Received April 30, 2020, accepted May 17, 2020, date of publication May 27, 2020, date of current version June 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2997917

Efficient Visual Tracking With Stacked Channel-Spatial Attention Learning

MD. MAKLACHUR RAHMAN, MUSTANSAR FIAZ, AND SOON KI JUNG, (Senior Member, IEEE)
School of Computer Science and Engineering, Kyungpook National University, Daegu 41566, South Korea

Corresponding author: Soon Ki Jung (skjung@knu.ac.kr)

This work was supported by the Development Project of Leading Technology for Future Vehicle of the Business of Daegu Metropolitan City under Grant 20171105.

ABSTRACT Template based learning, particularly Siamese networks, has recently become popular due to balancing accuracy and speed. However, preserving tracker robustness against challenging scenarios with real-time speed is a primary concern for visual object tracking. Siamese trackers confront difficulties handling target appearance changes continually due to less discrimination ability learning between target and background information. This paper presents stacked channel-spatial attention within Siamese networks to improve tracker robustness without sacrificing fast-tracking speed. The proposed channel attention strengthens target-specific channels increasing their weight while reducing the importance of irrelevant channels with lower weights. Spatial attention is focusing on the most informative region of the target feature map. We integrate the proposed channel and spatial attention modules to enhance tracking performance with end-to-end learning. The proposed tracking framework learns what and where to highlight important target information for efficient tracking. Experimental results on widely used OTB100, OTB50, VOT2016, VOT2017/18, TC-128, and UAV123 benchmarks verified the proposed tracker achieved outstanding performance compared with state-of-the-art trackers.

INDEX TERMS Deep learning, Siamese architecture, stacked channel-spatial attention, visual object tracking.

I. INTRODUCTION

Visual object tracking is a fundamental and challenging task for a wide range of computer vision applications, including intelligent surveillance [1], autonomous vehicles [2], game analysis [3], and human-computer interface [4]. An object bounding box is usually provided in the first frame of a video, and the tracking algorithm predicts new object locations in succeeding frames. Although many frameworks have been proposed, it remains an arduous task to develop a generic object tracker to handle various tracking challenges such as scale variation, illumination variation, fast motion, motion blur, occlusion, deformation, and background clutter.

Generative and discriminative strategies are commonly employed to solve the visual tracking problem. Generative strategies construct an analogous appearance representation for the target to find candidate positions in successive frames using neighborhood location searches around the existing target [8]. Discriminant strategies consider classification or regression frameworks to discriminate foreground from background for solving the tracking problem [9].

However, predicting target locations using discriminative methods classically requires large datasets for training or updating online to ensure acceptable classifier performance. This situation has altered somewhat with the introduction of the minimum output sum of squared error (MOSSE) [10] filter, which allows adaptive training schemes to perform robust and efficient object tracking. The MOSSE filter uses a Fourier transform to minimize the sum of the squared error between actual and desired output. Several previous studies have proposed approaches based around the MOOSE filter, e.g. CSK [11] used kernel methods to improve the underlying MOSSE filter, and CN tracker [12] employs color attributes to improve input data representation. However, handling challenges using hand-crafted features, such as histogram of oriented gradients (HOG) and color histograms with discriminative correlation filters (DCFs) significantly reduce performance due to

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

circular boundary effects, hence they are unsuitable for active tracking.

Recently, due to the powerful feature representation ability of the Convolutional neural networks (CNNs), have gained considerable research attention recently in many computer vision fields, such as semantic segmentation [13], object detection [14], and activity recognition [15]. Deep features are exploited within DCF based trackers to address these challenges, including HDT [16], deepSRDCF [17], and ECO [18]; or deep tracking frameworks MDNet [19], CNNSI [20], and FCNT [21]. However, using the pre-trained model as the tracker backbone for feature extraction is unsuitable due to inconsistencies between tracking and other visual tasks. Although CNNs provide better tracking, their data-hungry characteristics require considerable effort to collect sufficient data to train the end-to-end network. Ultimate tracking in real-time with good accuracy also needs to be considered when designing the tracker.

Several trackers have been proposed to overcome these difficulties. For example [5], [22], [23] consider tracking as a matching problem to learn similarity measures in end-to-end learning. The main advantage of similarity learning is that it employs offline end-to-end training to balance between speed and accuracy. Superior Siamese network matching can match the most analogous patch based on the target template, hence Siamese networks have shown great success in tracking. Recently, SiamFC [5] has gained enormous popularity for tracking due to its balanced performance using a simple Siamese architecture. Other siamese based trackers, for example, DsimaM [7] learns background suppression and appearance variations from earlier frames using a fast transformation learning model; whereas DCFNet [6] integrates a discriminant correlation filter (DCF) within a lightweight architecture and drives back-propagation to adjust the DCF layer using the probability heat map of the target location. However, these approaches lack robustness and are weak for handling challenging scenarios, particularly when for object appearance changes, as shown in Fig. 1.

To handle the aforementioned tracking challenges, we propose an extended underlying Siamese architecture incorporating stacked channel-spatial attention (SCSAtt) in the template branch with end-to-end trainable architecture. The SCSAtt channel attention module enhances target adaptability by utilizing different weights for channels depending on their contribution. After computing channel attention, we employ a spatial attention module to emphasize the most informative region on the feature map and hence identify the target location. The overall attention mechanism helps to improve feature representation power and discriminative ability, ensuring high tracking performance. We employ offline training to learn the similarity map, providing computational efficiency during tracking. The overall attention mechanism is extremely flexible to integrate with the Siamese architecture. We validated the effectiveness of our proposed framework using several challenging benchmarks [24]–[30] and compared performance results with other state-of-the-art trackers.

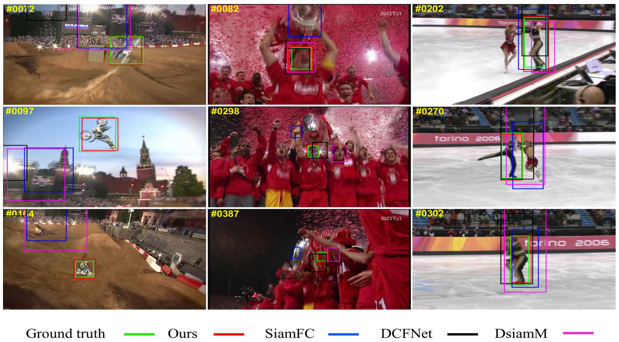

FIGURE 1. Compared our proposed tracker with siamese based trackers including SiamFC [5], DCFNet [6], and DsimaM [7] for MotorRolling (left column), Soccer (middle column), and skating2-2 (right column).

The main contributions from this study are as follows.

- We present stacked channel-spatial attention within a Siamese framework to learn effective feature representation and discrimination ability for high tracking performance.
- Rather than a single attention module, we combine multiple attention modules with residual skip connection in a specific order to enhance feature fusion training and target adaptability.
- We evaluated optimal attention module placement within fully convolutional single or multiple layers to enhance end-to-end training benefits for efficient tracking.
- We conducted extensive experiments using OTB100, OTB50, VOT2016, VOT2017/18, TC-128, and UAV123 benchmarks to validate the proposed approach, achieving 61 frames per second (fps) real-time processing speed and high accuracy compared with state-of-the-art tracking methods. To facilitate further studies, models and results are available at https://github.com/maklachur/SCSAtt.

II. RELATED WORK

Many visual object tracking frameworks have proposed over the last decade. It is inconvenient to cover a comprehensive survey of all trackers in the scope of this work. However, these survey studies [31]–[33] help to learn a detailed overview of the tracking frameworks for interested readers. This section provides short outlines for deep feature based trackers [18], [21], [34]–[36], Siamese based trackers [5], [7], [37]–[39], and attention based trackers [22], [40]–[45].

A. DEEP FEATURE BASED TRACKERS

The superior ability of the deep neural networks boost tracker performance by extracting significant features from the images. These deep features are then utilized by correlation filter tracking frameworks to improve performance, including DeepSRDCF [35], CF2 [36], and HDT [16]. Features from continuous convolution filters are also used to build trackers, such as ECO [18] and C-COT [34]. FCNT [21] selects features using regression, obtaining good accuracy but cannot perform in real-time due to high dimension convolutional feature representation. DeepTrack [46] considers tracking as a classification problem and learns feature weights by online training using iterative stochastic gradient descent (SGD) approach.

