Deep Collaborative Tracking Networks

Xiaolong Jiang[1,2]
jasperj1tmac@163.com
Xiantong Zhen[1,2]
zhenxt@buaa.edu.cn
Baochang Zhang[1]
bczhang@buaa.edu.cn
Jian Yang[4]
yangjianinee@163.com
Xianbin Cao[1,2]
xbcao@buaa.edu.cn

[1] Beihang University, Beijing,China
[2] The Key Laboratory of Advanced Technologies for Near Space Information Systems, Ministry of Industry and Information Technology of China, Beijing,China
[3] Insitute of North Electronic Equipment, Beijing,China

Abstract

Visual object tracking by convolutional neural networks has recently made great progress, which mainly focuses on exploring object appearance; while motion information has been largely overlooked, which however in its nature is essentially important for visual tracking. In this work, we propose deep collaborate tracking network (DCTN), a unified framework that jointly encodes both appearance and motion information for generic object tracking. DCTN establishes a two-stream network with an end-to-end learning architecture that is consisted of a motion net and an appearance net. MotionNet deploys the spotlight filtering in conjunction with the dual pooling operation to fully capture motion information, which is among the first to establish motion detection within an intact CNN architecture; AppearanceNet uses a pyramidal Siamese patch filtering to localize object by multi-scale dense appearance matching. The two nets work collaboratively and encode complementary motion and appearance information to generate two response maps, which are fused to produce the final tracking reuslt. The DCTN is the first generalized framework to model motion and appearance information with deep learning for object tracking. Extensive experiments on VOT2016 and OTB2015 datasets show that the DCTN can achieve high tracking performance, which demonstrates the great effectiveness of exploring both motion and appearance information for visual object tracking.

1 Introduction

Visual object tracking has been extensively studied in computer vision. Given the initial target state in the first frame, a generic tracker is to detect and localize the target relying only on information gathered on-the-fly. To deal with such scarcity of object information, both appearance and motion cues should be fully investigated to characterize and quantify the consistency in object appearance and motion patterns, thus solving the tracking in a searching and matching paradigm. However, existing methods based on the convolutional

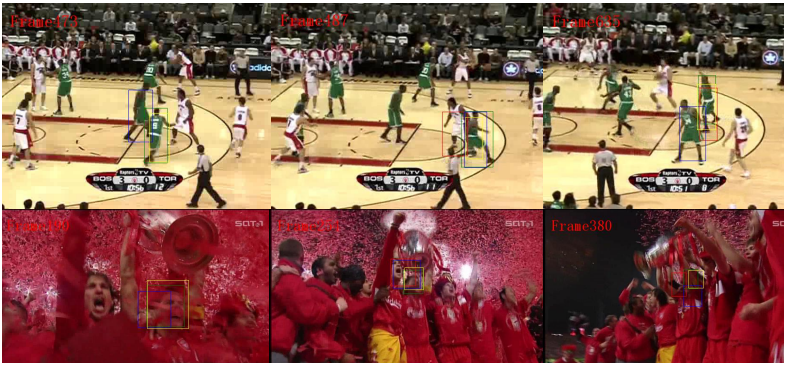

Figure 1: Tracking result comparison of our approach (green) with three state-of-the-art trackers (SiameFC is red, TCNN is blue, CCOT is yellow). As shown, in conditions such as occlusion, deformation, and cluttered background, the object appearance undergoes severe variations, DCTN outperforms the others thanks to the help of the motion cues.

neural network (CNN) have focused mainly on object appearance, while largely overlooking motion information which can also be well explored for improved tracking.

Appearance cues are better studied and developed than its motion counterpart. Early work focused on exploring handcrafted features, e.g. color histogram [59], color name [7], HOG [8], SURF [43], subspace features [38], and superpixels [52] to capture object appearance. With its emergence, CNN dominates appearance cues for the superior representation power [14, 22, 27, 53, 60, 61]. For generic object tracking, appearance cues can be established using CNN via online learned appearance models. However, this approach is challenged by limited sample volumes and inadequate computation efficiency. To solve these limitations, a popular alternative is to deploy CNN pre-trained on large dataset then on-line fine-tune the network to gain video-specific knowledge [34, 48, 50]. The prominent Correlation Filter (CF) paradigm falls into this category [9, 10, 11, 12, 21]. Besides deploying online-trained CNN, off-line trained Siamese network structure has attracted more attentions [4, 19, 46, 51]. This strategy does not solely aim at learning a deep appearance feature representation, but to learn an embedding to match two object instances by characterizing their appearance similarity. In this way, the appearance extractor and discriminator are integrated and trained compactly end-to-end, so that they can co-adapt and cooperate with each other.

