Robust Visual Tracking with Channel Attention and Focal Loss

Dongdong Li, Gongjian Wen, *Member, IEEE,* Yangliu Kuai, and Fatih Porikli, *Fellow, IEEE*

***Abstract*—Visual object tracking works as a critical component for many instrumentation and measurement applications such as UAV systems, optical tracking and measuring systems. Recently, the tracking community leads a fashion of end-to-end feature representation learning for visual tracking. Previous works treat all feature channels and training samples equally during training. This ignores channel interdependencies and foreground-background data imbalance, thus limiting the tracking performance. To tackle these problems, we introduce channel attention and focal loss into the network design to enhance feature representation learning. Specifically, a Squeeze-and-Excitation (SE) block is coupled to each convolutional layer to generate channel attention. Channel attention reflects the channel-wise importance of each feature channel and is used for feature weighting in online tracking. To alleviate the foreground-background data imbalance, we propose a focal logistic loss by adding a modulating factor to the logistic loss, with two tunable focusing parameters. The focal logistic loss down-weights the loss assigned to easy examples in the background area. Both the SE block and focal logistic loss are computationally lightweight and impose only a slight increase in model complexity. Extensive experiments are performed on three challenging tracking benchmarks (OTB100, UAV123, TC128). Experimental results demonstrate that the enhanced tracker achieves significant performance improvement while running at a real-time frame-rate.**

***Index Terms*—visual tracking, channel attention, focal logistic los.**

I. Introduction

VISUAL object tracking works as a key component for many instrumentation and measurement applications such as UAV systems, optical tracking and measuring systems, and *et al.* [1], [2], [3]. The task of visual tracking aims at estimating the spatial trajectory of a specified target given its initial state in a video sequence. An ideal tracker can adapt to target appearance variations and achieve invariance to occlusion, deformation, illumination changes and background clutters under complex scenarios. Despite significant process in recent years, persistent visual tracking is still challenging due to the stability-plasticity dilemma.

Driven by the great success of Convolutional Neural Networks (CNNs) in computer vision, many state-of-the-art trackers [4], [5] substitute handcrafted features with deep convolutional features and achieve superior tracking performance on multiple tracking benchmarks [6], [7]. However, these convolutional features are generally trained for a classification task. To fully use the potential of CNN for the tracking task, recent works learn feature representation for visual tracking in an end-to-end manner. Bertinetto et al. [8] proposed a fully convolutional Siamese network (SiamFC) to estimate the feature similarity between an exemplar-candidate pair. Later, Valmadre *et al.* [9] reformulated the closed-form correlation filter as a differentiable layer in a lightweight convolutional neural network (CFNet) and learned deep features tightly coupled to correlation filter tracking. Both SiamFC and CFNet are trained on a large video dataset specialized for video object detection, namely ImageNet-VID [10]. These videos capture all possible appearance variations of target object and thus the learned deep features can effectively encode appearance invariance to some extent. This is the main reason why deep features learned in an end-to-end manner are superior to handcrafted features.

Fig. 1: Visualization of the convolutional features extracted from the first convolutional layer of AdaCFNet. Activations are shown for two sample patches (left), taken from the **Basketball** (top row) and **Bolt** (bottom row) sequence respectively.These convolutional features have different activations on the target object.

Despite the strong power of CNN in feature representation learning, two major limitations have to be addressed for further performance improvement. First, previous works treat all feature channels equally and ignore the channel interdependencies in network design. However, as shown in Fig. 1, different feature channels capture different target information and contribute discriminatively to target representation. Therefore, the network architecture can be further improved to reflect the importance of each channel and boost the tracking performance. Second, discriminative trackers process massive candidate object locations densely sampled across the large search area, resulting in a mass of easy background examples and only a few foreground examples containing the object (as shown in Fig. 2). In this way, classifier training is inefficient as most locations are easy background negatives that contribute little useful training information. Meanwhile, the

D. Li, G. Wen and Y. Kuai are with College of Electronic Science and Technology, National University of Defense Technology, Changsha, Hunan, China (e-mail: moqimubai@sina.cn).

F. Porikli is with Australian National University.

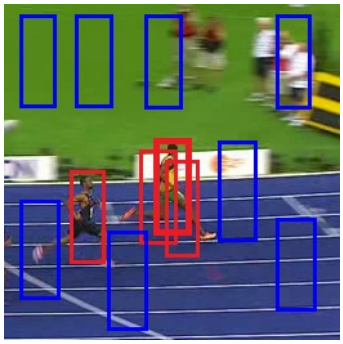


Fig. 2: Foreground-background data imbalance. Blue rectangles represent easy background examples, red rectangles represent foreground examples and hard background examples. Best viewed in color.

easy negatives can overwhelm filter training and lead to degenerate models. Previous works introduce channel reliability to weight feature channels [11] and hard negative mining [12] to tackle foreground-background data imbalance. However, these approaches are intuitively designed and hardly benefit from end-to-end training.

In this paper, we introduce channel attention and focal loss into the network architecture to tackle channel weighting and the foreground-background data imbalance in an end-to-end manner. To perform feature weighting, a Squeeze-and-Excitation (SE) block [13] is coupled to each convolutional layer to generate channel attention. Channel attention reflects the channel-wise importance of each feature channel and is used for feature weighting in online tracking. To alleviate the foreground-background data imbalance, we propose a focal logistic loss by adding a modulating factor to the logistic loss, with two tunable focusing parameters. The focal logistic loss down-weights the loss assigned to easy examples in the background area. The SE blocks and focal logistic loss can automatically down-weight the contribution of less discriminative channels and easy background examples to the training loss respectively.

The main contributions of this work are three-fold. First, we introduce channel attention and focal loss into an end-to-end framework to tackle feature weighting and foreground-background data imbalance. This framework is computationally lightweight and can be combined with many deep trackers with minor modification. Second, based on this framework, we select a correlation filter based deep tracker (CFNet [9]) as the baseline tracker and propose an improved tracker called AdaCFNet, named for its self-adaptive weighting of feature channels and training samples. Finally, extensive experiments have been conducted on three popular tracking benchmarks, including OTB100 [7], UAV123 [14] and TC128 [15]. Experimental results demonstrate that our approach achieves a remarkable performance improvement while running with a real-time frame-rate of 66 fps.

The remaining part of this paper is organized as follows. Section II gives an overview of the most relevant work. Section III provides a detailed description about our approach. Section IV shows the experimental results on different tracking benchmarks. Finally, the conclusion of this paper is provided in section V.

II. Related Works

In this section, we provide a brief overview of the most relevant works. In particular, deep feature based tracking, end-to-end learning based trackers, channel attention mechanisms and foreground-background data imbalance are discussed. The readers are referred to [16] for more details on visual tracking.