Although these trackers have outstanding feature representation power, they are difficult to train offline on large benchmarks. Thus, these online approaches diminish the network richness, which affects overall tracking performance, and particularly tracker speed.

B. SIAMESE BASED TRACKERS

Siamese architecture formulates a similarity learning problem where two parallel convolutional layer streams share parameters and calculate similarity loss between two input images to train the network through back-propagation. This network was first developed for signature verification [47]. Siamese based trackers [5], [7], [37]–[39] solve tracking as a similarity learning problem between target and search images and have become popular recently within the tracking community due to their balanced performance in terms of accuracy and speed.

For example, GOTURN [37] formulates a relative motion estimation solution to encounter the regression problem. SiamFC [5] casts tracking as a template matching problem where the network learns similarity from embedded features. Although SiamFC is one of the most popular and pioneering approaches for visual object tracking, due to its steady speed and accuracy, it struggles with various challenges, including appearance changes, background clutter, and deformations. Therefore, many subsequent studies have improved SiamFC to enhance tracking performance. CFNet [39] integrates the correlation filter at the end of the template branch in a closed-form equation. SiamMCF [38] and DSiam [7] incorporate cross-correlation on multiple layers to solve the similarity problem.

We modify the underlying Siamese architecture to include input image sizes and embedded more feature channels providing an appropriate complement for incorporating attention mechanisms.

C. ATTENTION BASED TRACKERS

Attention mechanisms within neural networks has become an important approach for computer vision applications, such as image classification [48], [49], object detection [50], and segmentation [51]. Attention, or focusing on important image features, is an effective mechanism to help solve object tracking problems, and has attracted strong research attention within the tracking community, with several attention based trackers proposed [22], [40]–[45].

SA-Siam [22] integrates channel attention in the semantic branch to compute channel-wise weights around the object location. RASNet [41] combines three attention modules to enhance tracker discriminative competence and adaptability. FICFNet [42] computes channel attention on both Siamese pipeline branches to weight feature channels. IMG-Siam [43] fuses the target foreground using channel attention and the super pixel based matting algorithm to provide enhanced target appearance with structural information. FlowTrack [44] uses temporal attention to capture target temporal information. MemTrack [40] and MemDTC [45] uses a long short term memory (LSTM) attention based controller to govern the feature map read and write operation using memory.

In contrast, we propose an attention mechanism with the end-to-end training facility where channel attention emphasizes 'what' informative part of the target image has to focus and spatial attention is responsible for 'where' the informative part is located. Therefore, combining these two attention modules learn 'what' and 'where' to focus or suppress the target information by refining intermediate features efficiently during the flow into the network.

III. PROPOSED METHOD

This section describes the proposed tracker methodology. The proposed tracking framework incorporates the stacked channel-spatial attention mechanism in the Siamese architecture target branch to improve tracker discrimination ability that helps to locate the target object in the search region efficiently. We also alter the underlying fully convolutional SiamFC [5] with different input sized images and internal architecture suitable for integrating the proposed attention mechanism to enhance target feature representation power. Fig. 2 shows the proposed tracker pipeline.

A. SIAMESE NETWORK FOR FEATURE LEARNING

The basic SiamFC framework generally includes two fully convolutional symmetric branches for learning features through weight sharing. SiamFC performs cross-correlation at the end of the feature extraction network between target and search image features to compute the similarity score map, where the maximum similarity score is taken as the predicted object location on the search image. This architecture can be expressed mathematically as

$$f(z, x) = \varphi(z) * \varphi(x) + b \cdot \mathbb{1}, \tag{1}$$

where $\varphi(\cdot)$ represents the fully convolutional network, $b \cdot \mathbb{1}$ is the bias value for every $b \in \mathbb{R}$, and $*$ represents the cross-correlation to compute response map between the target and search image feature maps.

During the Siamese object tracking, the responsible target branch remains stationary after taking fridge weights from the offline trained model for the first frame of the video sequence named template (target image). The target object's location is estimated for subsequent frames by matching with the template at the highest similarity score on the response map. The generalization ability of the target branch helps to improve tracker quality because it is static. Since object location in Siamese based tracking is predicated based on similarity score, we concentrated on generating the most robust and discriminative features for similarity learning to

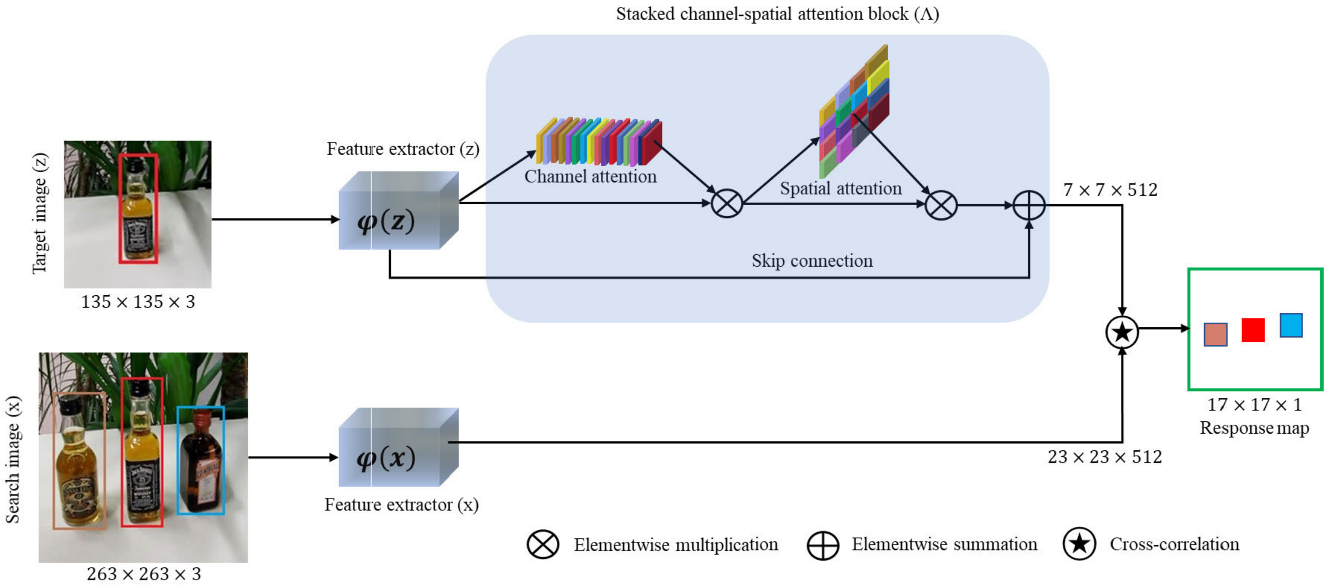

FIGURE 2. **The overall architecture of the proposed tracker. The shaded region represents the stacked channel-spatial attention block where channel and spatial attention modules are integrated after feature extractor for the target branch. The output of channel attention is forwarded as input to the spatial attention module. Finally, attention features are fused with skip connection for efficient discriminative features. A response map is constructed using cross-correlation between target and search image feature map. The red square in the response map resembles the highest similarity score that represents the target location in the search image.**

build efficient tracker. However, basic Siamese tracker frameworks are unable to handle challenging tracking cases due to their reduced discrimination ability. To improve tracker discrimination ability, we used asymmetric fully convolutional branches by integrating stacked channel-spatial attention in the target branch. In particular, we altered the underlying Siamese tracking architecture as follows to ensure high tracking performance,

$$f(z,x) = \Lambda(\varphi(z)) * \varphi(x) + b \cdot \mathbb{1}, \tag{2}$$

where $\Lambda(\cdot)$ denotes the stacked channel-spatial attention mechanism for the target feature map $\varphi(z)$ that learns to effectively highlight appearance and refine the location feature for the object.

B. STACKED CHANNEL-SPATIAL ATTENTION

We were inspired by human visual perception, which does not require concentrating on the whole scene, but rather focuses on the specific object for perceiving informative parts to understanding the appropriate visual pattern [52]. Similarly, attention mechanism prioritize important features to understand salient object parts [41]. Since single object tracking resembles focusing on the most salient feature, it is beneficial to concentrate on crucial regions of the target image.

