model. Using attention model, the interested target will have more weight and have more influence on the final output. In order to increase the attention of target position, we design the attention model by learning the appearance and motion information of target. In object tracking, the object location is obtained by the correlation operation $f(z, x) = \varphi(z)^*\varphi(\mathrm{X}) + b$. At pixel level, the correlation operation can be written as

$$f_{x',y'} = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\sum_{c=0}^{d-1}\varphi_{i,j,c}(z)\varphi_{x'+i,y'+j,c}(x) + b \qquad (12)$$

where x', y' are the location index in the response map, i, j, c are width, height, channel index in template image. In order to get different attention to each position of the template frame,

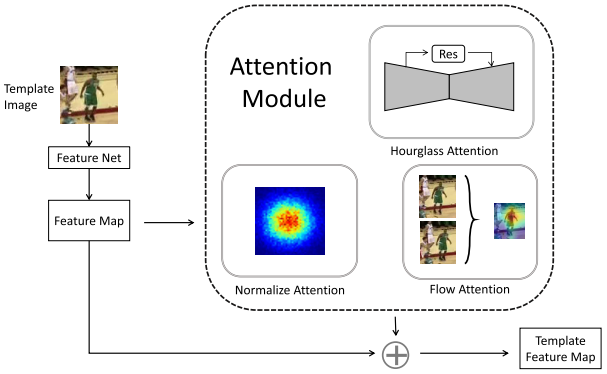

FIGURE 4. Attention models.

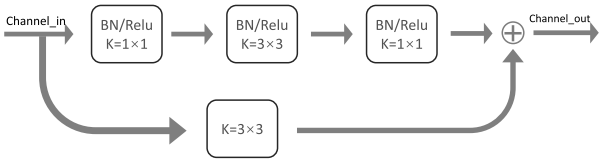

FIGURE 5. Residue block. BN is batch normalization, Relu is the Relu activation and K the filter size.

we add the weight γ to each template "pixel". Thus we have

$$f_{x',y'} = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\sum_{c=0}^{d-1}\gamma_{i,j,c}\varphi_{i,j,c}(z)\varphi_{x'+i,y'+j,c}(x) + b \qquad (13)$$

So, the correlation operation can be written as $f(z, x) = (\gamma \odot \varphi(z))^*\varphi(\mathrm{x}) + b$, where $\odot$ represents the element-wise multiplication and the weight γ is the weights given by the attention module. As shown in Fig. 4, our attention model is mainly composed of three parts, i.e. normalized attention module, Hourglass module [46], and optical flow module. Hourglass module consists of residual blocks, shown in Fig. 5, which are useful to extract deep level feature of the target. More details about the normalize attention model and hourglass attention model can be found in [45]. The overall attention model is

$$\omega = \lambda_1\omega_n + \lambda_2\omega_h + \lambda_3\omega_f \qquad (14)$$

where $\lambda_{1,2,3}$ are scalar weights, ω_n, ω_h and ω_f are the above three attention modules respectively. And the optical flow module ω_f is defined as

$$\omega_f = \sqrt{x^2 + y^2} \qquad (15)$$

The weight of each pixel location is the norm of the corresponding optical flow vector.

F. ONLINE TRACKING

1) SYSTEM DESCRIPTION AND TRAINING

The input images are passed through trained feature extraction network and optical flow network. Then the feature maps in previous frames are warped to the current one according to flow information. Warped feature maps as well as the

TABLE 1. Results of different template updating strategy on OTB-50.

	Fixed template	Fixed number of frames to replace the template			Update template dynamically	
		10	20	30	Use Euclidean distance	Use optical flow variable
Accuracy	0.642	0.591	0.603	0.624	0.626	0.598
Success rate	0.512	0.456	0.467	0.481	0.487	0.465

current frame's are input to the sequential scoring model to compute the weights for each frame. The detection frame is the weighted sum of the past frames by Eqn. 11. The estimate of the current target state is obtained by finding the maximum response in the score map of the detection frame.

After off-line training as described in Section III, the learned network is used to perform online tracking. During the training process, we train the feature net, TV-Net and sequential scoring model one by one. After training the adapted AlexNet in SiamFC framework on ILSVRC15 [33], we fix the feature network and train the TV-Net. And then fix the feature network and TV-Net, train the sequential scoring model that is used for combining different detected frames.