A. Deep feature based trackers

Driven by the great success of Convolutional Neural Networks (CNN) in computer vision, deep features have been widely employed in visual tracking due to the superior representation power. A popular trend is the combination of the DCF framework and convolutional features. DeepSRDCF [4] is proposed to substitute hand-crafted features with shallow CNN features in a spatially regularized DCF framework and achieves superior tracking performance. CNN features are extracted from multiple convolutional layers to encode both spatial details and high-level semantics in HCF [17]. The implicit interpolation method is exploited in CCOT [18] to solve the learning problem in the continuous space. Despite significant performance improvement, all the aforementioned methods extract CNN features from a pre-trained object classification network such as VGG [19]. Therefore, feature extraction is separated from filter training in these methods and the tracking results may be suboptimal.

B. End-to-end deep trackers

To benefit from end-to-end learning, researchers design network architectures specialized for the tracking task. These network models are trained offline on large video datasets [10] and evaluated on tracking benchmarks [6], [7] for online tracking. The pioneering deep tracker, MDNet [20], trains a small-scale network by multi-domain learning and separates domain independent information from domain-specific layers. Siamfc [8] poses tracking as a matching problem and learns a similarity metric with a Siamese architecture on the ILSVRC Imagenet Video dataset [10]. CFNet [9] interprets the correlation filter as a differential layer in a Siamese tracking framework and learns convolutional features coupled to DCF learning. Both Siamfc and CFNet achieve end-to-end feature representation learning and run at high frame-rates. The main drawback is their unsatisfying performance on tracking benchmarks. We argue that the aforementioned deep trackers can be further enhanced with better network architectures and loss functions to improve feature learning and tracking performance.

C. Channel attention mechanisms

Channel attention reflects the channel-wise quality of the multi-dimensional feature and is used for feature weighting in visual tracking. Some feature channels capture little information of the target appearance and hardly contribute to target localization. Therefore, these feature channels should be

assigned lower channel reliability. Based on this assumption, CSRDCF [11] estimates the channel reliability based on the ratio between the second and first highest non-adjacent peaks in the channel response map. ECO [21] jointly learns the correlation filter and a projection matrix with a factorized convolution operator. This projection matrix assigns small values to channels with negligible energy and learns a compact set of feature channels with significant energy. Although these channel weighting approaches work well, they are intuitively designed and can hardly benefit from the end-to-end training. Recently, the Squeeze-and-Excitation network (SENet) won the first place in the ILSVRC 2017 classification competition. SENet adaptively recalibrates channel-wise feature responses by explicitly modelling interdependencies between channels. Inspired by SENet, we propose to learn channel attention for the tracking task in an end-to-end deep network.

D. Foreground-background data imbalance

In visual tracking, the majority of negative examples are typically trivial or redundant while only a few distracting negative samples are effective for training of a classifier. Ordinary tracking methods treat all training samples equally and therefore suffer from a drift problem since the classifier training procedure is likely to be dominated by the easily classified background examples. A solution to address this issue is hard negative mining, which is widely used in two- stage R-CNN-like object detectors [22]. This idea has been adopted for online visual tracking in several works. CRT [12] assigned discriminative weights for training samples to improve the contribution of positive samples. MDNet [20] and CFNN [23] integrated hard negative mining steps into minibatch selection and selects a predefined number of hard negative samples in each iteration of the learning procedure. Recently, a new loss function named focal loss[24], [25] works as a more effective alternative to previous approaches for dealing with class imbalance. Intuitively, focal loss automatically down-weight the contribution of easy examples during training and rapidly focus the model on hard examples. In this sense, the potential of focal loss should be further explored to handle the foreground-background data imbalance in visual tracking.

III. Our Approach

In this section, we give a detailed description of our AdaCFNet. We first introduce the overall network architecture of AdaCFNet and then describe the basic components of AdaCFNet, namely the feature extraction sub-network, the correlation filter layer and the focal logistic loss layer. Squeeze-and-Excitation (SE) blocks are integrated into the feature extraction sub-network for channel calibration. A focal logistic loss is designed as a loss layer to replace the original logistic loss in CFNet [9]. At last, details of online model updating and scale estimation are described.

A. Network architecture

Our AdaCFNet follows the two-branch network architecture of CFNet [9]. Each branch contains a feature extraction sub-network with two convolutional layers. We couple a SE block to each convolutional layer. As shown in Fig. 3, the overall network architecture of AdaCFNet mainly consists of a feature extraction sub-network, a correlation filter layer and a focal logistic loss layer.

TABLE I: Architecture of the feature extraction sub-network.

Layer	Support	Chan.Map	Stride	Size.
input				255×255×3
conv1	11× 11	96×3	2	123×123× 96
SE1_global pooling	123×123			1×1× 96
SE1_FC 1	1×1	6×96	1	1×1× 6
SE1_FC 2	1×1	96×6	1	1×1× 96
pool1	3×3		2	61×61× 96
conv2	5× 5	32×48	1	57×57× 32
SE2_global pooling	57×57			1×1× 32
SE2_FC 1	1×1	2×32	1	1×1× 2
SE2_FC 2	1×1	32×2	1	1×1× 32

B. Feature extraction sub-network

The base network architecture (without SE blocks) for feature extraction is similar to the convolutional stage of AlexNet [26]. The dimensions of parameters and activations of the feature extraction sub-network are given in Table I. Batch normalization and rectified linear (ReLU) non-linearities are used after each convolutional layer. Max-pooling is used after the first convolutional layers. The feature stride of the final representation is four.

To achieve adaptive channel weighting, we directly apply the SE block to the feature extraction sub-network. The activations produced after each ReLU layer are fed into a SE block to perform feature calibration. These activations are first passed through a global pooling layer to produce a channel-wise descriptor. This descriptor embeds the global distribution of channel-wise feature map, enabling information from the global receptive filed of the network to be leveraged by its lower layers. A simple gating mechanism (a sigmoid activation) is employed to fully capture channel-wise dependencies. Two fully connected (FC) layers around it limit model complexity and aid generalization. The first FC layer is used for dimensionality reduction with a reduction factor 16 while the second FC layer is used to increase dimensionality. The activations of the SE block work as channel weights adapted to the input feature. Consequently, the SE blocks intrinsically introduce dynamics conditioned on the input, helping to improve the feature discriminability.

The channel weight and its corresponding feature map are visualized to illustrate the effectiveness of the SE blocks in feature calibration. In Fig. 4, the feature maps in the first row contain high energy and capture spatial details (e.g., edge, contour) of the target object and its surrounding background. Those feature maps are more discriminative and yield higher channel weights through the SE blocks than those in the second row. In challenging scenarios, the SE blocks emphasize ‘good’ feature maps and suppress ‘bad’ feature maps, hence enhancing target representation and reducing tracking drift.