Comparing to the flourishing appearance-based strategy, motion cues are less studied in deep tracking methods [13, 25, 56, 63]. In general, motion cues can facilitate tracking in two strategies, but with only a few attempts. For one, hand-crafted or learning based motion models are proposed to enable motion prediction [24, 41], aiming at generating object region-of-interest (ROI) to coarsely locate the object. Secondly, motion detection on basis of Optical Flow [13, 15, 63] or frame differencing [55, 56] are also utilized to provide object proposals. However, the motion information is largely underdeveloped, which would be due to the challenges in modeling motion caused by the following aspects. First, motion is not an universally available cue but is only present in sequence-based tasks, thus has gained less focus; Second, motion information is not as discriminative and representative as the appearance cues which are often hard image evidences; Third, motion cues are often contaminated with noises because of camera motion jitters and background movements. Nonetheless, ob-

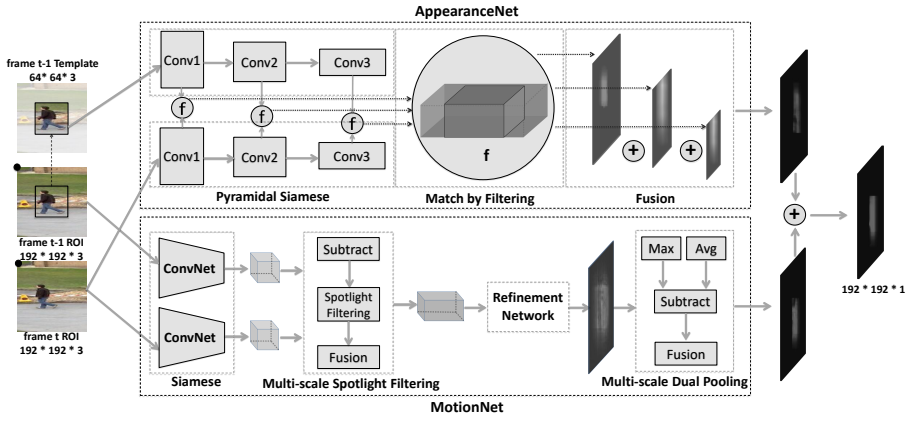

Figure 2: Overview of the proposed deep collaborative tracking network.

ject motion is still indispensable information to realize generic tracking.

Motion and appearance cues are highly complementary and can collaborate with each to improve object tracking. As demonstrated in Figure 1, motion information is capable to help overcome severe appearance variations and occlusions, and distinguish target objects from similar distracters, while appearance cues are dependable to provide hard image evidence to correct spurious and misleading motion information. In order to leverage the strength of both appearance and motion information, in this paper, we propose the Deep Collaborative Tracking Network (DCTN), a new strategy to establish object tracking in a collaborative way by jointly modeling appearance and motion cues in a two-stream network.

Specifically, to better utilize motion cues, we design an end-to-end trained frame differencing motion detection network called MotionNet to provide motion detection responses with robustness to camera motions. Such a design enables integrating motion features to help tracking without adding too much extra computation burden. Besides, the response map provided by MotionNet serves as a spatial attention mechanism to contribute in localizing the target with awareness to the target's shape and size. Within MotionNet, a Spotlight Filtering frame differencing layer first generates motion responses, and then the Dual Pooling layer performs background suppression and foreground enhancement to clean up the responses. To integrate with the MotionNet, appearance cues are encoded by AppearanceNet, which is essentially a Pyramidal Siamese Patch Filter Network to accomplish multi-scale appearance matching via filtering. Both sub-networks deploys generalized conv-nets architectures, outputting two dense response maps which are fused to generate the final estimation of the object state. The main contributions of this work are as follows:

- We propose the deep collaborative tracking network (DCTN) for visual object tracking. DCTN establishes a unified tracking framework of a two-stream network that can fully capture complementary motion and appearance information with an end-to-end learning architecture;

- We design a motion net (MotionNet) to fulfill end-to-end trainable motion detection, where a Spotlight Filtering layer is instantiated to conduct deep frame differencing motion detection, followed by the Dual Pooling layer to perform background suppression and foreground enhancement;

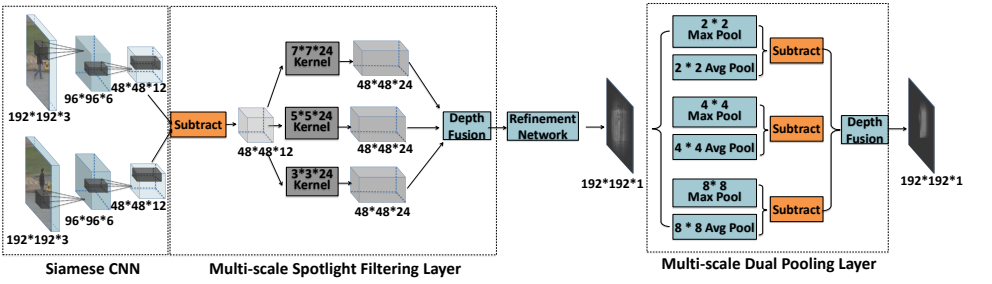

Figure 3: Illustration of MotionNet.

- We design an appearance net (AppearanceNet) for multi-scale appearance matching to achieve object localization, where a Pyramidal Siamese Filtering Network is implemented to compute appearance matching iteratively in a filtering way on each level of a CNN feature hierarchy.

2 Deep Collaborative Tracking Network (DCTN)

In this section we introduce our deep collaborative tracking network, which is a deep collaboration of appearance and motion cues in a two-stream network. Both streams share the same resized image crops as input. In the MotionNet, deep motion detection is conducted to localize the moving object. In the AppearanceNet, a pyramidal Siamese Filtering network is designed to locate the object via filtering based multi-scale appearance matching. The final tracking result is computed by the fusion of response maps output from both networks. The entire network is generalized in design and end-to-end trainable. In section 2.1 we illustrate the MotionNet module. Section 2.2 presents the AppearanceNet module.