Unlike other attention-based trackers, we integrated the attention mechanism only in the target branch to reduce the overall parameters overhead. It enables us to preserve fast-tracking speed and overall tracking process simple. Our attention mechanism is easily integrable to any convolutional layers of the network. However, during tracking, we required only a pre-trained model and the first frame of the video to track the sequence. On the other hand, the existing attention-based trackers including MemTrack [40] and MemDTC [45] maintain previous memory for the tracked object and update accordingly; IMG-Siam [43] uses super-pixel based mating to extract the target foreground; FlowTrack [44] utilizes the historical frames to model update; FICFNet [42] integrates attention module to both target and search branches.

Therefore, constructing an efficient object tracking, we propose stacked channel-spatial attention mechanism inside the Siamese framework named SCSAtt to enhance feature representation power for improving tracker discrimination ability. This attention approach linearly combines two popular attention modules, channel and spatial attention. The channel attention module measures the weight contribution of the channels, whereas spatial attention focuses on salient object regions in the feature maps.

The proposed attention mechanism first employs channel attention C_A on the output feature map F_M computed from the last convolution layer. The output from C_A is forwarded to the spatial attention module, yielding the spatial attention feature map S_A. To ensure our network efficient, we fuse the S_A with F_M using a residual skip connection.

We can summarize the process steps as

$$C_A = \phi_c(F_M) \otimes F_M, \tag{3}$$
$$S_A = \phi_s(C_A) \otimes C_A, \tag{4}$$

and

$$\Lambda(\varphi(z)) = S_A \oplus F_M, \tag{5}$$

where $\Lambda(\varphi(z))$ is the final stacked channel-spatial attention; $\phi_c(\cdot)$ and $\phi_s(\cdot)$ represent channel and spatial attention, respectively; and F_M is the fully convolutional feature map of z.

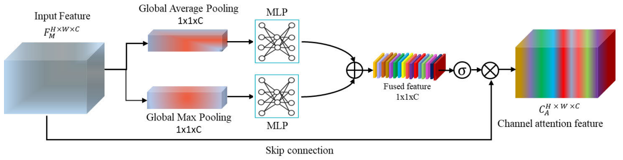

FIGURE 3. **Proposed channel attention module architecture.**

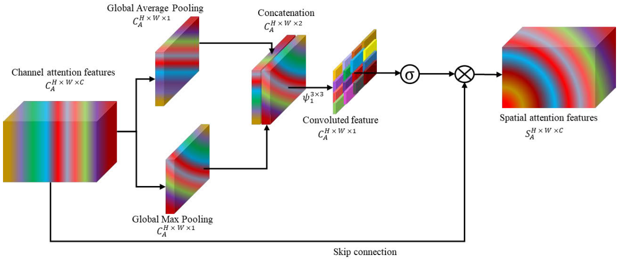
Global Average Pooling
$C_A^{H\times W\times 1}$
Concatenation
$C_A^{H\times W\times 2}$
Channel attention features
$C_A^{H\times W\times C}$
$\psi_1^{3\times3}$
Convoluted feature
$C_A^{H\times W\times 1}$
σ
Spatial attention features
$S_A^{H\times W\times C}$
Global Max Pooling
$C_A^{H\times W\times 1}$
Skip connection

FIGURE 4. **Proposed spatial attention module architecture.**

1) CHANNEL ATTENTION

Each feature channel represents a particular visual pattern. During training, convolution feature map contributions from each channel do not represent an object equally, with some channels representing an object's visual pattern better than others and vice-versa. Therefore, most previous attention models, e.g. [22], [41], [42], and [53], use either global average or max pooling with a multilayer perceptron (MLP) to calculate their gain. In contrast, rather than a single pooling operation, we consider the global average and max pooling together to construct a channel attention module that learns fused features. The global max-pooling operation focuses on distinctive and finer object features, whereas global average pooling provides overall knowledge on the feature map for channel attention.

After computing both pooling operations, we calculate individual MLPs using an rectified linear unit (ReLU) layer to learn the non-linearity between two fully-connected layers with 128 and 512 nodes, respectively. Hence, we obtain two feature vectors $F_{max}^{1\times1\times C}$ and $F_{avg}^{1\times1\times C}$ for max and average pooling, respectively. Before applying sigmoid activation for normalization, we fused both feature vectors using element-wise summation. Finally, we calculated the product with skip connection to propagate effects on the original feature map, providing the ultimate channel attention feature map $C_A^{H\times W\times C}$, as shown in Fig. 3.

The channel attention component can be expressed as

$$F_{max}^{1\times1\times C} = fc_2(ReLU(fc_1(GPool_{max}(F_M^{H\times W\times C})))), \quad (6)$$

$$F_{avg}^{1\times1\times C} = fc_2(ReLU(fc_1(GPool_{avg}(F_M^{H\times W\times C})))), \quad (7)$$

$$\phi_c(\cdot)^{1\times1\times C} = \sigma(F_{max}^{1\times1\times C} \oplus F_{avg}^{1\times1\times C}), \quad (8)$$

and

$$C_A^{H\times W\times C} = \phi_c(\cdot)^{1\times1\times C} \otimes F_M^{H\times W\times C}, \quad (9)$$

where σ represents the usual sigmoid function $f(x) = \frac{1}{1+e^{-x}}$.

2) SPATIAL ATTENTION

In contrast to channel attention, spatial attention highlights where informative features of the object in an image [48] for spotting the target location that provides a good complementary to channel attention. Previously, Qin and Fan [43] constructed a spatial mask using super-pixels to exploit target representation. Li and Yang [53] utilized global max pooling to encode the spatial attention in their model. We exploit the relationship among channels inter-spatial features to construct spatial attention. Pooling in the channel dimension highlights the informative area [54], which helps locate the desired target on the image by comparing overall weight gains. To formulate this attention, we compute global max pooling $S_{max}^{H\times W\times 1}$ and average pooling $S_{avg}^{H\times W\times 1}$ on the feature maps and fuse them in the channel domain. Since convolution operations consider as local operation and empirically, this approach focuses on target information.

We apply a convolution layer $\psi_1^{3\times3}$ after concatenating doubly pooled features, experimentally choosing a 3 × 3 convolutional filter for best results, and down-sample the number of feature channels to 1 to obtain the single channel feature map. After broadcasting this convoluted feature map through the sigmoid operation, we compute a product with the previously acquired channel attention feature map $C_A^{H\times W\times C}$ to obtain the ultimate effect on the spatial attention feature map $S_A^{H\times W\times C}$, as shown in Fig. 4. This attention feature map is calculated as

$$S_{max}^{H\times W\times 1} = GPool_{max}(C_A^{H\times W\times C}), \quad (10)$$

$$S_{avg}^{H\times W\times 1} = GPool_{avg}(C_A^{H\times W\times C}), \quad (11)$$

$$\phi_s(\cdot)^{H\times W\times 1} = \sigma(\psi_1^{3\times3}(concat[S_{max}^{H\times W\times 1}, S_{avg}^{H\times W\times 1}]), \quad (12)$$

and

$$S_A^{H\times W\times C} = \phi_s(\cdot)^{H\times W\times 1} \otimes C_A^{H\times W\times C}, \quad (13)$$

where $\psi_1^{3\times3}$ is the convolution operation with 3 × 3 kernel and stride and padding = 1.

C. IMPLEMENTATION DETAILS

We adopted an AlexNet-like [55] backbone for the proposed tracker framework to extract the feature map, with 135 × 135 × 3 and 263 × 263 × 3 target and search image sizes, respectively. Table 1 shows network architectural details for deep feature extraction.

During data curation, we use the SiamFC strategy to crop the target and search images z and x, respectively. We consider the target object as the center of both images because it reflects the most challenging sub-windows that are influential to tracker performance. Since the tracker is fully convolutional, we need not to worry about the model learn

TABLE 1. Proposed network architecture for convolutional feature extraction.

Layer	Target image	Search image	Channel	Filter size	Stride
Input	135 × 135	263 × 263	3	-	-
conv1	63 × 63	127 × 127	192	11 × 11	2
pool1	31 × 31	63 × 63	192	3 × 3	2
conv2	27 × 27	59 × 59	512	5 × 5	1
pool2	13 × 13	29 × 29	512	3 × 3	2
conv3	11 × 11	27 × 27	768	3 × 3	1
conv4	9 × 9	25 × 25	768	3 × 3	1
conv5	7 × 7	23 × 23	512	3 × 3	1

a central bias [5]. We trained the model using GOT10k [56] and ImageNet Large Scale Visual Recognition Challenge-2015 (ILSVRC15) VID [57] benchmarks.