2) THE ANALYSIS OF THE TEMPLATE UPDATE

Our proposed SiamFlow and tracking algorithms based on the Siamese framework convolve the current frame with the template to get the response map, which is used to determine the object location by finding the maximum. Therefore, the accuracy of such methods largely depends on the template quality. Some algorithms always set the first frame as the template frame, such as CMT [50] and SiamFC. This strategy preserves the original appearance of the object during the tracking process, but it is hard to deal with cases of object deformation and rotation.

On the other hand, some algorithms update the template statically or dynamically. Statical way updates the template at fixed intervals, while dynamical way updates the template when it is needed. Updating template statically suffers from the problem when the tracker loses the object or the object is occluded or temporarily disappears. Dynamical way may overcome these problems smartly, for example, by calculating the Euclidean distance between the tracking result and the template [51].

To study the effect of template update strategy, we use OTB-50 dataset to conduct tests with different way of template updating. Table 1 shows the tracking accuracy and overlap ratio of SiamFlow. It can be seen fixing the initial frame as template all the time is the best way for SiamFlow. This is because the initial frame retains all the information of the target appearance. Updating template may introduce background interference, which will decrease the tracking performance. In the following tests, we fix the first frame as the template and do not update during tracking.

IV. EXPERIMENT

We have tested the proposed SiamFlow using OTB and VOT datasets.

A. RESULTS ON OTB

OTB50 or OTB2013 [27] contains 50 fully annotated sequences that are collected from commonly used tracking sequences. OTB100 or OTB2015 [52] is the extension of OTB2013 and contains 100 video sequences. Some new sequences are more difficult to track. The evaluation is based precision plot and success plot. The precision plot shows the percentage of frames with respect to the center distance in pixels between the tracking result and ground truth. The success plot shows the percentage of successful frames with respect to the overlap ratio between the tracking result and ground truth. The area under curve (AUC) of each success plot is used to rank the tracking algorithm.

Fig. 6 shows the comparison results on OTB-50, OTB-100 and OTB-CVPR13 datasets. From the figure, we can see that SaimFlow is ranked at top 2 on OTB-50 and OTB-CVPR13, and top 3 on OTB-100. Specifically, when compared to SaimFC, SiamFlow has far exceeded the SiamFC in accuracy and success rate, which means adding a trained optical flow network and sequential scoring model can improve the robustness of the Siamese framework. SiamFlow is also significantly better than DeepSRDCF and other deep network methods. If under the same network complexity and amount of training data, the improvement will be even larger. CCOT, a correlation filter based method, ranks at top 3 on OTB-50 and OTB-CVPR13 datasets and top 2 on OTB-100, which is very close to that of SiamFlow. Overall, SiamFlow is better than CCOT in the three. In the three tests, MDNet [53] achieves the best performance among the methods being compared, especially on OTB-50. However, MDNet establishes a multi-domain network based on specific video sequence, online fine tunes part of its parameters and uses a trained classifier to perform the tracking. This makes the method prone to over-fitting and difficult to migrate to other datasets without retraining. To verify this, we further conduct experiments on VOT-2016 and VOT-2017 datasets.

B. RESULTS ON VOT

The Visual Object Tracking (VOT) challenges are well known competitions in tracking community, which have held several times from 2013. In this subsection, we compare SiamFlow with entries in VOT2016 and VOT2017 [54].

1) RESULTS ON VOT 2016

Fig. 7 compares the overlap ratio of different algorithms when testing VOT-2016 videos of different lengths. The X-axis of Fig. 7 is the video length. The methods are ranked by the evaluation score using video sequences shorter than 200 frames, which is the dash line in the figure. We can see

TABLE 2. Tracking results on VOT-2016 dataset.

Algorithm	Overlap rate	Failure rate	Overall assessment	Normalized frame rate
$SiamFlow$	0.5571	**13.6143**	**0.3644**	33.6507
$CCOT$	0.5332	16.5817	0.331	**50.9975**
$TCNN$	0.547	17.9393	0.3249	0.9971
$SSAT$	**0.5703**	19.272	0.3207	0.4629
$MLDF$	0.4873	15.0437	0.3106	1.4007
$Staple$	0.5433	23.895	0.2952	10.9754
DDC	0.5337	20.9812	0.2929	0.1928
EBT	0.4529	15.1935	0.2913	2.9669
$SRBT$	0.4839	21.325	0.2904	2.2189
$SSKCF$	0.5423	22.712	0.2771	29.153
$MDNet_N$	0.5366	21.0817	0.2575	0.5130