2.1 MotionNet

MotionNet is proposed to realize reliable generic motion detection in an end-to-end trainable way. As generic motion detection methods suffer from background noises, MotionNet is designed to also perform background noise suppression and foreground enhancement operations on top of the detections to clean up the response.

The design of MotionNet is shown in Figure 3 in which the convolution layers in the figure indicate convolution units with multiple conv-layers. To our knowledge, we are among the first solving frame differencing based motion detection in a deep learning framework with robustness to camera jitters. MotionNet takes two ROI patches (X_{t-1} is extracted on frame t-1 from bounding box $[x_{t-1}, y_{t-1}, 3*w_{t-1}, 3*h_{t-1}]$, X_t is extracted on frame t at the same bounding box location.) as input, then a pre-processing Siamese CNN structure is implemented to transform the input to representative features. With the produced feature maps, the Spotlight Filtering layer is designed to perform frame differencing motion detection. As follows, a refinement sub-network is deployed using a set of up-convolutional layers to restore the spatial resolution [13]. Subsequently, a Dual Pooling layer is implemented to achieve background suppression and foreground enhancement.

Spotlight Filtering layer. The central idea of Spotlight Filtering is to use element-wise

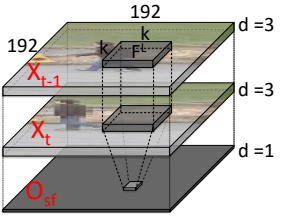

Figure 4: Illustration of the Spotlight Filtering operation.

subtraction and a set of different-sized filtering kernels to filter out the motion detection response given two input image patches. This layer is resilient to camera jitters by applying filtering kernels instead of simple element-wise subtraction, where the level of spatial abstraction dilutes the motion noise introduced by image-level movement. As shown in Figure 4, the Spotlight Filtering starts with aligning two feature maps X_1 and X_2 spatially, then conducting element-wise subtraction between aligned regions with same size as the kernel $F^l \in R^{k\times k}$. The filtered response sf is computed as the summation of all the subtractions.

$$\begin{aligned} sf &= \sum_j \sum_{i=1}^{k*k} (|X^i_{1,j} - X^i_{2,j}|), j \in \Omega \\ O^l_{sf}(X_1, X_2) &= sf(\rho^l(X_1), \rho^l(X_2)) \end{aligned} \tag{1}$$

In the definition, sf_j is the response of the j_{th} filtering location, Ω denotes the set of all filtering locations. $O^l_{sf} \in R^{3*M\times 3*M}$ is the resulting response map on the lth scale, with the same size as the input image patch (with stride = 1 and zero padding). $sf(*)$ denotes the Spotlight Filtering operation, $\rho^l(*)$ indicates the pre-processed feature map. l is added to specify the multi-scale implementation. For each scale l, we use a different kernel size to provide different receptive fields to adapt to motion with varied magnitude. All L response maps from different scales are fused depth-wise with a 1*1 convolution layer before passing to the refinement layers. To design the Spotlight Filtering end-to-end trainable, in implementation it is instantiated using basic convolution operations. In specifics, it first computes the element-wise subtraction between two ROI patches, then apply different size convolution filters on the output subtracted feature map. In this way, the $sf(*)$ operation is further extended into a weighted version, where $W_l \in R^{k*k}$ is the convolution filter on the l_{th} scale:

$$sf'_j = \sum_j \sum_{i=1}^{k} W^i_l * (|X^i_{1,j} - X^i_{2,j}|), j \in \Omega \tag{2}$$

Dual Pooling Layer. After the refinement sub-networks restored the spatial resolution, the refined response map is fed into the Dual Pooling layer. This layer establishes a set of max pooling and average pooling layers with different kernel size to realize foreground enhancement and background suppression. Each kernel offers a level of abstraction, while doing a max pooling will respond to the dominate foreground motion in the region, and an average pooling is similar to the effect of median image background subtraction operations [36] to suppress the background. At the meantime, the dual pooling can also achieve image morphological operations to clean up the response map [18]. By establishing a multi-scale hierarchy with different size kernels, the dual pooling layer is selectively responsive to motions with different magnitudes, as the same design as the multi-scale Spotlight Filtering setups. The final response map at each scale is computed by element-wisely subtracting the max pooled map with the average pooled map. All maps are fused in-depth with 1*1 convolution layers.

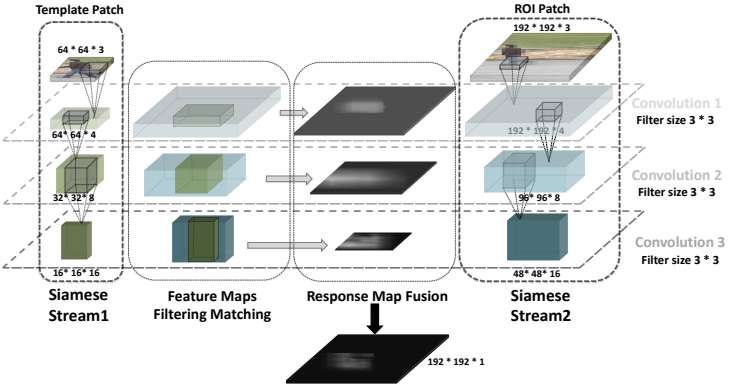

Figure 5: The illustration of the AppearanceNet Structure.