1) TRAINING

To train the model, we randomly selected training image pairs (z, x) from a sequence and adopted the logistic loss function,

$$\mathcal{L}(f(z,x),g) = \frac{1}{|M|}\sum_{m\in M}\log(1+\exp(-f(z,x)[m].g[m])), \qquad (14)$$

where M is the set of possible locations on the response map, $f(z,x)[m]$ is the similarity score, and $g[m] \in \{+1, -1\}$ is the ground truth corresponding to location m. To learn the Siamese network parameters θ, we used SGD to minimize the following function over the training sample N,

$$argmin_\theta \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(f(z_i, x_i), g_i). \qquad (15)$$

We experimentally selected batch size $= 32$ and randomly choose 10 image pairs (z, x) from a video sequence of training benchmarks. We consider maximum distance between z and x to be 100 frames when selecting the image pairs, to ensure robustness to appearance changes. We used SGD to optimize network weights with momentum $= 0.9$, decayed learning rate from 10^{-2} to 10^{-5} exponentially, and set weight decay $= 5e^{-4}$.

2) TESTING

Similarly to SiamFC, we computed tracking treating the first video frame as a stationary template, with subsequent frames considered as search images that change. The response map was calculated independently from template matching between the fixed template and search images. The tracker predicted target position in subsequent frames from the maximum response map score. Finally, we used bicubic interpolation to estimate target location more precisely. We also considered scale penalty $=$ 0.9745 with image scales $= 1.0375^{\{-1,0,+1\}}$ to address target scale changes.

We implemented the proposed tracker using python with the PyTorch deep learning framework and performed all experiments on a desktop with Intel(R) Core(TM) i7-8700 CPU @ 3.20 GHz and Nvidia GeForce RTX 2080 Super GPU. We achieved 61 *fps* average tracker speed during testing.

IV. EXPERIMENTS

Before comparing results on the whole benchmark, we utilize the response map for computing the visualization effects of fused heatmap on the corresponding search image. This visualization results for channel attention module and spatial attention module with siamese architecture represented by CAtt and SAtt, respectively, and SCSAtt, as shown in Fig. 5. We can easily notice that the proposed SCSAtt learns well to compute the target region efficiently than CAtt and SAtt by reducing the distractor and background information significantly. Thus, SCSAtt can ensure high tracking performance than other variants of the proposed tracker.

We also found that the benefit of SCSAtt over the existing attention based trackers is the fast-tracking speed with maintaining high tracking accuracy. Table 2, illustrates the average tracking speed comparison among attention based trackers where we found that our proposed method achieved 61 *fps* which is superior to others. Hence, the proposed tracker would be more applicable to real-time tracking applications.

Furthermore, We evaluated the proposed tracker experimentally on OTB100, OTB50 [24], [25], VOT2016 [26], VOT2017/18 [27], [28], Temple-color-128 (TC-128) [29], and UAV123 [30] benchmarks. The experimental results computed using OTB and VOT toolkit.

A. EVALUATION ON OTB100 BENCHMARK

The popular OTB100 [24], [25] benchmark comprises 100 annotated video sequences, including 11 challenging attributes illumination variation (IV), scale variation (SV), occlusion (OC), deformation (DF), motion blur (MB), fast motion (FM), in-plane rotation (IR), out-of-plane rotation (OR), out-of-view (OV), background clutter (BC), and low resolution (LR). We employed one pass evaluation (OPE) to compute success and precision plots. Success plots show the overall percentage of the overlap score, whereas precision plots show the percentage of center error distance between ground-truth and predicted bounding box. To keep our comparison fair, we accumulated various trackers types, including Siamese based trackers (SiamFC [5], SiamTri [23], SIAMRPN [58], and CFNet [39]), attentional Siamese trackers (MemTrack [40] and MemDTC [45]), correlation filter based trackers (STAPLE [59], CREST [60], SRDCF [61], DSAR-CF [62]), and others (UDT [63], DSiamM [7], and MLT [64]).

Fig. 6 compares the proposed with other considered tracker success and precision outcomes for the OTB100 dataset. The proposed tracker SCSAtt achieves the best performance for both measurement criteria with beyond real-time speed. The proposed model achieved 64.1% and 85.5% score for success and precision plots, respectively, 10.14% and 10.89% superior to the baseline SiamFC tracker. The proposed model achieved 2.40% and 4.27%, and 7.19% and 8.37% increased success and precision, respectively, compared with memory attention mechanism Siamese tracker MemTrack [40] and correlation filter based tracker SRDCF [61], respectively.

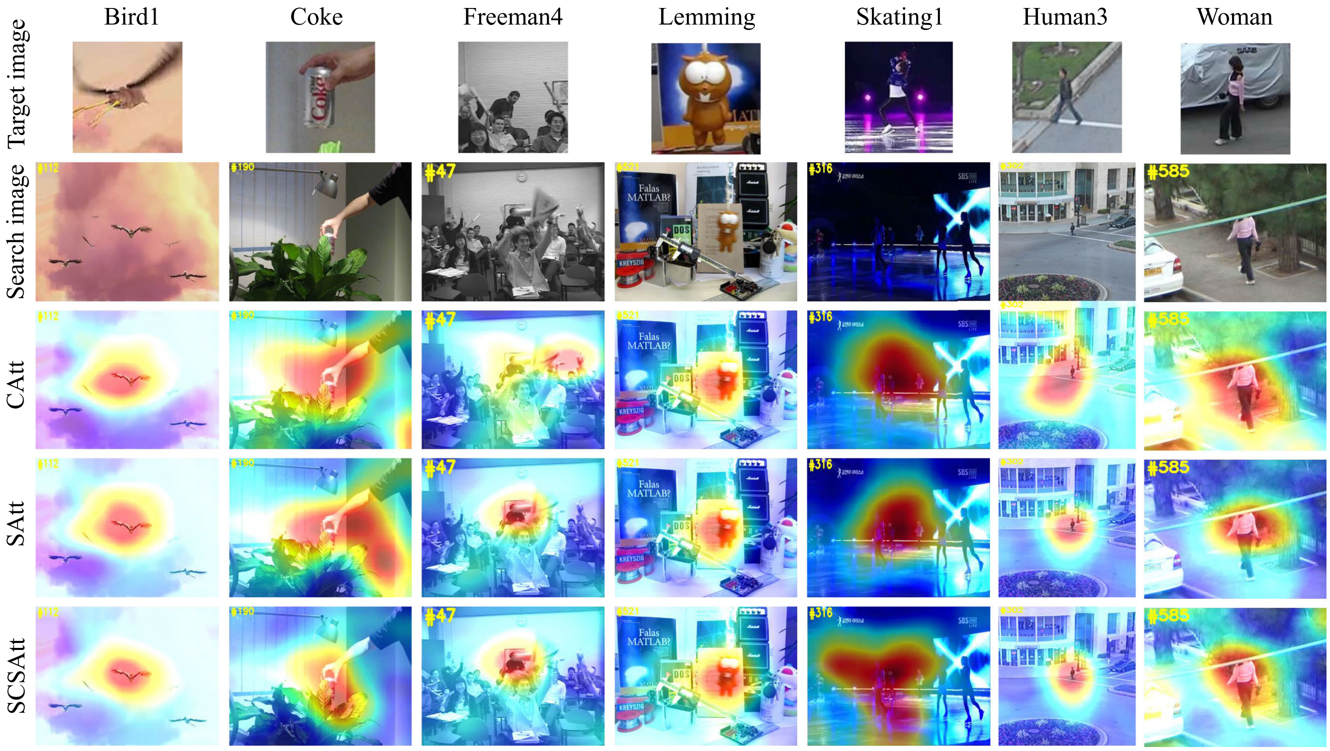

FIGURE 5. We compared the similarity scored heatmap visualization results for the corresponding search images using CAtt, SAtt, and SCSAtt. The response maps between target and search images are fused to the corresponding search images to produce these visualization results. The SCSAtt framework computes the target region better than others by reducing distractor and background information significantly. The target and search image sequences are considered from the OTB100 benchmark.

TABLE 2. Comparison of the average tracking speed among attention based trackers.