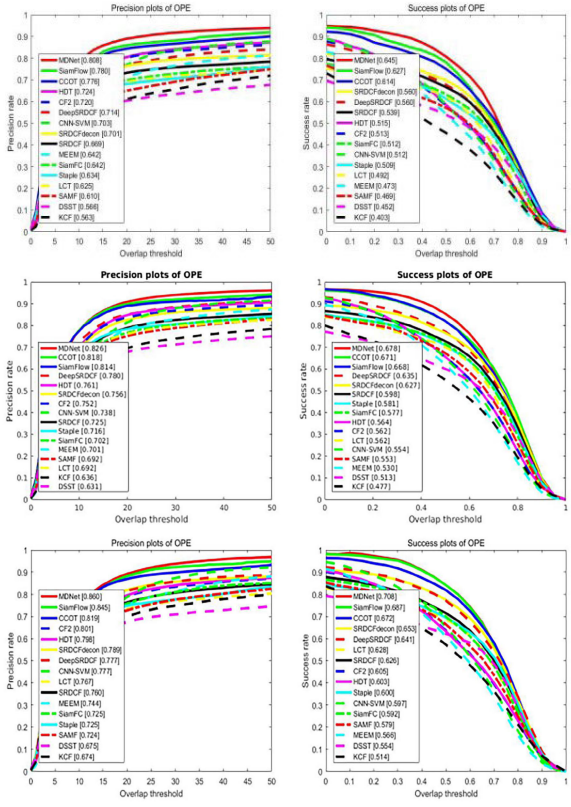

FIGURE 6. Performance comparison on OTB-50 (upper), OTB-100 (middle) and OTB-CVPR13 (lower) datasets.

that the expected overlap rate of the SiamFlow algorithm is significantly higher than other compared methods.

Table 2 compares the performance of these methods on VOT-2016. From the table, the overlap rate of the SiamFlow algorithm is 0.5571, second to SSAT (0.57), the failure rate is lowest, significantly less than that of others. The overall metric is 0.3644, which is the best among these methods. The FPS of SiamFlow is 33.65, less than that of CCOT, but SiamFlow is sufficient to work in real-time tracking scenarios.

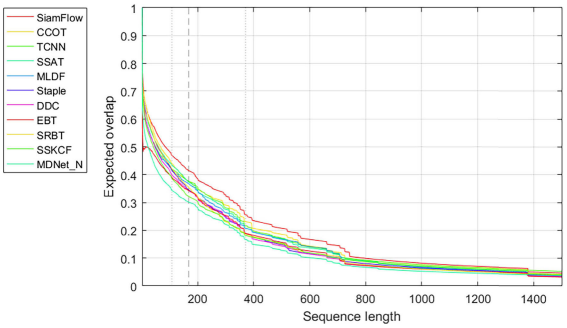

FIGURE 7. Overlap rate comparison on VOT2016 dataset.

2) RESULTS ON VOT 2017

Compared with VOT-2016, VOT-2017 adds some video sequences with complex backgrounds and tiny objects. Except for the baseline experiment, VOT-2017 adds real-time experiments, in which if the tracker does not return the tracking result within 40ms for each frame, the toolkit will not wait for the tracker.

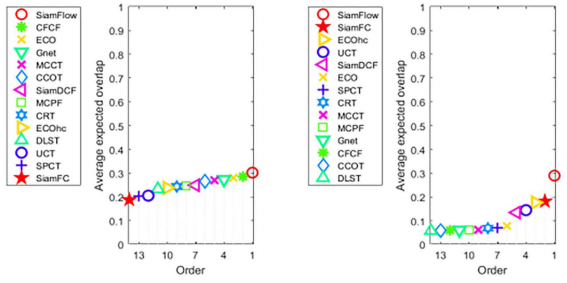

FIGURE 8. Baseline results (left) and real-time results (right) on VOT-2017 dataset.

Fig. 8 shows the results of baseline and real-time tests on VOT-2017 dataset. It can be seen from the two figures that SiamFlow has the best average expected overlap ratio. Especially, in the real-time experiment, SiamFlow is

TABLE 3. FPS (frame rate per second) comparison on VOT-2017 videos.