2.2 AppearanceNet

Given an object template and a Region-of-Interest (ROI), AppearanceNet is designed to localize that object within the region. This localization is in form of an appearance matching response map, which is generated by filtering the object template densely across the ROI, and computing cross-correlation along the way with each pair of sampled sub-windows.

Generally, two approaches have been proposed to perform the matching task. For one, candidate object proposals are sparsely sampled in the ROI, and then a binary classification is preformed on them to generate individual similarity score [9, 34, 45, 49]. The second approach(i.e. the Siamese filtering method we implement in this work) performs the matching in a dense filtering way. As the similarity score can be computed efficiently with a cheap cross-correlation computation between the template and the sampled region, this approach affords to operate in a sliding window strategy, resulting in a dense response map traversing the ROI. As the filtering process, it can be configured as a network layer and easily integrated into an end-to-end trainable CNN. The difference between this configured layer and a normal convolution layer is that, instead of computing convolution between a filter and a feature map, it computes the cross correlation between two feature maps. Such an operation is not parameterized, but gradients can flow through easily in back propagation. This Siamese filtering approach can be further categorized as Correlation Filter based [16, 46] or plain feature map based filtering [4, 19], depending on how the similarity is computed at each location. In the former one, extra computation has to be spared for training to maintain the Correlation Filter, and in return the template is more discriminative and representative.

In this work, we adopt the plain feature map template strategy, although performance can be even improved by using more sophisticated techniques. As shown in Figure 5 (in the figure each convolution operation represents a convolution unit), the two L-layered CNN streams share the same parameters, taking the object template and ROI as input. For a filtering based localization task, the spatial resolution information matters, therefore we deploy a shallow network structure with no downsize pooling to preserve the spatial information. Besides, no zero padding is added in the filtering process to keep the resulting response map clean. For the pyramidal implementation, at each of the lth layer the appearance features of the template patch and the ROI patch are parallelly extracted and represented at a particular level of abstraction, so that the multi-scale matching and searching is reasonable. The

feature transformation from spatial dimension to depth dimension is achieved via stride 2 convolutions. At frame t, the template patch $Z_{t-1} \in R^{M\times M\times D}$ is extracted from the tracked bounding box in frame t-1 $[x_{t-1}, y_{t-1}, w_{t-1}, h_{t-1}]$. Meanwhile, ROI $X_t \in R^{3*M\times 3*M\times D}$ is extracted from the bounding box $[x_{t-1}, y_{t-1}, 3*w_{t-1}, 3*h_{t-1}]$ at frame t. The filtering-based matching computation at the lth parallel layer is formulated as:

$$O_t{}^l = f(\phi^l(Z_{t-1}), \phi^l(X_t)) \tag{3}$$

where $O^l{}_t \in R^{3*M\times 3*M\times 1}$ is the resulting response map and ϕ^l denotes the embedded feature extracted at the lth layer, while $f(*)$ is the similarity computation carried out repeatedly through the filtering process.

In our formulation, $f(*)$ is the cross correlation which is fast to compute and back propagate friendly. The difference between (3) and a regular convolution layer in a CNN is that, instead of instantiating another variable W_t as filter to slide through the feature map $\phi^l(X_t)$, here we use another feature map as the filter, where gradient is defused during training. The cross-correlation layer provides an simple method to implement the filtering efficiently within the framework of existing conv-net libraries [4]. The fusion of the total L response maps $\{O_t{}^l | l \in L\}$ is conducted through an $1*1$ convolutional layer. For the fusion of two response maps from both sub-networks, we stack the two response maps in depth, and then also apply a 1*1 convolution layer to generate a depth 1 output.

3 Experiments

We conduct experiments on OTB2015 [54] and VOT2016 [37] datasets. On OTB2015 we show the results of one-pass evaluation using precision and success plot. 16 trackers (ECO[11], CCOT [10], SINT [44], SimaeseFC [4], CFNet[46], Struck [17], HCF [32], SCM [62], TLD [26], ASLA [23], VTD[28], DFT[42], CT[57], IVT[39], CSK[20], MIL[1]) participate in the comparison. We compare with 15 published trackers on VOT 2016 dataset (C-COT [10], TCNN [35], Staple [3], MDNet_N [34], DeepSRDCF [8], SiamAN [4](SiameseFC), MAD [2], ASMS [47], DSST2014 [12], MIL [1], STRUCK2014 [17], FCT [33], STC [58], IVT [39], CTF), the performance is measured by the expected average overlap (EAO) metric.

Experimental results have shown that our DCTN achieves top overall performance by jointly considering the tracking accuracy and speed. DCTN offers a unified tracking network that jointly capturing both motion and appearance information for visual tracking.