Tracker	Ours	SA-Siam [22]	FICFNet [42]	IMG-Siam [43]	FlowTrack [44]	MemTrack [40]	MemDTC [45]
Average speed (*fps*)	61	50	28	50	12	50	40

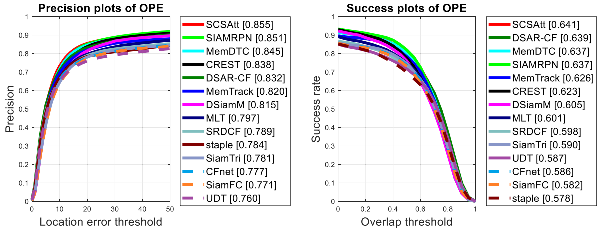

FIGURE 6. Overall precision and success for the considered trackers on the OTB100 benchmark.

We also compared our proposed tracker with the most recent trackers including DSAR-CF [62], MLT [64], and UDT [63]. The proposed tracker achieved 2.76%, 7.28%, and 12.5% improvement in precision score and 0.31%, 6.66%, and 9.20% improvement in success score compared to DSAR-CF, MLT, and UDT trackers, respectively. Moreover, DSAR-CF and MLT perform 16 *fps* and 48 *fps*, respectively, whereas the proposed tracker performs at 61 *fps*. Therefore, SCSAtt maintains a balanced performance in terms of speed and accuracy, which is the main objective of the proposed tracker.

Furthermore, to prove the effectiveness of our proposed tracker, we present the tracker performance for 11 challenges individually for solo comparison in Table 3 and Table 4. The proposed tracker consistently performed outstandingly for compared challenging attributes. Thus, the tracker provides consistent performance even for challenging circumstances. Fig. 10 compares frame-wise visualization, a qualitative comparison for visual understanding. The proposed tracker has significantly improved performance compared with state-of-the-art trackers in several challenging sequences.

B. EVALUATION ON OTB50 BENCHMARK

The OTB50 benchmark is a subset of OTB100, comprising the 50 most challenging video sequences. Fig. 7 compares overall performance for the considered trackers on the OTB50 benchmark. We considered the same trackers that we compared in OTB100 benchmark for evaluating OTB50 benchmark. We observed that the proposed SCSAtt tracker secures the first place among other trackers in the OTB50 benchmark. It exhibits 16.67% and 19.65% increase from the baseline SiamFC in the success and precision score,

TABLE 3. Precision score comparison for various challenging attributes: scale variation (SV), low resolution (LR), occlusion (OC), deformation (DF), motion blur (MB), fast motion (FM), in-plane rotation (IR), out-of-plane rotation (OR), out-of-view (OV), background clutter (BC), and illumination variation (IV) on the OTB100 benchmark.

Tracker	SV	LR	OC	DF	MB	FM	IR	OR	OV	BC	IV
Ours	0.867	0.872	0.817	0.821	0.867	0.856	0.830	0.852	0.769	0.837	0.852
SIAMRPN [59]	0.843	0.868	0.785	0.830	0.833	0.810	0.846	0.853	0.728	0.803	0.866
DSAR-CF [63]	0.829	0.758	0.780	0.793	0.819	0.800	0.778	0.813	0.707	0.815	0.813
MemDTC [45]	0.821	0.867	0.797	0.783	0.803	0.826	0.812	0.843	0.804	0.802	0.807
MemTrack [40]	0.802	0.812	0.762	0.718	0.782	0.817	0.796	0.815	0.720	0.794	0.797
CREST [61]	0.789	0.830	0.786	0.776	0.815	0.794	0.838	0.844	0.734	0.829	0.873
MLT [65]	0.784	0.842	0.764	0.742	0.714	0.724	0.751	0.782	0.646	0.760	0.777
DSiamM [7]	0.775	0.824	0.786	0.752	0.739	0.776	0.793	0.831	0.684	0.792	0.791
SiamTri [23]	0.752	0.884	0.726	0.680	0.744	0.776	0.759	0.761	0.723	0.715	0.751
SRDCF [62]	0.749	0.631	0.735	0.734	0.782	0.773	0.737	0.744	0.597	0.775	0.787
CFNet [39]	0.748	0.861	0.713	0.669	0.761	0.774	0.785	0.758	0.650	0.731	0.763
SiamFC [5]	0.739	0.815	0.722	0.690	0.724	0.758	0.728	0.754	0.669	0.690	0.740
STAPLE [60]	0.731	0.591	0.728	0.751	0.719	0.729	0.751	0.737	0.668	0.749	0.782
UDT [64]	0.714	0.688	0.706	0.670	0.714	0.753	0.741	0.747	0.651	0.749	0.700

TABLE 4. Success score comparison for various challenging attributes: scale variation (SV), low resolution (LR), occlusion (OC), deformation (DF), motion blur (MB), fast motion (FM), in-plane rotation (IR), out-of-plane rotation (OR), out-of-view (OV), background clutter (BC), and illumination variation (IV) on the OTB100 benchmark.

Tracker	SV	LR	OC	DF	MB	FM	IR	OR	OV	BC	IV
Ours	0.642	0.639	0.612	0.591	0.680	0.653	0.616	0.624	0.573	0.620	0.637
SIAMRPN [59]	0.625	0.601	0.592	0.622	0.635	0.613	0.630	0.631	0.550	0.601	0.655
DSAR-CF [63]	0.629	0.571	0.611	0.593	0.646	0.621	0.583	0.615	0.552	0.637	0.646
MemDTC [45]	0.612	0.612	0.604	0.568	0.638	0.634	0.604	0.619	0.590	0.610	0.624
MemTrack [40]	0.606	0.577	0.581	0.539	0.626	0.634	0.596	0.604	0.549	0.599	0.615
CREST [61]	0.575	0.546	0.592	0.569	0.646	0.619	0.609	0.615	0.566	0.618	0.642
MLT [65]	0.591	0.627	0.573	0.536	0.573	0.567	0.572	0.579	0.498	0.579	0.591
DSiamM [7]	0.574	0.584	0.575	0.537	0.576	0.591	0.592	0.600	0.509	0.589	0.597
SiamTri [23]	0.568	0.627	0.549	0.501	0.585	0.594	0.573	0.562	0.543	0.542	0.580
SRDCF [62]	0.565	0.480	0.559	0.544	0.610	0.599	0.543	0.550	0.460	0.583	0.610
CFNet [39]	0.555	0.619	0.536	0.492	0.593	0.590	0.580	0.557	0.480	0.543	0.576
SiamFC [5]	0.557	0.573	0.543	0.506	0.568	0.578	0.551	0.557	0.506	0.523	0.569
STAPLE [60]	0.525	0.394	0.542	0.550	0.553	0.547	0.542	0.534	0.476	0.561	0.589
UDT [64]	0.553	0.510	0.546	0.512	0.582	0.597	0.563	0.566	0.511	0.571	0.551

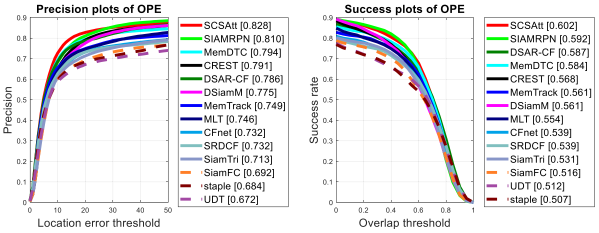

FIGURE 7. Overall precision and success for the considered trackers on the OTB50 benchmark.

respectively. SCSAtt also achieved 7.31%, 5.99%, 11.69% and 7.31% progress in success score and 10.55%, 4.68%, 13.11% and 6.84% progress in precision score than the MemTrack [40], CREST [60], SRDCF [61], and DsiamM [7] trackers, respectively.

Moreover, the proposed method has shown that the performance improvement of 5.34% and 2.56%, 4.28% and 3.03%, 10.99% and 8.66%, and 23.21% and 17.58% in precision and success than the most recent trackers including DSAR-CF [62], MemDTC [45], MLT [64], and UDT [63], respectively. SCSAtt, therefore, constantly outperform on both success and precision scoring metric that demonstrates the effectiveness of our tracker in terms of robustness.

C. EVALUATION ON TEMPLE COLOR-128 BENCHMARK

The temple color-128 (TC-128) benchmark [29] includes 128 video sequences for tracker performance evaluation to address the lack of color information in visual tracking. We compared the proposed SCSAtt tracker performance on this benchmark with current best-practice trackers including SSR-CCOT [65], PTAV [66], SRDCF [61], MEEM [67], MUSTER [68], SAMF [69], DSST [70], Struck [71], KCF [72], TLD [73], and CSK [11]. Fig. 8 shows overall success and precision for the considered tracker frameworks.

