Deeper and Wider Siamese Networks for Real-Time Visual Tracking
—— Supplementary Material ——

Zhipeng Zhang
University of Chinese Academy of Sciences&CASIA
zhipeng.zhang@nlpr.ia.ac.cn

Houwen Peng*
Microsoft Research
houwen.peng@micrsoft.com

This supplementary material presents additional details of Section 3 and 5.

- **(Section 3) Analysis of Performance Degradation**.
  We detail the network architectures used in *Fig. 1* and *Tab. 1* of the main manuscript.
- **(Section 5) Experiments**.
  We provide some additional ablation study results.

3. Analysis of Performance Degradation

Details of *Fig. 1*. Tab. I presents the network architectures utilized in *Fig. 1* and *Tab. 2* of the main manuscript, including Alex-5, VGG-10, Incep.-16, ResNet-17, Incep.-22 and ResNet-33. The structures of these networks are slightly different from their original versions [2, 4, 5]. Specifically, Alex-5 consists of five convolution layers and two max-pooling layers. ReLUs follow every convolution layer except for the last one [1]. VGG-10 is a modification version of VGG-13 [4], where the fully-connected layers and the last two max-pooling layers are removed according to [1]. Incep.-16 and Incep.-22 are built upon GoogLeNet[5], but the number of feature channels is reduced to speed training and testing, as shown in Fig. 1. Moreover, ResNet-17 and ResNet-33 have similar structures to the original ResNet-18 [2] and ResNet-34 [2] respectively, where the last fully-connected layers are chopped off.

Details of *Tab. 1*. Tab. II presents the network architectures utilized in *Tab. 1* of the main manuscript. To tune the size of receptive field (RF) and output feature (OFS), we vary the convolutional kernel size in the last few blocks of networks, i.e. the Adjustment Layers in Tab. II. For example, we vary the kernel size from 1 to 5, then the output feature size of Alex-5 changes from 8 to 4, and the receptive field changes from 87-16 to 87+16. Similarly, in ResNet-17, we vary the kernel size from 3 to 7, then the output feature size changes from 7 to 3, and the receptive field changes from 91-16 to 91+16.

5. Experiments

In this section, we present some additional ablation study results. Moreover, in Tab. III, we present the network structures used in *Tab. 8* of the main manuscript.

Ablation study. The ratio of receptive field to the size of exemplar image plays an important role in network design. We further conduct an experiments to evaluate the effects of different exemplar image sizes. Tab. IV shows the results of different exemplar image size ranging from 127-24 to 127+24. The corresponding search image sizes vary from 255-24 to 255+24 accordingly (step=8). We can observe that the optimal ratio lies in a small range from 60% to 80% approximately, which is consistent with our analysis in the main manuscript.

*corresponding author

Table I: Architecture of networks used in Fig.1 and Tab.2 of the main manuscript.

Stage	Alex-5	VGG-10	Incep.-16	Incep.-22	ResNet-17	ResNet-33
Conv1	$\begin{bmatrix}11\times 11, 96, s=2\end{bmatrix}$	$\begin{bmatrix}3\times 3, 64, s=1\end{bmatrix}\times 2$	$\begin{bmatrix}7\times 7, 64, s=2\\ 1\times 1, 64\\ 3\times 3, 192\end{bmatrix}$		$\begin{bmatrix}7\times 7, 64, s=2\end{bmatrix}$	
	3 × 3 max pool, stride 2					
Conv2	$\begin{bmatrix}5\times 5, 256\end{bmatrix}$	$\begin{bmatrix}3\times 3, 128\end{bmatrix}\times 2$	$\begin{bmatrix}\text{InceptionA}^1\end{bmatrix}\times 2$	$\begin{bmatrix}\text{InceptionA}\end{bmatrix}\times 2$	$\begin{bmatrix}3\times 3, 64\\ 3\times 3, 64\end{bmatrix}\times 2$	$\begin{bmatrix}3\times 3, 64\\ 3\times 3, 64\end{bmatrix}\times 3$
Conv3	$\begin{bmatrix}3\times 3, 384\end{bmatrix}\times 2$ $\begin{bmatrix}3\times 3, 256\end{bmatrix}$	$\begin{bmatrix}3\times 3, 256\end{bmatrix}\times 2$ $\begin{bmatrix}3\times 3, 512\end{bmatrix}\times 4$	$\begin{bmatrix}\text{InceptionB}\end{bmatrix}\times 4$ $\begin{bmatrix}1\times 1, 512\end{bmatrix}$	$\begin{bmatrix}\text{InceptionB}\end{bmatrix}\times 5$	$\begin{bmatrix}3\times 3, 128\\ 3\times 3, 128\end{bmatrix}\times 2$ $\begin{bmatrix}3\times 3, 256\\ 3\times 3, 256\end{bmatrix}\times 2$ $\begin{bmatrix}3\times 3, 512\\ 3\times 3, 512\end{bmatrix}\times 2$	$\begin{bmatrix}3\times 3, 128\\ 3\times 3, 128\end{bmatrix}\times 4$
Conv4				$\begin{bmatrix}\text{InceptionB}\end{bmatrix}\times 2$ $\begin{bmatrix}1\times 1, 512\end{bmatrix}$		$\begin{bmatrix}3\times 3, 256\\ 3\times 3, 256\end{bmatrix}\times 6$ $\begin{bmatrix}3\times 3, 512\\ 3\times 3, 512\end{bmatrix}\times 3$
	cross correlation					