3.1 Implementation Details

Network Training. The DCTN network is end-to-end trained from scratch. We use training data generated from NUS-PRO [30], TempleColor128 [31], and MOT2015 datasets [29]. Sequences overlap with the test set are eliminated from the first two. The network inputs are three resized patches, i.e. the object template patch and the ROI centered at the template patch in frame t-1, also the ROI in frame t. ROI in frame t is cropped at bounding box $[x_{t-1}, y_{t-1}, 3*w_{t-1}, 3*h_{t-1}]$. Template patch is resized to 64 * 64, while ROI patches to 192 * 192. Training data patches are pre-cropped and resized offline. In experiments, we use mini-batch size of 16, Xavier initialization, Adam optimizer with weight decay of 0.005, learning rate start at 1 * 10-3, step-wise dropping to 1 * 10-5. The loss L to be minimized is defined as an element-wise ridge loss between two response maps, ($S_{pred} \in R^{M\times M}$ is the network output, $S_{gt} \in R^{M\times M}$ is the ground truth response map, j denotes all elements in the map, $M = 192$). The regularization term in the loss is achieved implicitly using the weight decay method. S_{pred} values are squashed by a sigmoid layer to [0,1]. S_{gt} is generated by

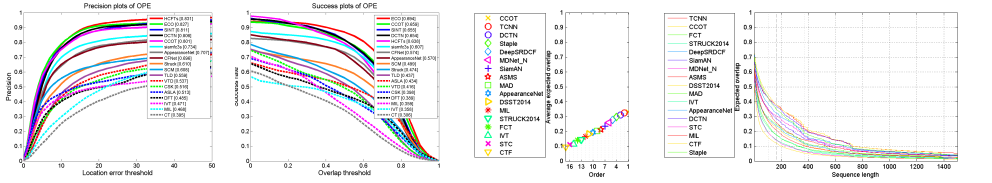

Figure 6: Experiments results on OTB2015 and VOT2016

placing a 2D Gaussian distribution peak at the ground truth bounding box location, with the radius equals to the bigger value of box width and height to ensure complete enclosure.

$$L = \sum_{j} \left\| S^{j}_{pred} - S^{j}_{gt} \right\|^{2} + L_{regularization} \tag{4}$$

Tracking Algorithm. with the trained DCTN network, the tracking in test time is described as follows: at frame t, three image patches are cropped and resized online relying on the estimated object state at frame t-1. These patches are then fed into the DCTN network to compute the fused response map, upon which the estimation of current object state is obtained by searching the maximum value into a bounding box annotation [6].

3.2 Results

OTB2015. As shown in Figure 6, CFNet, SiameseFC, and SINT are latest Siamese based trackers. Amongst the three, SINT is more related to the proposed method as it integrates motion cues by taking optical flow as motion features. Noteworthily, the adoption of optical flow in SINT is off-the-shelf and not end-to-end trainable. Even so, We perform on par with SINT and achieve much faster speed. SINT runs at around 4fps, while DCTN can reach 26.7fps. CFNet adds a correlation layer based on SiameseFC, but the performance gain is not considerable. Speed-wise, SiameseFC and CFnet run at approximately 80fps, but we outperform both of them on success and precision rate measurements. HCF, CCOT and ECO are the elite trackers applying correlation filter within the deep feature hierarchy pyramidally. Such a strategy is highly effective but hinders the real-time performance of the tracker. HCF and CCOT operate at 1fps, ECO speed up CCOT to 8 FPS with the implementation of factorized convolution operators. In comparison, the proposed tracker reports comparative performance with significantly increased speed.

VOT2016. As shown in Figure 6, DCTN reports consistent results as on OTB2015, maintaining an overall favorable performance among all participants. Particularly, DCTN still outperforms SiamAN, and also shows better results than FCT tracker, which uses pyramidal Lucas-Kanade optical flow algorithm to track points object points with motion cues in pyramid levels. TCNN is one of the best tracker in the VOT challenge, but for maintaining multiple CNNs, it runs at only 2 FPS.

Ablation Study. To demonstrate the efficacy of the deep collaboration and highlight the contribution of the motion cues, a further ablation study is conducted. In tracking scenes where object undergoes challenges such as occlusion, deformation, and illumination changes, the appearance of the object alters therefore contaminates the appearance information. Intuitively, motion cues is insensitive to such variations, and thus is helpful in these cases. As depicted in Figure 7, the success plots of DCTN and AppearanceNet in conditions of occlusion, rotation, scale and illumination changes are reported on OTB2015. In the

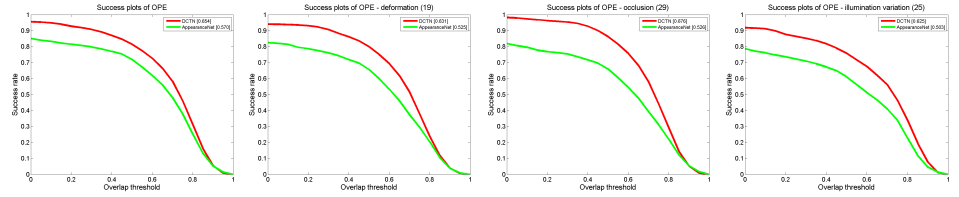

Figure 7: Success plot of DCTN and AppearanceNet in cases of deformation, Occlusion, and illumination variations