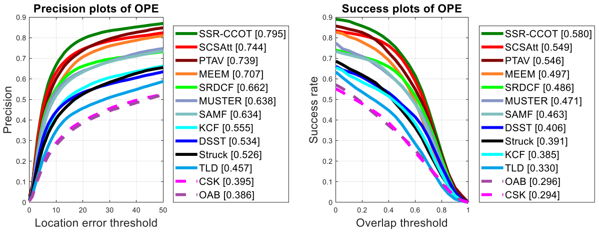

FIGURE 8. Overall precision and success for the considered trackers on the TC-128 benchmark.

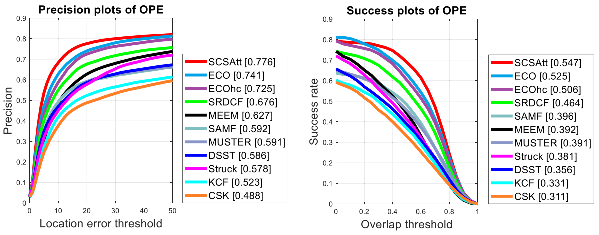

FIGURE 9. Overall precision and success for the considered trackers on the UAV123 benchmark.

The proposed SCSAtt tracker achieved 54.9% and 74.4% success and precision, respectively, significantly improved compared with the other trackers except for SSR-CCOT [65]. But, the speed of the SSR-CCOT tracker is only 1.74 *fps* whereas the proposed tracker achieved very high tracking speed (61 *fps*). Therefore, we believed that our tracker would be useful for real-time applications.

D. EVALUATION ON UAV123 BENCHMARK

In contrast with typical visual object tracking datasets including OTB [24], [25], VOT [26]–[28], and TC-128 [29]; Unmanned Aerial Vehicle (UAV) benchmark [30] provide low altitude aerial videos for object tracking. UAV123 is one of the largest object tracking benchmarks, comprising 123 video sequences with more than 110,000 frames; whereas OTB100, OTB50, and TC128 together contain about 90,000 frames. UAV123 has become more popular recently due to its real-life applications, such as navigation, wild-life monitoring, crowd surveillance. Trackers with a good balance between accuracy and real-time speed will be more useful for these objectives. Since the proposed tracker operates in real-time with high accuracy of 54.7% success score and 77.6% precision score, which are 4.19% and 4.72% increase from one of the prominent tracker ECO for this benchmark as shown in Fig. 9. The ECOhc (60 *fps*) variant of ECO (not real-time) also performs in real-time, but the proposed SCSAtt tracker achieved 8.10% and 7.03% success and precision, respectively, improvement over ECOhc.

E. EVALUATION ON VOT2016 BENCHMARK

The VOT2016 benchmark [26] comprises 60 sequences. In this evaluation, the three most important aspects accuracy (A: higher is best.), robustness (R: lower is best.), and expected average overlap (EAO: higher is best.) are computed to measure the tracker performance. We compared the proposed tracker with the top performing trackers including C-COT [34], Staple [59], DNT [74], MDNet_N (variation of MDNet) [19], SRDCF [61], SiamFC [5], SO-DLT [75], ASMS [8] and MvCFT [76] over VOT2016 benchmark. From the Table 5, we observed that SCSAtt performs well than other trackers in terms of accuracy and robustness. The proposed tracker SCSAtt ranked second for EAO, whereas C-COT ranked best but its accuracy and robustness are less than the proposed tracker. We also compared with underlying SiamFC [5], proposed tracker achieves 28.51% increase in terms of EAO score than the baseline.

TABLE 5. Comparison with the state-of-the-art trackers on the VOT2016 benchmark in terms of accuracy (A), robustness (R), and expected average overlap (EAO).

Tracker	A (↑)	R(↓)	EAO (↑)
Ours	0.554	0.193	0.302
C-COT [34]	0.539	0.238	0.331
Staple [60]	0.544	0.378	0.293
DNT [75]	0.515	0.329	0.278
MDNet_N [19]	0.541	0.337	0.257
SRDCF [62]	0.535	0.419	0.247
SiamFC [5]	0.532	0.461	0.235
SO-DLT [76]	0.516	0.499	0.221
ASMS [8]	0.503	0.522	0.212
MvCFT [77]	0.491	0.606	0.182

TABLE 6. Comparison with the state-of-the-art trackers on VOT2017/18 benchmark in terms of accuracy (A), robustness (R), and expected average overlap (EAO).

Tracker	A (↑)	R(↓)	EAO (↑)
Ours	0.524	0.355	0.199
ECOhc [18]	0.494	0.435	0.238
DSiam [7]	0.512	0.646	0.196
MEEM [68]	0.463	0.534	0.192
SiamFC [5]	0.502	0.585	0.188
DCFNet [6]	0.470	0.543	0.182
DensSiam [78]	0.462	0.688	0.174
ASMS [8]	0.494	0.623	0.169
SSKCF [79]	0.533	0.651	0.166
KCF [73]	0.447	0.773	0.135
SRDCF [62]	0.490	0.974	0.119

F. EVALUATION ON VOT2017/18 BENCHMARK

The VOT2017 [27] and VOT2018 [28] benchmarks are identical, comprising 60 videos. Similar to VOT2016 [26], we compared accuracy, robustness, and expected average overlap for the proposed SCSAtt tracker with state-of-the-art trackers including ECOhc [18], DSiam [7], MEEM [67], SiamFC [5], DCFNet [6], DensSiam [77], ASMS [8], SSKCF [78], KCF [72], and SRDCF [61] for VOT2017 and VOT2018 challenges. The trackers compared outcomes,as shown in Table 6. The proposed SCSAtt tracker exhibits a large margin for robustness and accuracy compared with all other considered trackers, aside from slightly less accuracy than SSKCF, the highest of VOT2017 challenge outcome.

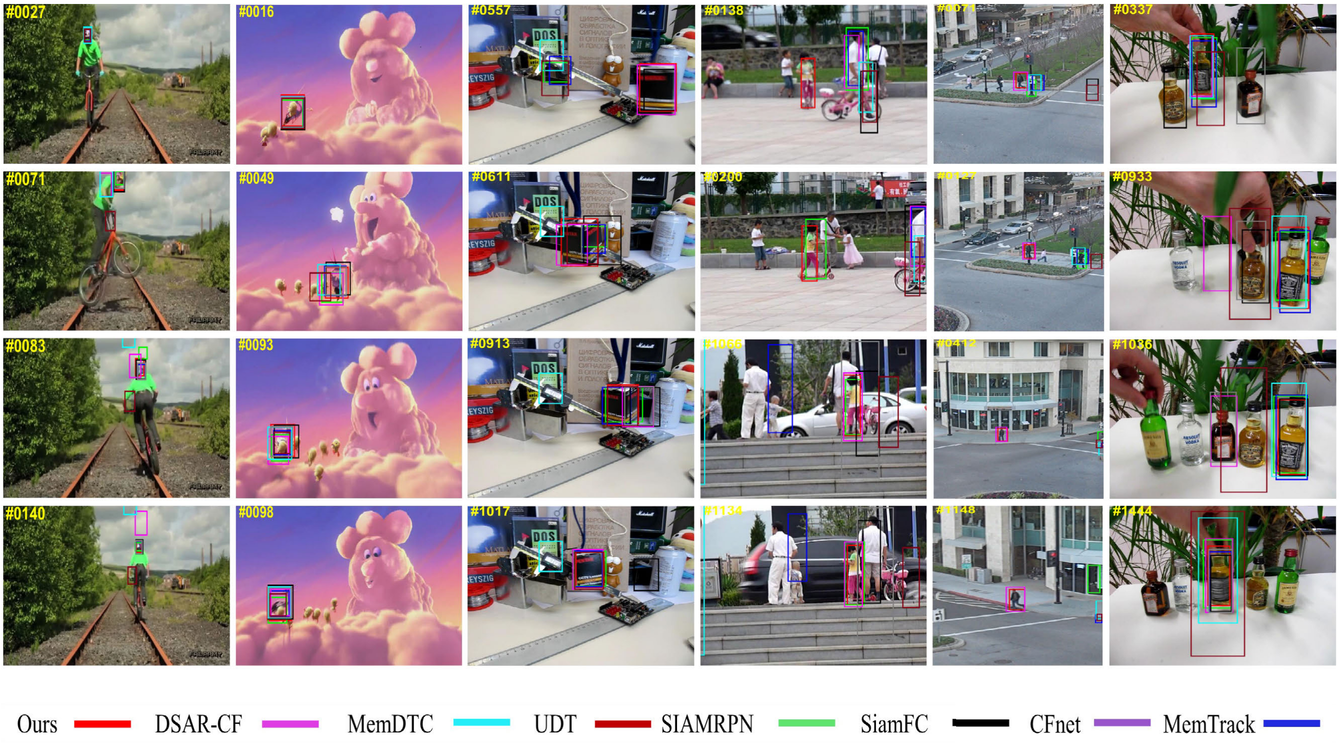

FIGURE 10. **Qualitative comparison of different state-of-the-art trackers for several challenging sequences from the OTB100 benchmark including *biker, bird2, box, girl2, human3,* and *liquor*.**