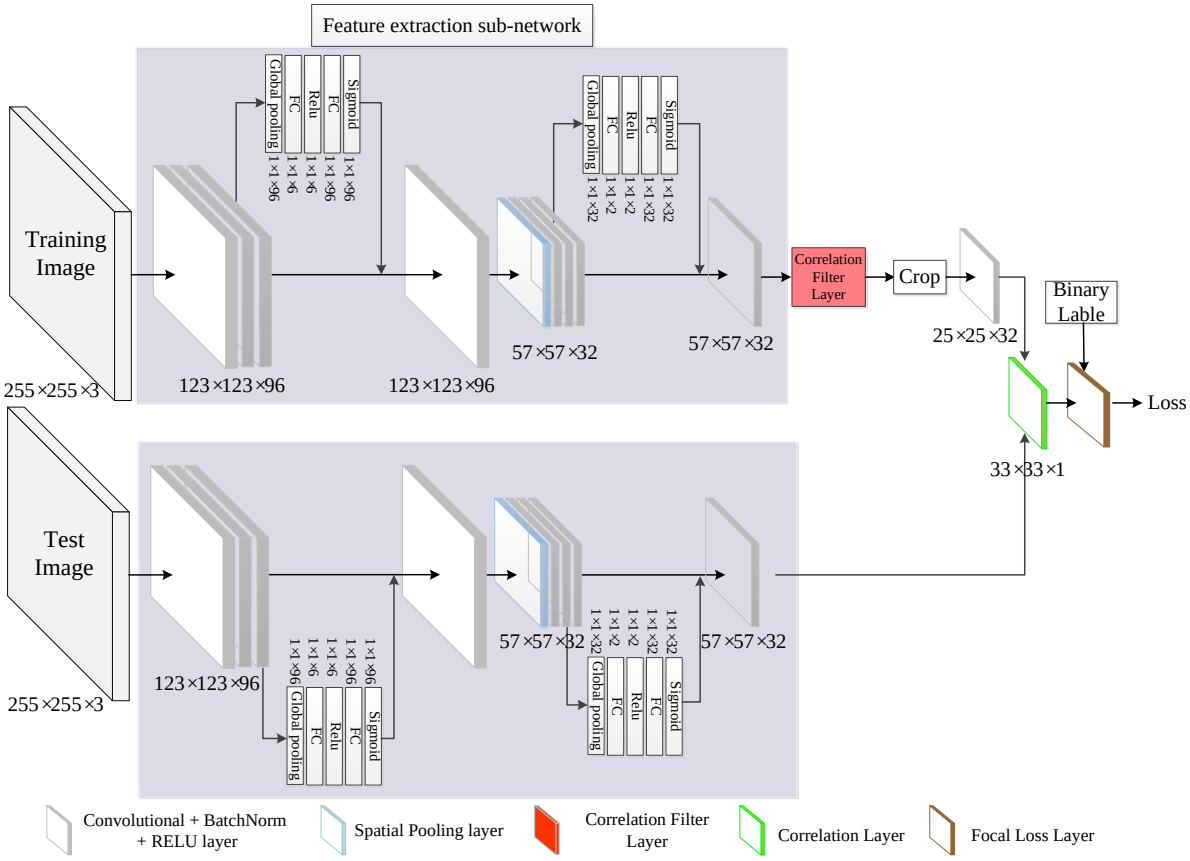


Fig. 3: The overall network architecture of our approach. With CFNet [9] as the baseline network, our network integrates the SE block into the feature extraction sub-network and focal logistic into the loss layer. Best viewed in color.

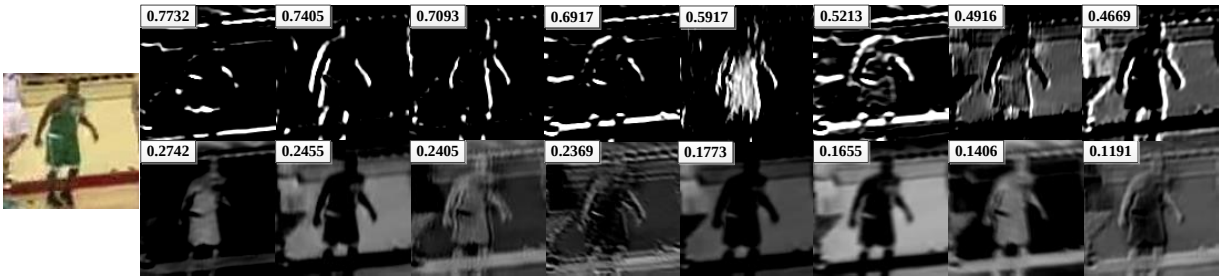


Fig. 4: The visualization of feature maps and corresponding channel weights for a sample patch taken from the **Basketball** sequence. The first and second rows show the features maps with the top 8 highest and lowest channel weights, respectively.

C. Correlation filter layer

The correlation filter layer computes a standard correlation filter template from the feature map generated from the feature extraction sub-network. Given a feature map $x \in R^{M\times N\times C}$, the aim of the correlation filter layer is to learn a correlation filter $w \in R^{M\times N\times C}$. The feature channel $l \in \{1, \cdots, d\}$ of x is denoted by x^l. The desired response y includes a label for each location in the feature map x. The desired correlation filter w is obtained by minimizing the following target function:

$$\varepsilon(w) = \frac{1}{2n}\left\|\sum_{l=1}^{C} x^l * w^l - y\right\|^2 + \frac{\lambda}{2}\sum_{l=1}^{C}\left\|w^l\right\|^2. \quad (1)$$

Here, $*$ denotes the convolution operator, $n = MN$ is the number of training samples and the regularization scalar λ controls the impact of the regularization term.

Based on the circulant assumption, the solution to (1) is derived as

$$\begin{cases} \hat{k} = \frac{1}{n}(\hat{x}^* \cdot \hat{x}) + \lambda \mathbb{1} \\ \hat{\alpha} = \frac{1}{n}\hat{k}^{-1} \cdot \hat{y} \\ \hat{w} = \hat{\alpha}^* \cdot \hat{x} \end{cases} \quad (2)$$

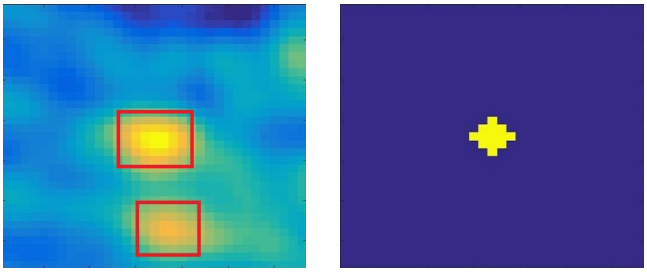


Fig. 5: The score map (left) and ground-truth label (right). Foreground samples and hard background samples are shown in the red rectangle.