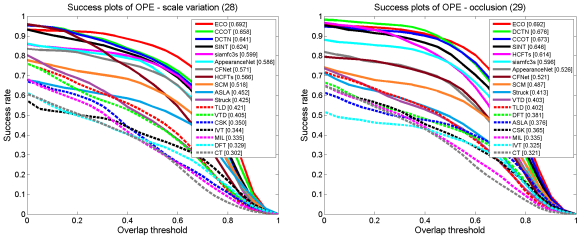

Figure 8: Comparison in scenarios of scale changes and occlusion on OTB2015.

comparison between the DCTN and the AppearanceNet, we aim to demonstrate the contribution of the MotionNet by ablating it out of the DCTN. In accordance with the intuition, overall DCTN outperforms its appearance-only counterpart by 8.4% in the AUC measure. While in the appearance-altering scenarios, the improvement ranges increase to 10.6%, 15%, 12.2%, respectively. Moreover, as shown in Figure 8, the ranking of DCTN increase from the fourth in general to the second best among all in case of occlusion, further indicates the effectiveness of collaborating appearance and motion cues. What's more, the performance improvement of DCTN over AppearanceNet indicates the contribution of the MotionNet.

Table 1: Success plot and speed performance of top-ranking trackers on OTB2015. In the table, superior readings over the proposed DCTN is marked in green.

Trackers	DCTN	ECO	SINT	SiameseFC	CFNet	TCNN	HCF
FPS	27	8	4	86	75	2	1
AUC	0.654	0.694	0.655	0.607	0.574	0.654	0.638

Discussion. DCTN achieves best performance in terms of the overall tracking accuracy and speed. As shown in Table 1, no listed tracker can beat DCTN in both accuracy and speed measures. Specifically, only SiameseFC and CFNet are faster then DCTN, while DCTN outperforms them greatly in tracking accuracy. Meanwhile, in comparison with ECO and SINT who are the only two that have better tracking accuracy, DCTN runs significantly faster. This result can be attributed to the effective collaboration of appearance and motion cues, resulting in more informative and robust feature representation. Particularly, the motion cue is compactly integrated with acceptable computation overhead, and is proven to be beneficial and contributive to handle appearance variations. Besides, the AppearanceNet is also lightweight and cheap-to-compute, and the pyramidal feature hierarchy equips the tracker with scale adaptivity (DCTN ranks the third overall in handling scale variations as shown in Figure 8). What is more, the overall end-to-end training of the DCTN tightly couples all the components in the network, achieving an intact and cooperative solution. The offline training

strategy of DCTN relieves the online training and updating expenses, further contributing to the network efficiency. We highlight that the major contribution made in this work is the unified tracking framework that jointly explores both motion and appearance information. Indeed, the overall performance of DCTN can be further boosted by off-line training the AppearanceNet on larger dataset such as ImageNet Video [40].

Even though DCTN shows leading performance jointly considering tracking accuracy and speed, yet there are still challenges that could lead to degradation of tracking performance, or even tracking failures. Firstly, the MotionNet can absorb the camera motion to some degree, but excessive camera motion may still result in noisy detections by the MotionNet as false positive responses, therefore limits the contribution of the motion features in locating the target; Secondly, as in AppearanceNet we progressively update the target template using newly tracked target state, so that when drastic occlusion or deformation occurs, the template may fail to represent the true appearance of target thus lead to drifting problem. As a solution, in future works we plan to fuse the target template acquires from initialization together with the updated template, so that the fused template can be updated with stability.

4 Conclusions

In this paper, we have presented deep collaborative tracking network, a generalized framework that capturing both motion and appearance for visual tracking. We design the MotionNet to realize deep frame differencing motion detection with background suppression and foreground enhancement. We design AppearanceNet to conduct pyramidal Siamese filtering based appearance matching. Extensive experiments results demonstrate the contribution of the motion cues and the benefits of collaborating motion with appearance in tracking.

5 Acknowledgements

This paper was supported in part by the National Key Research and Development Program of China under Grant 2016YFB1200100, in part by the National Natural Science Foundation of China under Grant 91538204 and Grant 61425014, in part by the Foundation for Innovative Research Groups of the National Natural Science Foundation of China under Grant 61521091.

References

[1] Boris Babenko, Ming-Hsuan Yang, and Serge Belongie. Robust object tracking with online multiple instance learning. *IEEE transactions on pattern analysis and machine intelligence*, 33(8):1619–1632, 2011.

[2] Stefan Becker, Sebastian B Krah, Wolfgang Hübner, and Michael Arens. Mad for visual tracker fusion. In *Optics and Photonics for Counterterrorism, Crime Fighting, and Defence XII*, volume 9995, page 99950L. International Society for Optics and Photonics, 2016.

[3] Luca Bertinetto, Jack Valmadre, Stuart Golodetz, Ondrej Miksik, and Philip HS Torr. Staple: Complementary learners for real-time tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1401–1409, 2016.

[4] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *European conference on computer vision*, pages 850–865. Springer, 2016.

[5] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[6] Martin Danelljan, Gustav Häger, Fahad Khan, and Michael Felsberg. Accurate scale estimation for robust visual tracking. In *British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press, 2014.

[7] Martin Danelljan, Fahad Shahbaz Khan, Michael Felsberg, and Joost Van de Weijer. Adaptive color attributes for real-time visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, Ohio, USA, June 24-27, 2014*, pages 1090–1097. IEEE Computer Society, 2014.