Algorithm	*Videos*							
	*ants*1	*ants*	*bag*	*ball*1	*ball*2	*basketball*	*birds*1	*blanket*
SiamFlow	**28.6987**	**28.2493**	34.0611	**30.8467**	**31.6175**	39.9129	20.3744	26.5435
SiamFC	25.6597	25.1416	**40.1062**	24.2537	17.0278	**45.5797**	**24.174**	**44.9994**
ECO	4.796	5.5821	3.696	3.2384	0.9144	5.0201	3.2532	4.7656
CFCF	1.2099	1.215	0.6715	0.9691	0.7048	0.7789	1.0577	1.1498
Gnet	1.4117	1.4192	1.4272	1.3563	1.2142	1.4505	1.3508	1.3871
CCOT	0.1357	0.1434	0.1116	0.1429	0.072	0.1064	0.171	0.1731
SPCT	4.7021	5.7617	0.949	4.557	10.5645	2.1675	6.3062	6.0095
SiamDCF	5.7258	6.0801	16.3134	6.543	2.1117	22.5168	7.4637	20.3726
ECOhc	17.3349	21.4655	20.447	17.5164	7.2909	26.1938	20.4282	32.0339
MCPF	0.3209	0.2256	0.163	0.4145	0.5898	0.4162	0.6305	0.7748
DLST	2.4548	2.6646	2.2912	2.4355	1.5573	2.5105	2.4096	2.6754
CRT	3.9336	3.8282	3.0091	3.455	2.5081	4.1417	3.5932	4.5117
UCT	10.7427	11.5286	16.9184	10.2795	4.0762	19.4009	9.8967	18.4538
MCCT	1.2995	1.2778	1.3314	1.3417	1.3297	1.3069	1.3083	1.2982

TABLE 4. Results comparison on VOT-2017 dataset.

Algorithm	Overlap rate	Failure rate	Overall assessment	Normalized frame rate
SiamFlow	**0.5347**	18.8776	**0.3021**	29.3197
ECO	0.4758	17.6628	0.2805	3.7056
CFCF	0.5049	19.6495	0.2857	0.8543
Gnet	0.4999	**17.3674**	0.2737	1.2912
CCOT	0.4848	20.4138	0.2671	0.1463
SPCT	0.4658	33.1965	0.2036	4.3956
SiamDCF	0.4957	29.4063	0.2494	10.7328
SiamFC	0.5002	34.0259	0.188	**31.889**
ECOhc	0.4893	28.7674	0.2384	17.7107
MCPF	0.5035	25.96	0.2478	0.4195
DLST	0.5038	24.6046	0.2332	1.8855
CRT	0.4639	21.0611	0.2441	3.2436
UCT	0.4839	29.7991	0.2058	12.1953
MCCT	0.5228	19.4526	0.2703	1.3189

significant better than other methods, including SiamFC and SiamDCF, which is also based on Siamese framework.

Table 3 shows the FPS on some of videos of VOT-2017. It shows that the FPS of SiamFC and SiamFlow is much higher than that of other deep learning methods such as ECO and CFCF. Because we add the optical flow network and sequence scoring model based on SiamFC, SiamFlow is slower than SiamFC. However, it is still much faster than other methods.

Table 4 shows the test results on VOT-2017. As shown in the table, SaimFlow has the best overlap rate, its failure rate is 18.8776, less than the best one CCOT (17.3674). But in terms of the overall metric, SaimFlow ranks at top 1 with the score of 0.3021, which is significant better than other methods. And the FPS is 29.3197, second to the best one SiamFC (31.889).

C. ABLATION ANALYSIS

In this section, we conduct two experiments to study how the algorithm configuration affects the tracking performance. The first test with OTB-50 dataset is to analyze the effect of the optical flow network and sequential scoring model. Table 5 shows the corresponding results. From the table, we can conclude that trained optical flow and sequential scoring model improve the tracking accuracy and success rate, and using fixed optical flow plus sequential scoring model or trained optical flow alone gives much worse results.

The second test is to investigate the impact of attention model on the Siamese tracking network. The experiment is done on datasets of OTB-50, OTB-100 and OTB-CVPR13. In Fig. 4 and Eqn. 14, we use the weighted sum of three attention modules. In Table 6, we compare the results of different combination of these attention models. We can see that in the first row, the performance is the worst, which shows using attention model enhances the ability of recognizing the object on the template. Secondly, when it comes to different individual attention model, hourglass attention model is the best in the table, due to its complex network structure and deep optimization of the residual blocks. Thirdly, comparing

TABLE 5. Results of different algorithm module configurations on OTB-50.