Here, $\hat{x}$ represents the Fourier transform of x and $\hat{x}^*$ represents the complex conjugation of $\hat{x}$. The product and division in (2) are point-wise operations.

According to the chain rule of back-propagation map in [9], $\nabla_x l$ can be derived as

$$\begin{cases} \widehat{\nabla_\alpha l} = \hat{x} \cdot (\widehat{\nabla_w l})^* \\ \widehat{\nabla_y l} = \frac{1}{n} \hat{k}^{-*} \cdot \widehat{\nabla_\alpha l} \\ \widehat{\nabla_k l} = -\hat{k}^{-*} \cdot \hat{\alpha}^* \cdot \widehat{\nabla_\alpha l} \\ \widehat{\nabla_x l} = \hat{\alpha} \cdot \widehat{\nabla_w l} + \frac{2}{n} \hat{x} \cdot Re\left\{\widehat{\nabla_k l}\right\} \end{cases} \tag{3}$$

Once the back-propagation of the focal logistic loss l with respect to the feature map x is derived, the correlation filter layer can be formulated as a differential layer in AdaCFNet for end-to-end feature representation learning. The correlation filter layer is followed by a cropping layer to obtain the template for feature correlation.

D. Focal logistic layer

Before introducing our proposed focal logistic loss, we first revisit the logistic loss defined in CFNet [9]. The logistic loss in CFNet is formulated as

$$l(y, v) = log(1 + exp(-yv)) \tag{4}$$

where $v \in R^{m \times n}$ is the real-valued score map of a single exemplar-candidate pair and $y \in \{+1, -1\}$ is its ground-truth label. The logistic loss of the score map is defined as the mean of the individual losses.

$$L(y, v) = \frac{1}{mn} \sum_{i \in [0,m], j \in [0,n]} l(y(i,j), v(i,j)) \tag{5}$$

A notable problem of this logistic loss is that the training procedure may be dominated by easily classified background samples. Fig. 5 shows the score map and its corresponding background-truth labels. There are only a limited number of foreground samples and hard background samples but a substantial amount of easy samples across the whole background of the score map. Therefore, when summed over the large number of easy background samples, these small loss values still overwhelm the total logistic loss and dominate the gradient.

Motivated by the recently proposed focal loss [24], in this subsection, we design a variant of the logistic loss, named focal logistic loss. The proposed focal logistic loss is formulated as

$$l(y, v) = \frac{a}{1 + exp(b \cdot yv)} \cdot log(1 + exp(-yv)). \tag{6}$$

In (6), $\frac{a}{1+exp(b \cdot yv)}$ works as a modulating factor to the logistic loss. The modulating factor adjusts the contribution of each sample to the training loss according to input yv.

As shown in Fig. 6, the modulating factor assigns small

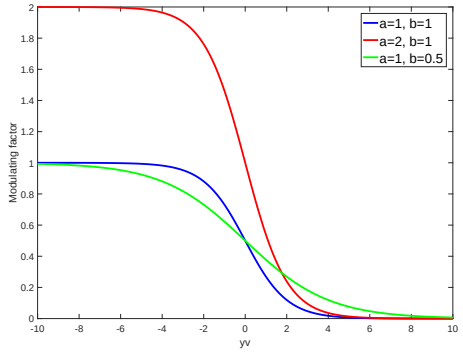


Fig. 6: The modulating factor with respect to the input yv.

weight to the logistic loss if $yv > 0$. It is obvious that $yv > 0$ indicates that both y and v classify this sample to be positive or negative samples. That is, this sample is an easy sample and should be assigned small weight. On the other hand, $yv < 0$ indicates that this sample is misclassified and should be assigned large weight as a 'hard' sample.

Combining the focal loss and logistic loss, our focal logistic loss can be easily implemented as a differential loss layer in AdaCFNet to handle the foreground-background data imbalance.

E. Online Tracking

During online tracking, it is computationally efficient to extract lightweight convolutional features from the feature extraction sub-network with GPU. The score map can be efficiently computed with the fast CNN forward propagation in AdaCFNet. The target location is estimated by finding the maxima on the score map. Scale variation is estimated by processing the search image at several scales with a fixed aspect ratio. To achieve robust online tracking, the correlation filter template derived in the exemplar branch is updated using a rolling average with a fixed learning rate.

Different from the baseline CFNet which assigns a same weight to different feature channels, our AdaCFNet adaptively learns channel-wise weights from the SE block during online tracking. The channel weights estimated by the SE block are channel-aware and target-aware. Channel awareness means that the SE block assigns different weights to different feature channels while target awareness means that the SE block assigns different weights to different targets in each feature channel. Further, target awareness also works for the same target in different frames of a given video. That is, the channel weights change temporally during online tracking. As shown

in Fig. 7, given different target objects, the SE block derives different values in each feature channel and different statistic characteristics among all channels.

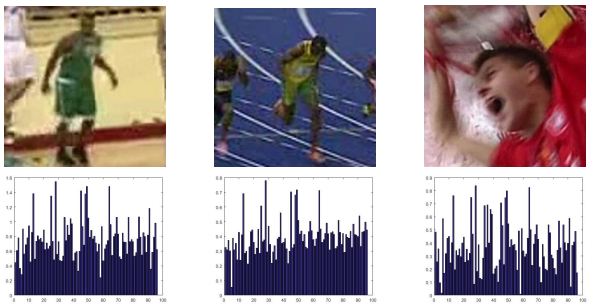


Fig. 7: Three sample patches (first row) and the corresponding channel weights (second row) estimated from the SE block coupled to the first convolutional layers.

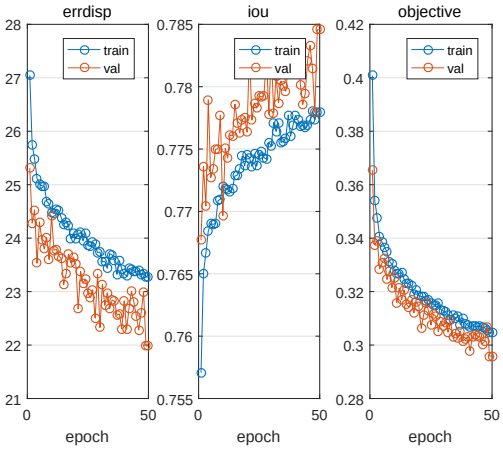


Fig. 8: Training and validation error plots in offline training over 50 epochs.