[8] Martin Danelljan, Gustav Hager, Fahad Shahbaz Khan, and Michael Felsberg. Learning spatially regularized correlation filters for visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4310–4318, 2015.

[9] Martin Danelljan, Gustav Hager, Fahad Shahbaz Khan, and Michael Felsberg. Convolutional features for correlation filter based visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 58–66, 2015.

[10] Martin Danelljan, Andreas Robinson, Fahad Shahbaz Khan, and Michael Felsberg. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In *European Conference on Computer Vision*, pages 472–488. Springer, 2016.

[11] Martin Danelljan, Goutam Bhat, F Shahbaz Khan, and Michael Felsberg. Eco: efficient convolution operators for tracking. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA*, pages 21–26, 2017.

[12] Martin Danelljan, Gustav Häger, Fahad Shahbaz Khan, and Michael Felsberg. Discriminative scale space tracking. *IEEE transactions on pattern analysis and machine intelligence*, 39(8):1561–1575, 2017.

[13] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2758–2766, 2015.

[14] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[15] Susanna Gladh, Martin Danelljan, Fahad Shahbaz Khan, and Michael Felsberg. Deep motion features for visual tracking. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pages 1243–1248. IEEE, 2016.

[16] Qing Guo, Wei Feng, Ce Zhou, Rui Huang, Liang Wan, and Song Wang. Learning dynamic siamese network for visual object tracking. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 1–9, 2017.

[17] Sam Hare, Stuart Golodetz, Amir Saffari, Vibhav Vineet, Ming-Ming Cheng, Stephen L Hicks, and Philip HS Torr. Struck: Structured output tracking with kernels. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):2096–2109, 2016.

[18] Hugo Hedberg, Fredrik Kristensen, Peter Nilsson, and Viktor Owall. A low complexity architecture for binary image erosion and dilation using structuring element decomposition. In *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, pages 3431–3434. IEEE, 2005.

[19] David Held, Sebastian Thrun, and Silvio Savarese. Learning to track at 100 fps with deep regression networks. In *European Conference on Computer Vision*, pages 749–765. Springer, 2016.

[20] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In *European conference on computer vision*, pages 702–715. Springer, 2012.

[21] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):583–596, 2015.

[22] Dafei Huang. Enable scale and aspect ratio adaptability in visual tracking with detection proposals. In *British Machine Vision Conference*, 2015.

[23] Xu Jia, Huchuan Lu, and Ming-Hsuan Yang. Visual tracking via adaptive structural local sparse appearance model. In *Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on*, pages 1822–1829. IEEE, 2012.

[24] Zhen Jia, Arjuna Balasuriya, and Subhash Challa. Vision based data fusion for autonomous vehicles target tracking using interacting multiple dynamic models. *Computer vision and image understanding*, 109(1):1–21, 2008.

[25] Samira Ebrahimi Kahou, Vincent Michalski, and Roland Memisevic. Ratm: recurrent attentive tracking model. *arXiv preprint arX-iv: 1510.08660*, 2015.

[26] Zdenek Kalal, Jiri Matas, and Krystian Mikolajczyk. Pn learning: Bootstrapping binary classifiers by structural constraints. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 49–56. IEEE, 2010.

[27] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[28] Junseok Kwon and Kyoung Mu Lee. Visual tracking decomposition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1269–1276. IEEE, 2010.

[29] Laura Leal-Taixé, Anton Milan, Ian Reid, Stefan Roth, and Konrad Schindler. Motchallenge 2015: Towards a benchmark for multi-target tracking. *arXiv preprint arXiv:1504.01942*, 2015.

[30] A Li, M Lin, Y Wu, MH Yang, and S Yan. NUS-PRO: A New Visual Tracking Challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):335–349, 2016.

[31] P. Liang, E. Blasch, and H. Ling. Encoding color information for visual tracking: Algorithms and benchmark. *IEEE Transactions on Image Processing*, 24(12):5630–5644, Dec 2015. ISSN 1057-7149. doi: 10.1109/TIP.2015.2482905.

[32] Chao Ma, Jia-Bin Huang, Xiaokang Yang, and Ming-Hsuan Yang. Robust visual tracking via hierarchical convolutional features. *arXiv preprint arXiv:1707.03816*, 2017.

[33] Mario Edoardo Maresca and Alfredo Petrosino. Clustering local motion estimates for robust and efficient object tracking. In *European Conference on Computer Vision*, pages 244–253. Springer, 2014.

[34] Hyeonseob Nam and Bohyung Han. Learning multi-domain convolutional neural networks for visual tracking. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pages 4293–4302. IEEE, 2016.

[35] Hyeonseob Nam, Mooyeol Baek, and Bohyung Han. Modeling and propagating cnns in a tree structure for visual tracking. *arXiv preprint arXiv:1608.07242*, 2016.

[36] Vladimir Reilly, Haroon Idrees, and Mubarak Shah. Detection and tracking of large number of targets in wide area surveillance. In *European Conference on Computer Vision*, pages 186–199. Springer, 2010.