Accuracy	Fixed optical flow + sequenntial scoring model	Trained optical flow + no sequential scoring model	SiamFlow: (Trained optical flow + sequential scoring model)
FlowNet	0.755	0.724	0.772
TV-Net	0.759	0.736	0.780

Success rate	Fixed optical flow + sequenntial scoring model	Trained optical flow + no sequential scoring model	SiamFlow: (Trained optical flow + sequential scoring model)
FlowNet	0.593	0.587	0.615
TV-Net	0.590	0.579	0.627

TABLE 6. Result of using different combination of attention models. HA: Hourglass attention model, NA: Normalize attention model, FA: Optical flow attention model.

	OTB-50		OTB-100		OTB-CVPR13	
	Accuracy	Success Rate	*Accuracy*	Success Rate	*Accuracy*	Success Rate
No attention model	0.642	0.512	0.702	0.584	0.725	0.592
NA	0.658	0.523	0.734	0.589	0.756	0.601
HA	0.705	0.547	0.792	0.618	0.803	0.623
FA	0.661	0.525	0.747	0.591	0.760	0.598
NA + HA	0.734	0.570	0.813	0.622	0.824	0.633
FA + HA	**0.791**	**0.620**	**0.856**	**0.646**	**0.881**	**0.657**
NA + FA	0.785	0.614	0.851	0.639	0.873	0.637
NA + HA + FA	0.740	0.569	0.751	0.625	0.772	0.635

the performance of single models and that of combined models, combined models shows advantages over single models. Specifically, optical flow attention model plus hourglass attention model achieves the best performance on all three tests.

V. CONCLUSION

This paper proposes SiamFlow based on SiamFC to improve the tracking performance in two ways. First, the optical flow information of past video frames by the optical flow network are warped into the feature map of the current frame. The warped feature map are then combined using the weights given by the sequential scoring model to get the detection frame, in which the object is to be located. Second, we apply the attention modules on the template to further improve the tracking effect. Specifically, the attention model is the weighted sum of normalized, hourglass and optical flow attention modules. Experiments on OTB and VOT datasets have shown that combining optical flow information of history frames and attention models improves the tracking performance. As part of future work, to further improve the performance and robustness of object tracking, we are exploring new methods such as geometrical algebra as it can be used in higher dimensional image representation [55], [56].

REFERENCES

[1] K. Q. Huang, X. T. Chen, Y. F. Kang, and T. N. Tan, "Intelligent visual surveillance: A review," *Chin. J. Comput.*, vol. 38, no. 6, pp. 1093–1118, 2015.

[2] L. Sevilla-Lara and E. Learned-Miller, "Distribution fields for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1910–1917.

[3] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1619–1632, Aug. 2011.

[4] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.

[5] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Jun. 2000, pp. 142–149.

[6] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, pp. 35–45, Mar. 1960.

[7] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 425–437, Feb. 2002.

[8] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.

[9] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[11] W. Cao, Q. Lin, Z. He, and Z. He, "Hybrid representation learning for cross-modal retrieval," *Neurocomputing*, vol. 345, pp. 45–47, Jun. 2019.

[12] W. Cao, J. Yuan, Z. He, Z. Zhang, and Z. He, "Fast deep neural networks with knowledge guided training and predicted regions of interests for real-time video object detection," *IEEE Access*, vol. 6, pp. 8990–8999, 2018.

[13] M. Dan, L. Zhang, G. Cao, W. Cao, G. Zhang, and H. Bing, "Liver fibrosis classification based on transfer learning and FCNet for ultrasound images," *IEEE Access*, vol. 5, pp. 5804–5810, 2017.