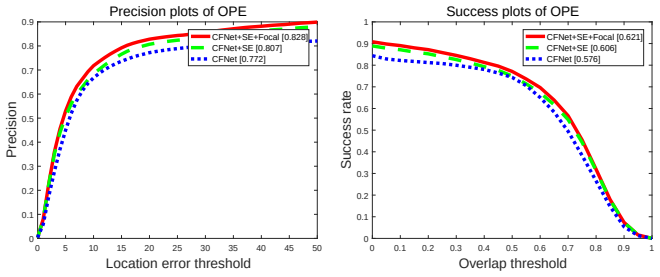


Fig. 9: Precision plot (left) and success plot (right) using one-pass evaluation on the OTB2013 dataset.

IV. Experiments

We test our AdaCFNet on OTB100 [27], UAV123 [14] and TC128 [15].

Evaluation Methodology: We follow the protocol in [7] to conduct experiments on OTB100, TC128 and UAV123. The evaluation is based on two metrics in one-pass evaluation:

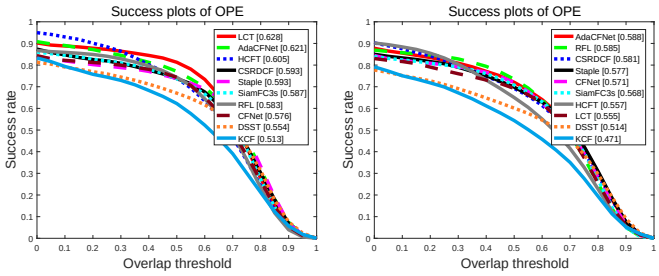


Fig. 10: Success plots of the trackers under comparison on the OTB2013 (left) and OTB100 (right) benchmark datasets.

precision plot and success plot. The precision plot is computed as the percentage of frames in the sequences where Euclidean distance between the ground-truth and the estimated target position is smaller than a certain threshold. The success plot is plotted over the range of Intersection Over Union (IoU) thresholds over all videos. We use the Distance Precision Rate (DPR) at 20 pixels to rank trackers in the precision plot and the Area Under Curve (AUC) to rank trackers in the success plot.

Comparison Scenarios: An ablation study on OTB2013 [27] is done to evaluate the contribution of the SE blocks and focal logistic loss in AdaCFNet. On OTB100, TC128 and UAV123, we compare AdaCFNet with existing trackers in the literature.

Implementation Details: Our AdaCFNet was implemented in Matlab using Matconvnet [28] and trained on the ILSVRC Imagenet Video dataset [10] using both the training and validation sets. We use a reduction factor of 16 in the SE block and set a = 2, b=1 in (6) for the focal logistic loss. The network parameters of AdaCFNet are initialized with the improved Xavier method [29] and optimized with straightforward Stochastic Gradient Descent (SGD) using mini-batches of size 8 during offline training. Training is conducted for 50 epoches as shown in Fig. 8. During online training, we search for the target over three scales $1.0575^{\{-1,0,1\}}$ and update the scale by linear interolation with a factor of 0.52 to provide damping. Comparative experiments were performed on a single NVIDIA GeForce GTX Titan X and an Intel Core i7 CPU at 4.0GHz.

A. Experiments on OTB

OTB2013 [27] is a popular tracking dataset containing 50 fully annotated videos. OTB100 [7] is an extension of OTB2013 and contains 100 sequences. Compared with OTB2013, several more challenging sequences are included in OTB100. In this section, we first conduct an ablation experiment on OTB2013 and then a comparative experiment on OTB100.

1) Ablation study: An ablation study on OTB2013 is conducted to demonstrate the effectiveness of the SE block and focal logistic loss. To verify the contribution of each component individually, we introduce two variants of the baseline tracker CFNet, namely CFNet+SE and CFNet+SE+Focal, by progressively integrating our contributions. CFNet+SE is implemented by integrating the SE blocks into the feature extraction sub-network of CFNet while preserving the logistic

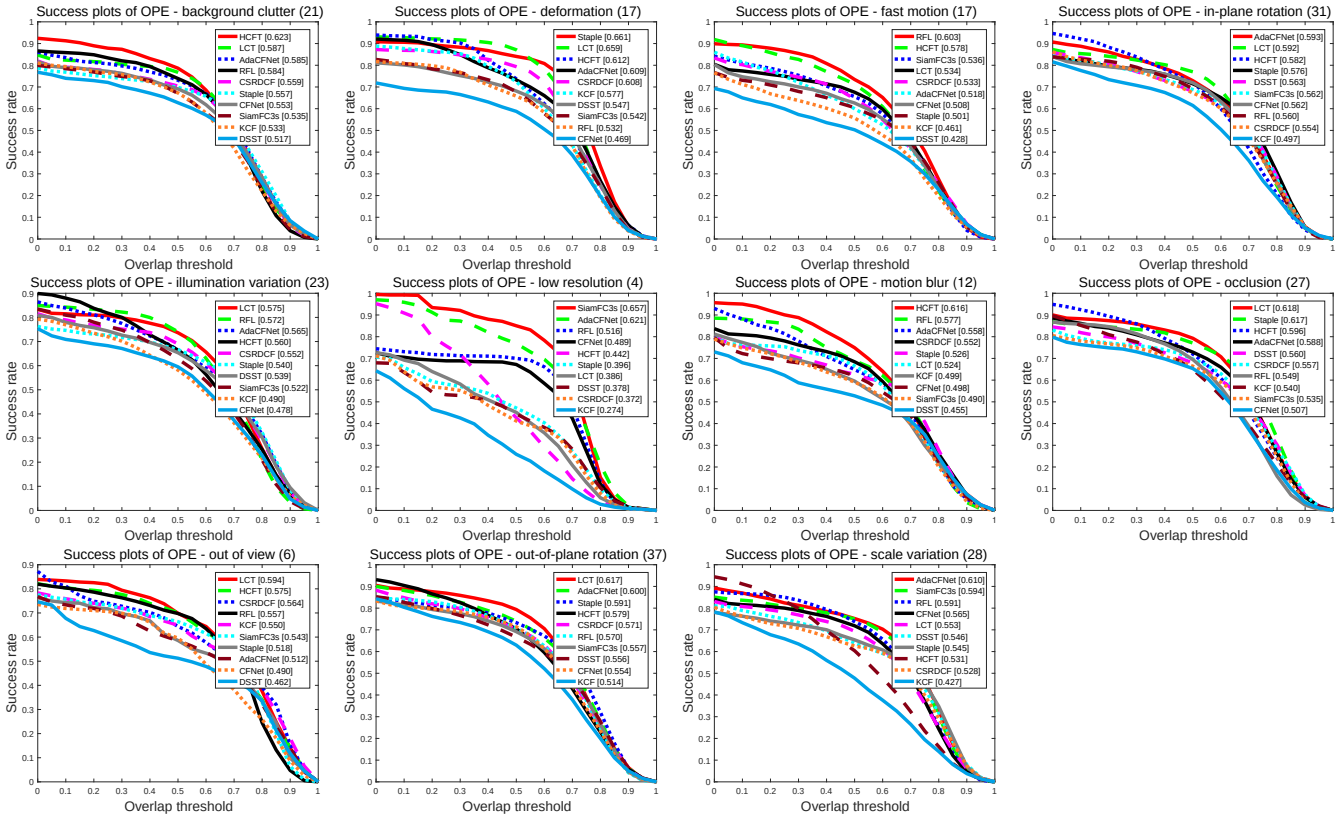


Fig. 11: *Success ratio* plots on 11 attributes of the OTB2013 dataset. All trackers under comparison are ranked by their AUC scores.