G. ABLATION STUDY

The appropriate channel and spatial attention configuration is important for the proposed SCSAtt tracker. To validate the selected tracker configuration, we empirically evaluated the performance of various alternate designs. In particular, we measured solo performance for the proposed channel and spatial attention modules, and then considered the integration pattern for the modules. Finally, we investigated how best to incorporate the stacked channel-spatial attention mechanism in single or multiple convolution layers. To keep our comparison rational on the different variants of the proposed tracker, we utilized GOT10k and ILSVRC15 benchmarks to train all variants including the proposed model, and measured their performances on the OTB100 benchmark.

Fig. 11 compares success and precision for these variations on the OTB100 challenging benchmark, where SAtt and Catt systems achieved 62.8% and 83.0%, and 63.1% and 84.1% accuracy and precision, respectively. For the spatial-first attention (SFAtt) case, we first computed spatial attention and then stacked channel attention on it. We empirically validate the results between SFAtt and channel first attention for concluding the stacked channel-spatial attention (SCSAtt) module.

We also validated the proposed tracker by adding the stacked channel-spatial attention mechanism in different convolutional layers. SCSAtt1-5 placed the stacked channel-spatial attention mechanism in all convolution layers since we consider every layer is significant to learn the target features and we did not want to lose any layer's important information. However this configuration performance was significantly lower than the other designs. We also experimented with integrating stacked channel-spatial attention in the third and fifth convolution layer (SCSAtt35), which achieved competitive performance because the latter layers capture the most discriminative features. Therefore, incorporating the stacked channel-spatial attention mechanism solely in the final layer, achieved the best performance.

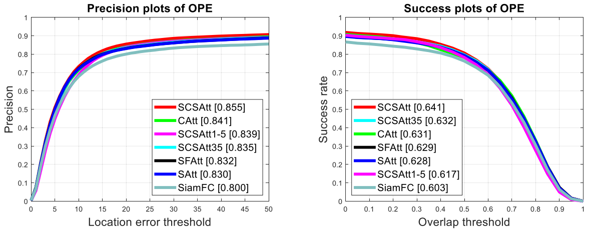

FIGURE 11. **Ablation studies for several variants of the proposed tracker on the OTB100 benchmark.**

H. DISCUSSION

In this article, we utilized Siamese tracking framework to exploit the importance of deep features to improve the robustness of the tracker. We proposed channel attentional module to re-calibrate the deep features channels for better target feature representation, whereas spatial attentional module uses to highlight the important spatial regions in each deep feature channel. We integrated channel and spatial attentional modules within Siamese tracking framework using residual skip connection (called SCSAtt), as shown in Fig. 2.

The SCSAtt learns the most discriminative features to adapt the target features from channel and spatial attentional networks. As each module of the SCSAtt has different functions, the order of the arrangement has an impact on the overall tracker performance. From the spatial feature point of view, the channel attention network applied globally, while spatial attention network responsible to work locally on the feature map. The overall attention tells where to focus, and also enhance the representation of interests. Therefore, the proposed tracker improves the representation ability by utilizing the attention mechanism: highlighting important features and reducing unnecessary ones.

We performed ablation study to show the impact of several tracker's design configurations using Siamese tracking framework. The visualization results of CAtt, SAtt, and ScSAtt as shown in Fig. 5, that represents SCSAtt learns well to compute the target region effectively than CAtt and SAtt by reducing the distractor and background information significantly.

To prove the effectiveness, we also compared our proposed SCSAtt tracker with many state-of-the-art trackers that revealed SCSAtt showed improved performance with real-time tracking facility at 61 *fps* for overall benchmarks including OTB50, OTB2015, VOT2016, VOT2018, UAV123, and TC128. Therefore, the proposed tracker maintains a balanced performance in terms of speed and accuracy.

V. CONCLUSION

This paper proposed a stacked channel-spatial attention mechanism inside the fully convolutional Siamese architecture to suppress irrelevant information and concentrate on object appearance with effective location feature refinement during tracking. The proposed channel attention focused on important feature channels, whereas the spatial attention module responsible for highlighting the object location. We used a cross-feature blending attention mechanism to enhance feature representation power for boosting the tracking performance. We performed extensive experiments to validate the proposed SCSAtt method effectiveness on several challenging benchmarks, including OTB100, OTB50, VOT2016, VOT2017/18, TC -128, and UAV123.

REFERENCES

[1] L. Attard and R. A. Farrugia, "Vision based surveillance system," in *Proc. Int. Conf. Comput. Tool*, Apr. 2011, pp. 1–4.

[2] J. Janai, F. Gäney, A. Behl, and A. Geiger, "Computer vision for autonomous vehicles: Problems, datasets and state of the art," 2017, *arXiv:1704.05519*. [Online]. Available: http://arxiv.org/abs/1704.05519

[3] W.-L. Lu, J.-A. Ting, J. J. Little, and K. P. Murphy, "Learning to track and identify players from broadcast sports videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1704–1716, Jul. 2013.

[4] V. I. Pavlovic, R. Sharma, and T. S. Huang, "Visual interpretation of hand gestures for human-computer interaction: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 677–695, Jul. 1997.

[5] M. Cen and C. Jung, "Fully convolutional siamese fusion networks for object tracking," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*. Athens, Greece: Springer, Oct. 2018, pp. 850–865.

[6] Q. Wang, J. Gao, J. Xing, M. Zhang, and W. Hu, "DCFNet: Discriminant correlation filters network for visual tracking," 2017, *arXiv:1704.04057*. [Online]. Available: http://arxiv.org/abs/1704.04057

[7] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning dynamic siamese network for visual object tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1763–1771.

[8] T. Vojir, J. Noskova, and J. Matas, "Robust scale-adaptive mean-shift for tracking," *Pattern Recognit. Lett.*, vol. 49, pp. 250–258, Nov. 2014.

[9] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 597–606.

[10] D. 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.

[11] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proc. Eur. Conf. Comput. Vis.* Florence, Italy: Springer, 2012, pp. 702–715.

[12] M. Danelljan, F. S. Khan, M. Felsberg, and J. V. D. Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1090–1097.

[13] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[15] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 568–576.

[16] Y. Qi, S. Zhang, L. Qin, H. Yao, Q. Huang, J. Lim, and M.-H. Yang, "Hedged deep tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4303–4311.

[17] S. Gladh, M. Danelljan, F. S. Khan, and M. Felsberg, "Deep motion features for visual tracking," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 1243–1248.

[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. (CVPR)*, Jul. 2017, pp. 6638–6646.

[19] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4293–4302.

[20] M. Fiaz, A. Mahmood, and S. K. Jung, "Convolutional neural network with structural input for visual object tracking," in *Proc. 34th ACM/SIGAPP Symp. Appl. Comput.*, Apr. 2019, pp. 1345–1352.

[21] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3119–3127.

[22] A. He, C. Luo, X. Tian, and W. Zeng, "A twofold siamese network for real-time object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4834–4843.

[23] X. Dong and J. Shen, "Triplet loss in siamese network for object tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 459–474.

[24] 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.

[25] 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.

[26] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Äehovin Zajc, G. F. Dominguez, A. Gupta, A. Petrosino, A. Memarmoghadam, A. Garcia-Martin, A. Montero, A. Vedaldi, A. Robinson, A. Ma, A. Varfolomieiev, and Z. Chi, "The visual object tracking vot2016 challenge results," in *Proc. Eur. Conf. Comput. Vis.*, vol. 9914, Oct. 2016, pp. 777–823.