[37] Giorgio Roffo, Matej Kristan, Jiri Matas, Michael Felsberg, Roman Pflugfelder, Luka Cehovin, Tomas Vojjir, Gustav HAger, Simone Melzi, and Gustavo Fernandez. The visual object tracking vot2016 challenge results. In *IEEE European Conference on Computer Vision Workshops*, 2016.

[38] David A Ross, Jongwoo Lim, Ruei-Sung Lin, and Ming-Hsuan Yang. Incremental learning for robust visual tracking. *International journal of computer vision*, 77(1-3): 125–141, 2008.

[39] David A Ross, Jongwoo Lim, Ruei-Sung Lin, and Ming-Hsuan Yang. Incremental learning for robust visual tracking. *International journal of computer vision*, 77(1-3): 125–141, 2008.

[40] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115 (3):211–252, 2015.

[41] Robin Schubert, Eric Richter, and Gerd Wanielik. Comparison and evaluation of advanced motion models for vehicle tracking. In *Information Fusion, 2008 11th International Conference on*, pages 1–6. IEEE, 2008.

[42] Laura Sevilla-Lara and Erik Learned-Miller. Distribution fields for tracking. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1910–1917. IEEE, 2012.

[43] Duy-Nguyen Ta, Wei-Chao Chen, Natasha Gelfand, and Kari Pulli. Surftrac: Efficient tracking and continuous object recognition using local feature descriptors. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2937–2944. IEEE, 2009.

[44] Ran Tao, Efstratios Gavves, and Arnold WM Smeulders. Siamese instance search for tracking. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pages 1420–1429. IEEE, 2016.

[45] Zhu Teng, Junliang Xing, Qiang Wang, Congyan Lang, Songhe Feng, and Yi Jin. Robust object tracking based on temporal and spatial deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1144–1153, 2017.

[46] Jack Valmadre, Luca Bertinetto, João Henriques, Andrea Vedaldi, and Philip HS Torr. End-to-end representation learning for correlation filter based tracking. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 5000–5008. IEEE, 2017.

[47] Tomas Vojir, Jana Noskova, and Jiri Matas. Robust scale-adaptive mean-shift for tracking. *Pattern Recognition Letters*, 49:250–258, 2014.

[48] Lijun Wang, Wanli Ouyang, Xiaogang Wang, and Huchuan Lu. Visual tracking with fully convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3119–3127, 2015.

[49] Lijun Wang, Wanli Ouyang, Xiaogang Wang, and Huchuan Lu. Stct: Sequentially training convolutional networks for visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1373–1381, 2016.

[50] Naiyan Wang, Siyi Li, Abhinav Gupta, and Dit-Yan Yeung. Transferring rich feature hierarchies for robust visual tracking. *arXiv preprint arXiv:1501.04587*, 2015.

[51] Qiang Wang, Zhu Teng, Junliang Xing, Jin Gao, Weiming Hu, and Stephen Maybank. Learning attentions: residual attentional siamese network for high performance online visual tracking. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2018.

[52] Shu Wang, Huchuan Lu, Fan Yang, and Ming-Hsuan Yang. Superpixel tracking. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1323–1330. IEEE, 2011.

[53] Yifan Wang, Jie Song, Limin Wang, Luc Van Gool, and Otmar Hilliges. Two-stream sr-cnns for action recognition in videos. In *BMVC*, 2016.

[54] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1834–1848, 2015.

[55] Weihua Xiong, Lei Xiang, Junfeng Li, and Xinlong Zhao. Moving object detection algorithm based on background subtraction and frame differencing. In *Control Conference (CCC), 2011 30th Chinese*, pages 3273–3276. IEEE, 2011.

[56] Haiying Zhang and Kun Wu. A vehicle detection algorithm based on three-frame differencing and background subtraction. In *Computational Intelligence and Design (ISCID), 2012 Fifth International Symposium on*, volume 1, pages 148–151. IEEE, 2012.

[57] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang. Real-time compressive tracking. In *European conference on computer vision*, pages 864–877. Springer, 2012.

[58] Kaihua Zhang, Lei Zhang, Qingshan Liu, David Zhang, and Ming-Hsuan Yang. Fast visual tracking via dense spatio-temporal context learning. In *European Conference on Computer Vision*, pages 127–141. Springer, 2014.

[59] Qi Zhao, Zhi Yang, and Hai Tao. Differential earth mover's distance with its applications to visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(2):274–287, 2010.

[60] Xiantong Zhen, Zhijie Wang, Ali Islam, Mousumi Bhaduri, Ian Chan, and Shuo Li. Multi-scale deep networks and regression forests for direct bi-ventricular volume estimation. *Medical image analysis*, 30:120–129, 2016.

[61] J. Zheng, X. Cao, B. Zhang, X. Zhen, and X. Su. Deep ensemble machine for video classification. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2018. ISSN 2162-237X. doi: 10.1109/TNNLS.2018.2844464.

[62] Wei Zhong, Huchuan Lu, and Ming-Hsuan Yang. Robust object tracking via sparsity-based collaborative model. In *Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on*, pages 1838–1845. IEEE, 2012.

[63] Zheng Zhu, Wei Wu, Wei Zou, and Junjie Yan. End-to-end flow correlation tracking with spatial-temporal attention. *arXiv preprint arXiv:1711.01124*, 2017.