TABLE II: Comparisons of the distance precision rate (DPR) at a threshold of 20 pixels, the overlap success rate (OSR) at an overlap threshold of 0.5 and the average frame-rate on OTB2013.

Tracker	AdaCFNet	LCT	Staple	CSRDCF	DSST	KCF	CFNet	SiamFC3s	HCFT	RFL
DPR (%)	82.8	84.8	78.2	80.3	74.1	74.1	77.2	79.4	89.1	78.6
OSR (%)	77.1	81.3	73.8	73.8	67.0	62.2	74.3	73.7	74.0	74.3
FPS	66	5.5	60	13.0	22	243	75	86	11	15

loss. CFNet+SE+Focal is further implemented by replacing the logistic loss in CFNet+SE with the focal logistic loss. As shown in Fig. 9, integrating the SE block into the baseline CFNet leads to an absolute performance improvement of 3.5% in the precision plot and 3.0% in the success plot. The focal logistic loss further improves the performance by 2.1% in the precision plot and 1.5% in the success plot.

2) Overall performance: We further test our AdaCFNet on OTB2013 and OTB100 with comparison to 9 trackers from two typical categories: (1) correlation filter based trackers, including Staple [30], CSRDCF [11], LCT [31], DSST [32] and KCF [33]; (2) deep trackers, including SiamFC3s [8], CFNet [9], RCF [34] and HCFT [17]. Among them, Staple adopts additional color histogram for feature representation. CSRDCF [11] is a spatially constrained correlation filter with channel reliability. LCT [31] is a long-term tracker equipped with a re-detection module. SiamFC3s [8] and CFNet [9] are two CNN based deep trackers while RFL [34] is a LSTM based tracker. HCFT [17] learns correlation filters with hierarchical features extracted from different layers of a deep neural network.

Following the protocol in [27], [7], the success plot of different trackers are shown in Fig. 10. Overall, our tracker ranks second on OTB2013 and first on OTB100. Additionally, we report the distance precision rate (DPR) at 20 pixels, the overlap success rate (OSR) at 0.5 and the average frame-rates on OTB2013 in Table II. Our AdaCFNet achieves a DPR of 89.1% and an OSR of 77.1% while running with a high frame-rate of 66 fps. Compared to the baseline tracker CFNet, our AdaCFNet achieves an absolute gain of 5.6% in DPR and 2.8% in OSR respectively, with only a slight decrease in frame-rate.

3) Attribute-based evaluation: We further analyze the performance of AdaCFNet under different attributes on OTB2013. All the videos in OTB2013 are annotated with 11 differ-

Fig. 12: Tracking screenshots of AdaCFNet, CFNet and SiamFC3s on 11 challenging sequences from OTB100. Best viewed in color.

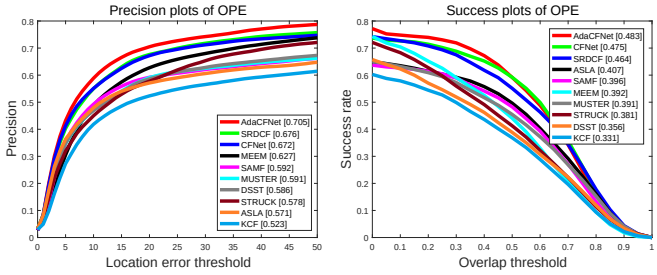


Fig. 13: Precision plot (left) and Success plot (right) using one-pass evaluation on the UAV123 dataset.

ent attributes, namely, illumination variation, scale variation, occlusion, deformation, motion blur, fast motion, in-plane rotation, out-of-plane rotation, out-of-view, background clutter and low resolution. Fig. 11 shows the comparative results achieved by AdaCFNet and other tracking algorithms on these 11 attributes. These trackers are ranked by AUC of the success plots. AdaCFNet achieves consistent superior performance than the baseline tracker CFNet on all 11 attributes. This demonstrates the effectiveness of the SE block and focal logistic loss in boosting feature discriminability.

4) Qualitative comparison: We further visually compare AdaCFNet with two state-of-the-art deep trackers without the SE block and focal logistic loss, CFNet [9] and SiamFC3s [8]. Fig. 12 shows qualitative comparisons of AdaCFNet, CFNet and SiamFC3s on 12 challenging sequences in OTB100. In these sequences, the target undergoes poor illumination (*CarDark*, *Human8*), partial occlusion (*Subway, Box*), similar distracters (*Liquor*, *Coupon*), fast motion (*MotorRolling*, *Skiing*), target deformation (*Bolt*), in-plane rotation (*Rubik*) and