[27] M. Kristan, "The visual object tracking vot2017 challenge results," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2017, pp. 1949–1972.

[28] M. Kristan, "The sixth visual object tracking vot2018 challenge results," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, p. 5.

[29] P. Liang, E. Blasch, and H. Ling, "Encoding color information for visual tracking: Algorithms and benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5630–5644, Dec. 2015.

[30] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for uav tracking," in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 445–461.

[31] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1442–1468, Jul. 2014.

[32] M. Fiaz, A. Mahmood, S. Javed, and S. K. Jung, "Handcrafted and deep trackers: Recent visual object tracking approaches and trends," *ACM Comput. Surv.*, vol. 52, no. 2, pp. 1–44, Oct. 2019.

[33] M. Fiaz, A. Mahmood, and S. K. Jung, "Deep siamese networks toward robust visual tracking," in *Proc. Visual Object Tracking Deep Neural Netw. Era*. London, U.K.: IntechOpen, 2019.

[34] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 472–488.

[35] 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. Workshop (ICCVW)*, Dec. 2015, pp. 58–66.

[36] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3074–3082.

[37] D. Held, S. Thrun, and S. Savarese, "Learning to track at 100 fps with deep regression networks," in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 749–765.

[38] H. Morimitsu, "Multiple context features in siamese networks for visual object tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 116–131.

[39] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-End representation learning for correlation filter based tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2805–2813.

[40] T. Yang and A. B. Chan, "Learning dynamic memory networks for object tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 152–167.

[41] 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/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4854–4863.

[42] D. Li, G. Wen, Y. Kuai, and F. Porikli, "End-to-end feature integration for correlation filter tracking with channel attention," *IEEE Signal Process Lett.*, vol. 25, no. 12, pp. 1815–1819, Dec. 2018.

[43] X. Qin and Z. Fan, "Initial matting-guided visual tracking with siamese network," *IEEE Access*, vol. 7, pp. 41669–41677, 2019.

[44] Z. Zhu, W. Wu, W. Zou, and J. Yan, "End-to-End flow correlation tracking with spatial-temporal attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 548–557.

[45] T. Yang and A. B. Chan, "Visual tracking via dynamic memory networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jul. 23, 2019, doi: 10.1109/TPAMI.2019.2929034.

[46] H. Li, Y. Li, and F. Porikli, "DeepTrack: Learning discriminative feature representations online for robust visual tracking," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1834–1848, Apr. 2016.

[47] 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.

[48] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "Cbam: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[49] 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. (CVPR)*, Jul. 2017, pp. 3156–3164.

[50] F. S. Khan, J. van de Weijer, and M. Vanrell, "Modulating shape features by color attention for object recognition," *Int. J. Comput. Vis.*, vol. 98, no. 1, pp. 49–64, May 2012.

[51] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3146–3154.

[52] M. Corbetta and G. L. Shulman, "Control of goal-directed and stimulus-driven attention in the brain," *Nature Rev. Neurosci.*, vol. 3, no. 3, pp. 201–215, Mar. 2002.

[53] C. Li and B. Yang, "Adaptive weighted cnn features integration for correlation filter tracking," *IEEE Access*, vol. 7, pp. 76416–76427, 2019.

[54] S. Zagoruyko and N. Komodakis, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," 2016, *arXiv:1612.03928*. [Online]. Available: http://arxiv.org/abs/1612.03928

[55] 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.

[56] L. Huang, X. Zhao, and K. Huang, "Got-10k: A large high-diversity benchmark for generic object tracking in the wild," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Dec. 4, 2019, doi: 10.1109/TPAMI.2019.2957464.

[57] 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.

[58] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8971–8980.

[59] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1401–1409.

[60] Y. Song, C. Ma, L. Gong, J. Zhang, R. W. H. Lau, and M.-H. Yang, "CREST: Convolutional residual learning for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2555–2564.

[61] 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. (ICCV)*, Dec. 2015, pp. 4310–4318.

[62] W. Feng, R. Han, Q. Guo, J. Zhu, and S. Wang, "Dynamic saliency-aware regularization for correlation filter-based object tracking," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3232–3245, Jul. 2019.

[63] N. Wang, Y. Song, C. Ma, W. Zhou, W. Liu, and H. Li, "Unsupervised deep tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1308–1317.

[64] J. Choi, J. Kwon, and K. M. Lee, "Deep meta learning for real-time target-aware visual tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 911–920.

[65] Q. Guo, R. Han, W. Feng, Z. Chen, and L. Wan, "Selective spatial regularization by reinforcement learned decision making for object tracking," *IEEE Trans. Image Process.*, vol. 29, no. 5, pp. 2999–3013, Dec. 2020.

[66] H. Fan and H. Ling, "Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5486–5494.

[67] J. Zhang, S. Ma, and S. Sclaroff, "Meem: Robust tracking via multiple experts using entropy minimization," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2014, pp. 188–203.

[68] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, and D. Tao, "MUlti-store tracker (MUSTer): A cognitive psychology inspired approach to object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 749–758.

[69] M. Kristan, R. Pflugfelder, A. Leonardis, and J. Matas, "The visual object tracking vot2014 challenge results," in *Proc. ECCV*, 2014, vol. 1, no. 2, p. 6.

[70] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Discriminative scale space tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1561–1575, Aug. 2017.

[71] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M.-M. Cheng, S. L. Hicks, and P. H. S. Torr, "Struck: Structured output tracking with kernels," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2096–2109, Oct. 2016.

[72] 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.

[73] 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.

[74] Z. Chi, H. Li, H. Lu, and M.-H. Yang, "Dual deep network for visual tracking," *IEEE Trans. Image Process.*, vol. 26, no. 4, pp. 2005–2015, Apr. 2017.

[75] N. Wang, S. Li, A. Gupta, and D.-Y. Yeung, "Transferring rich feature hierarchies for robust visual tracking," 2015, *arXiv:1501.04587*. [Online]. Available: http://arxiv.org/abs/1501.04587

[76] X. Li, Q. Liu, Z. He, H. Wang, C. Zhang, and W.-S. Chen, "A multi-view model for visual tracking via correlation filters," *Knowl.-Based Syst.*, vol. 113, pp. 88–99, Dec. 2016.

[77] M. H. Abdelpakey, M. S. Shehata, and M. M. Mohamed, "Denssiam: End-to-end densely-siamese network with self-attention model for object tracking," in *Int. Symp. Vis. Comput.* Springer, 2018, pp. 463–473.

[78] J.-Y. Lee and W. Yu, "Visual tracking by partition-based histogram back-projection and maximum support criteria," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, Dec. 2011, pp. 2860–2865.

MD. MAKLACHUR RAHMAN received the B.S. degree in computer science and engineering from the Chittagong University of Engineering and Technology, Bangladesh, in 2013. He is currently pursuing the master's degree in computer science and engineering from Kyungpook National University, South Korea. He was a Software Engineer with Samsung Research and Development Institute, Bangladesh, for a period of about three years. His research interests include computer vision, deep learning, visual object tracking, and autonomous vehicles.

MUSTANSAR FIAZ received the bachelor's degree from the Pakistan Institute of Engineering and Applied Sciences, Pakistan, and the M.S. degree from the Department of Digital Contents, Sejong University, South Korea. He is currently a Research Assistant with the Virtual Reality Laboratory, Kyungpook National University, South Korea. His current research interests include visual object tracking, video object segmentation, and autonomous vehicles.

SOON KI JUNG (Senior Member, IEEE) received the Ph.D. degree in computer science from KAIST, in 1997. From 1997 to 1998, he was a Research Associate with the University of Maryland Institute for Advanced Computer Studies (UMIACS). Since 1998, he has been with the School of Computer Science and Engineering, Kyungpook National University (KNU), Daegu, South Korea, where he is currently a Professor. From 2001 to 2002, he was a Research Associate and from 2008 to 2009, he was a Visiting Faculty with the IRIS Computer Vision Laboratory, University of Southern California. He is the author of over two hundred articles on computer vision and graphics. He holds over twenty patents deriving from his research. His research interests include improving the understanding and performance of the intelligent vision systems and VR/AR systems, mainly through the application of 3D computer vision, computer graphics, visualization, and HCI. He serves as the Vice President for the Korean Computer Graphics Society, the Korean HCI Society, and the Korean Multimedia Society.