[14] D. Meng, G. Cao, Y. Duan, M. Zhu, L. Tu, D. Xu, and J. Xu, "Tongue images classification based on constrained high dispersal network," *Evidence-Based Complementray Alternative Med.*, vol. 2017, Mar. 2017, Art. no. 7452427.
[15] W. Cao, J. Zhong, G. Cao, and Z. He, "Physiological function assessment based on Kinect V2," *IEEE Access*, vol. 7, pp. 105638–105651, 2019.
[16] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Dec. 2015, pp. 58–66.
[17] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Computer Vision—ECCV*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, pp. 472–488.
[18] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6638–6646.
[19] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2544–2550.
[20] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Computer Vision—ECCV*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Germany: Springer, 2012, pp. 702–715.
[21] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.
[22] M. Danelljan, F. S. Khan, M. Felsberg, and J. Van de Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1090–1097.
[23] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Discriminative scale space tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1561–1575, Sep. 2016.
[24] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 4310–4318.
[25] H. Kiani Galoogahi, T. Sim, and S. Lucey, "Correlation filters with limited boundaries," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4630–4638.
[26] N. Wang and D.-Y. Yeung, "Learning a deep compact image representation for visual tracking," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 809–817.
[27] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2411–2418.
[28] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional siamese networks for object tracking," in *Computer Vision—ECCV 2016 Workshops*, G. Hua and H. Jégou, Eds. Cham, Switzerland: Springer, 2016, pp. 850–865.
[29] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a 'siamese' time delay neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 737–744.
[30] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2805–2813.
[31] Z. Zhengyu, L. Bing, R. Yunbo, and L. Qiao, "STResNet_CF tracker: The deep spatiotemporal features learning for correlation filter based robust visual object tracking," *IEEE Access*, vol. 7, pp. 30142–30156, 2019.
[32] M. Kristan, J. Matas, A. Leonardis, T. Vojir, R. Pflugfelder, G. Fernandez, G. Nebehay, F. Porikli, and L. Čehovin, "A novel performance evaluation methodology for single-target trackers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 11, pp. 2137–2155, Nov. 2016.
[33] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[35] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, "FlowNet: Learning optical flow with convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2758–2766.
[36] L. Fan, W. Huang, C. Gan, S. Ermon, B. Gong, and J. Huang, "End-to-end learning of motion representation for video understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6016–6025.
[37] V. Patraucean, A. Handa, and R. Cipolla, "Spatio-temporal video autoencoder with differentiable memory," 2015, *arXiv:1511.06309*. [Online]. Available: https://arxiv.org/abs/1511.06309
[38] T. Pfister, J. Charles, and A. Zisserman, "Flowing convnets for human pose estimation in videos," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1913–1921.
[39] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei, "Deep feature flow for video recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2349–2358.
[40] X. Zhu, Y. Wang, J. Dai, L. Yuan, and Y. Wei, "Flow-guided feature aggregation for video object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 408–417.
[41] Z. Zhu, W. Wu, W. Zou, and J. Yan, "End-to-end flow correlation tracking with spatial-temporal attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 548–557.
[42] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 3156–3164.
[43] Q. Ye, S. Yuan, and T.-K. Kim, "Spatial attention deep net with partial pso for hierarchical hybrid hand pose estimation," in *Computer Vision—ECCV*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, pp. 346–361.
[44] Y. Yu, J. Choi, Y. Kim, K. Yoo, S.-H. Lee, and G. Kim, "Supervising neural attention models for video captioning by human gaze data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 490–498.
[45] Q. Wang, Z. Teng, J. Xing, J. Gao, W. Hu, and S. Maybank, "Learning attentions: Residual attentional siamese network for high performance online visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4854–4863.
[46] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 483–499.
[47] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "FlowNet 2.0: Evolution of optical flow estimation with deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2462–2470.
[48] J. S. Pérez, E. Meinhardt-Llopis, and G. Facciolo, "Tv-L_1 optical flow estimation," *Image Process. On Line*, vol. 2013, pp. 137–150, Jul. 2013.
[49] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[50] G. Nebehay and R. Pflugfelder, "Clustering of static-adaptive correspondences for deformable object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2784–2791.
[51] P.-E. Danielsson, "Euclidean distance mapping," *Comput. Graph. Image Process.*, vol. 14, no. 3, pp. 227–248, 1980.
[52] Y. Wu, J. Lim, and M. H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1834–1848, Sep. 2015.
[53] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4293–4302.
[54] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. C. Zajc, T. Vojir, G. Hager, A. Lukezic, A. Eldesokey, and G. Fernandez, "The visual object tracking vot2017 challenge results," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, Oct. 2017, pp. 1949–1972.
[55] R. Wang, W. Zhang, Y. Shi, X. Wang, and C. Wenming, "GA-ORB: A new efficient feature extraction algorithm for multispectral images based on geometric algebra," *IEEE Access*, vol. 7, pp. 71235–71244, 2019.
[56] R. Wang, M. Shen, T. Wang, and W. Cao, "L_1-norm minimization for multi-dimensional signals based on geometric algebra," *Adv. Appl. Clifford Algebras*, vol. 29, p. 33, Apr. 2019.