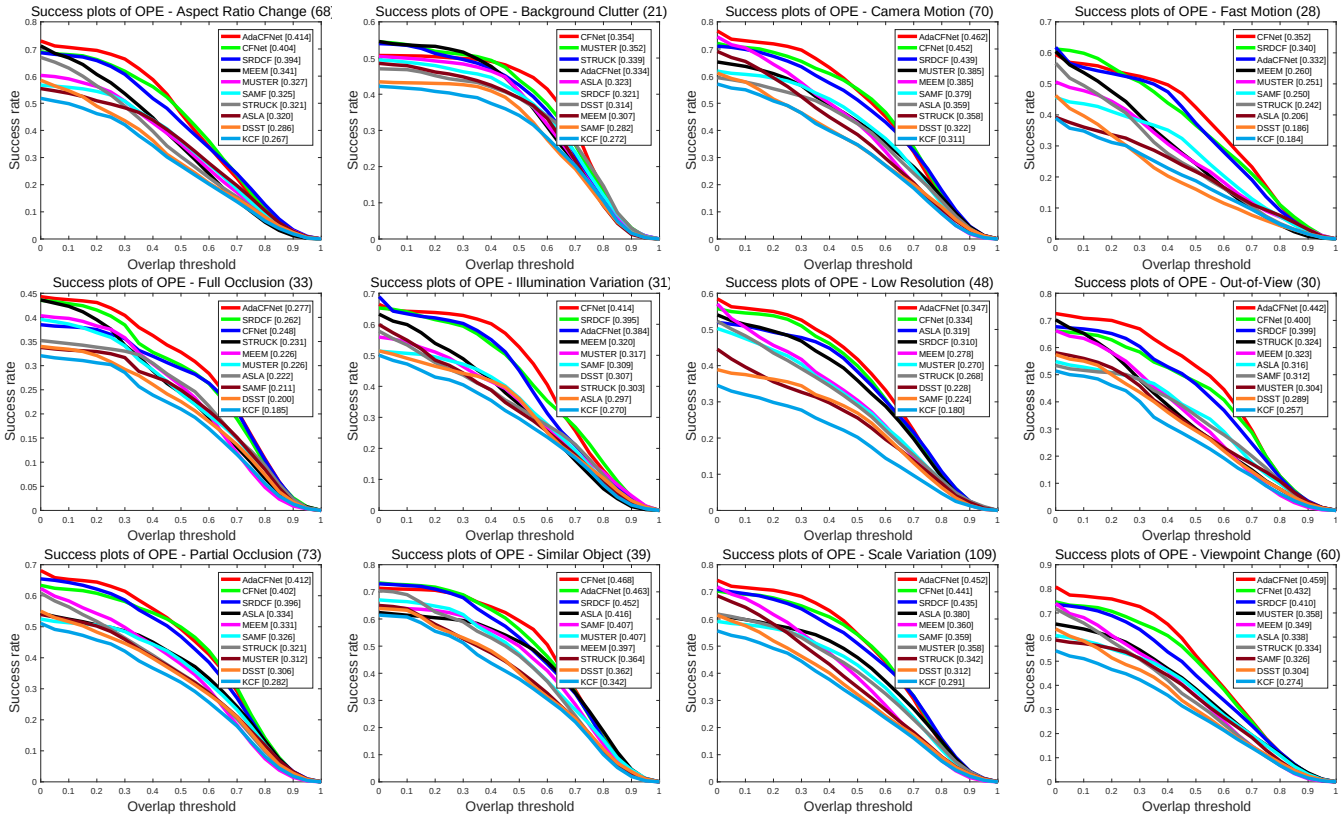


Fig. 14: *Success ratio* plots on 12 attributes of the UAV123 dataset. All trackers in comparison are ranked by their AUC scores.

background clutter (*Board*). It can be seen that our AdaCFNet performs better than CFNet and SiamFC3s under challenging scenarios. This can be attributed to the SE block and focal logistic loss which improve the feature discriminability.

It is worth noting that, like most trackers, AdaCFNet is prone to drifting in presence of long-term and/or full occlusion. As shown in Fig. 12, AdaCFNet drifts to the background in the *Lemming* sequence. We attribute this tracking failure to the boundary effect which leads to a restricted target search area. In future works, we tend to introduce spatial and temporal attention [35], [36] into our framework to alleviate this problem.

B. Experiments on UAV123

In this subsection, we evaluate our AdaCFNet on the UAV123 dataset [14]. UAV123 is a recently introduced aerial video benchmark for low altitude UAV target tracking. It contains 123 aerial videos with more than 110K frames. Different from OTB2013, UAV123 contains both realistic and simulated sequences from an aerial viewpoint. These sequences contain common visual tracking challenges including long-term full and partial occlusion, scale variation, illumination variation, viewpoint change, background clutter and camera motion.

Fig. 13 shows the comparative results achieved by AdaCFNet, CFNet and 8 state-of-the-art trackers included in [14]. Our AdaCFNet achieves the best performance in both the precision plot (65.34%) and success plot (47.11%) while running at an average frame-rate of 58fps.

As shown in Fig. 14, we perform an attribute based analysis of AdaCFNet on the UAV123 dataset. All the videos in UAV123 are annotated with 12 different attributes, namely: aspect ratio change, background clutter, camera motion, fast motion, full occlusion, illumination variation, low resolution, out-of-view, partial occlusion, similar object, scale variation, and viewpoint change. Our AdaCFNet achieves the best performance on 8 out of 12 attributes. Despite no explicit deformation or occlusion handling component, our tracker performs favorably in cases with aspect ratio change and occlusion, as shown in Fig. 14.

C. Experiments on TC128

The TC128 dataset [15] contains 128 color sequences and is specifically designed to evaluate the tracking performance in color sequences. As shown in Fig. 15, our AdaCFNet achieves the best performance among all trackers under comparison. CFNet achieves a precision rate of 63.11% and a success rate of 47.09%. On contrast, our AdaCFNet achieves 65.34% in the precision plot and 47.11% in the success plot. We attribute the favorable performance of AdaCFNet to the large training dataset, namely the VID [10] dataset which contains more than 4000 color videos. Therefore, the learned convolutional features are more effective than the HOG (DSST [32]) and Haar-like features (FCT [37]) in encoding color information.

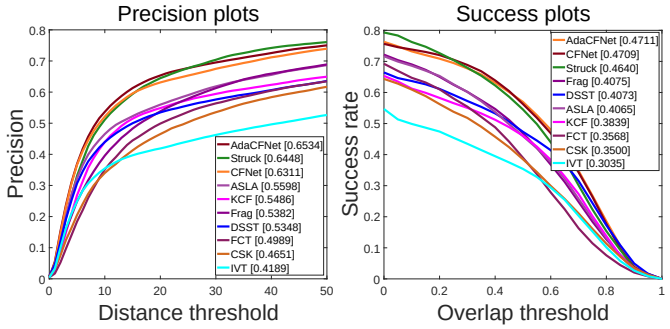


Fig. 15: Precision plots (left) and success plots (right) achieved by different trackers on the TC128 benchmark dataset.

V. Conclusion

We propose a generic end-to-end framework for CNN based trackers to tackle feature calibration and foreground-background data imbalance. This framework significantly increases the feature discriminability at low computational cost and can be combined with any CNN based tracker with minor modification. A lightweight squeeze-and-excitation block is coupled to each convolutional layer to generate channel-wise weight for each feature channel. Focal loss is introduced into the loss layer to tackle the foreground-background data imbalance in network training. Extensive experiments show that our approach improves the tracking performance while running at a real-time frame-rate. Our future work will focus on substituting the feature extraction sub-network with lightweight architectures (e.g. SqueezeNet [38] and ShuffleNet [39]) for higher frame-rates and introducing spatial/temporal attention to achieve higher accuracy.

Acknowledgements

This work was partially supported by the National Natural Science Foundation of China (No. 61602499) and the Australian Research Council's Discovery Projects funding scheme (project no DP150104645).

References

[1] Q. Hu, Y. Guo, Z. Lin, and et al, "Object tracking using multiple features and adaptive model updating," *IEEE Transactions on Instrumentation and Measurement.*, vol. 66, no. 11, pp. 2882–2897, 2017. 1

[2] I. H. Choi, J. M. Pak, C. K. Ahn, and et al, "New preceding vehicle tracking algorithm based on optimal unbiased finite memory filter," *Measurement.*, vol. 73, pp. 262–274, 2015. 1

[3] T. K. kang, Y. H. Mo, D. Pae, and et al, "Robust visual tracking framework in the presence of blurring by arbitrating appearance and feature-based detection," *Measurement.*, vol. 95, pp. 50–69, 2017. 1

[4] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *IEEE International Conference on Computer Vision Workshops*, 2015, pp. 58–66. 1, 2

[5] Y. Song, C. Ma, L. Gong, J. Zhang, R. W. Lau, and M.-H. Yang, "Crest: Convolutional residual learning for visual tracking," in *IEEE International Conference on Computer Vision*, 2017, pp. 2555–2564. 1

[6] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojir, G. Häger, A. Lukežič, and G. Fernandez, "The visual object tracking vot2016 challenge results," Springer. 1, 2

[7] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015. 1, 2, 6, 7

[8] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *European Conference on Computer Vision*. Springer, 2016, pp. 850–865. 1, 2, 7, 8

[9] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. Torr, "End-to-end representation learning for correlation filter based tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2805–2813. 1, 2, 3, 4, 5, 7, 8

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015. 1, 2, 6, 9

[11] A. Lukezic, T. Vojir, L. Cehovin Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6309–6318. 2, 3, 7

[12] K. Chen and W. Tao, "Convolutional regression for visual tracking," *IEEE Transactions on Image Processing*, vol. 27, no. 7, pp. 3611–3620, 2018. 2, 3

[13] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141. 2

[14] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for uav tracking," in *European Conference on Computer Vision*. Springer, 2016, pp. 445–461. 2, 6, 9

[15] P. Liang, E. Blasch, and H. Ling, "Encoding color information for visual tracking: Algorithms and benchmark," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5630–5644, 2015. 2, 6, 9

[16] P. Li, D. Wang, L. Wang, and H. Lu, "Deep visual tracking: Review and experimental comparison," *Pattern Recognition*, vol. 76, pp. 323–338, 2018. 2

[17] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *IEEE International Conference on Computer Vision*, 2015, pp. 3074–3082. 2, 7

[18] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *European Conference on Computer Vision*. Springer, 2016, pp. 472–488. 2

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014. 2

[20] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4293–4302. 2, 3

[21] M. Danelljan, G. Bhat, F. Shahbaz Khan, and M. Felsberg, "Eco: Efficient convolution operators for tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6638–6646. 3

[22] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99. 3

[23] Y. Li, Z. Xu, and J. Zhu, "Cfnn: Correlation filter neural network for visual object tracking." in *IJCAI*, 2017, pp. 2222–2229. 3

[24] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *IEEE international Conference on Computer Vision*, 2017, pp. 2980–2988. 3, 5

[25] X. Lu, C. Ma, B. Ni, X. Yang, I. Reid, and M.-H. Yang, "Deep regression tracking with shrinkage loss," in *European Conference on Computer Vision (ECCV)*, 2018, pp. 353–369. 3

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105. 3

[27] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2411–2418. 6, 7

[28] A. Vedaldi and K. Lenc, "Matconvnet: Convolutional neural networks for matlab," in *23rd ACM International Conference on Multimedia*. ACM, 2015, pp. 689–692. 6

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034. 6

[30] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr, "Staple: Complementary learners for real-time tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1401–1409. 7

[31] C. Ma, X. Yang, C. Zhang, and M.-H. Yang, "Long-term correlation tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5388–5396. 7

[32] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press, 2014. 7, 9

[33] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015. 7

[34] T. Yang and A. B. Chan, "Recurrent filter learning for visual tracking," in *IEEE International Conference on Computer Vision*, 2017, pp. 2010–2019. 7

[35] C. Sun, D. Wang, H. Lu, and M.-H. Yang, "Learning spatial-aware regressions for visual tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8962–8970. 9

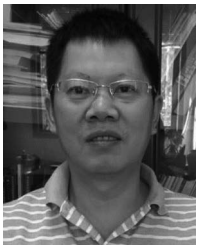
[36] Z. Zhu, W. Wu, W. Zou, and J. Yan, "End-to-end flow correlation tracking with spatial-temporal attention," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 548–557. 9

[37] K. Zhang, L. Zhang, and M.-H. Yang, "Fast compressive tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 10, pp. 2002–2015, 2014. 9

[38] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016. 10

[39] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6848–6856. 10

Dongdong Li is currently pursuing his Ph.D. degree with College of Electronic Science, National University of Defense Technology (NUDT), Changsha, Hunan, China. He has been working on camera calibration, object detection and visual object tracking problems. He serves as a reviewer for Optical Engineering and Optics & Lasers in Engineering.



Gongjian Wen received the B.S., M.S., and Ph.D. degrees from the National University of Defense Technology, Changsha, China, in 1994, 1997 and 2000, respectively. Since 2009, he has been a Professor with the College of Electronic Science and Engineering, National University of Defense Technology, where he served as the head of the fourth department of the National Key Laboratory of Automatic Target Recognition. His research interests include image understanding, remote sensing and target recognition.

Yangliu Kuai received the B.S. degree and the M.S. degree from the National University of Defense Technology in 2013 and 2015 respectively. Currently, she is pursuing her Ph.D. degree with College of Electronic Science, National University of Defense Technology, Changsha, Hunan, China. SHe has been working on camera calibration, object detection and visual object tracking problems.

Fatih Porikli is an IEEE Fellow and a Professor with the Research School of Engineering, Australian National University, Canberra, Australia. He is also acting as the Leader of the Computer Vision Group at NICTA, Australia. He received his Ph.D. degree from NYU. Previously he served as a Distinguished Research Scientist at Mitsubishi Electric Research Laboratories, Cambridge, USA. He has contributed broadly to object detection, motion estimation, tracking, image-based representations, and video analytics. He is the coeditor of two books on Video Analytics for Business Intelligence and Handbook on Background Modeling and Foreground Detection for Video Surveillance. He is an Associate Editor of five journals. His publications won four Best Paper Awards and he has received the R & D 100 Award in the Scientist of the Year category in 2006. He served as the General and Program Chair of numerous IEEE conferences in the past. He has 66 granted patents..