[1] Inception A and B modules are illustrated in Fig. 1

Table II: Architecture of networks used in Tab. 1 of the main manuscript.

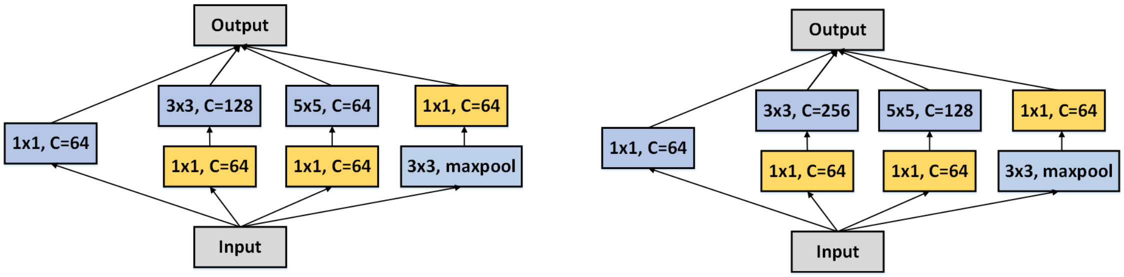
	Stage	Alex-5[1]	VGG-10[2]	ResNet-17[3]	Incep.-22[4]
Basic Layers	Conv1	$\begin{bmatrix}11\times 11, 96\end{bmatrix}$, stride 2	$\begin{bmatrix}3\times 3, 64\end{bmatrix}\times 2$, stride 1	$\begin{bmatrix}7\times 7, 64\end{bmatrix}$, stride 2	
		3 × 3 max pool, stride 2			
	Conv2	$\begin{bmatrix}5\times 5, 256\end{bmatrix}$	$\begin{bmatrix}3\times 3, 128\end{bmatrix}\times 2$	$\begin{bmatrix}1\times 1, 64\\ 3\times 3, 64\\ 1\times 1, 256\end{bmatrix}\times 3$	$\begin{bmatrix}1\times 1, 64\\ 3\times 3, 64\\ 1\times 1, 256\end{bmatrix}\times 3$ $\begin{bmatrix}1\times 1, 64\end{bmatrix}\times 3$
	Conv3	$\begin{bmatrix}3\times 3, 384\end{bmatrix}\times 2$	$\begin{bmatrix}3\times 3, 256\end{bmatrix}\times 2$ $\begin{bmatrix}3\times 3, 512\end{bmatrix}\times 2$	$\begin{bmatrix}1\times 1, 128\\ 3\times 3, 128\\ 1\times 1, 512\end{bmatrix}\times 2$	$\begin{bmatrix}1\times 1, 128\\ 3\times 3, 128\\ 1\times 1, 512\end{bmatrix}\times 3$ $\begin{bmatrix}1\times 1, 128\end{bmatrix}\times 3$
Adjustment Layers		$\begin{bmatrix}k\times k, 256\end{bmatrix}$	$\begin{bmatrix}k_1\times k_1, 512\\ k_2\times k_2, 512\end{bmatrix}$	$\begin{bmatrix}k\times k, 512\end{bmatrix}$	$\begin{bmatrix}1\times 1, 128\\ k\times k, 128\\ 1\times 1, 512\end{bmatrix}\times 1$ $\begin{bmatrix}1\times 1, 128\end{bmatrix}\times 1$
		cross correlation			

[1] The structure of Alex-5 is the same as in SiamFC [1], where the padding operations in convolutions are removed. We vary the kernel size k in the Adjustment Layer from 1 to 5, the receptive field (RF) size changes from 87-16 to 87+16 (step=8), and output feature size (OFS) changes from 8 to 4 accordingly.

[2] Similar to Alex-5, the padding in VGG-10 is also removed. The feature downsampling in the third stage, i.e. 'Conv3', locates in the first layer. To tune the RF and OFS, we vary the kernel size k_1 and k_2. For example, when $k_1 = 3$, if we change k_2 from 3 to 1, RF will vary from 87+16 to 87, and OFS will vary from 4 to 6.

[3] ResNet-17 consists of five residual blocks and two convolution layers. In each residual block, the padding in bottleneck layer is removed, while a cropping operation is inserted in shortcut connection. When we vary the kernal size k in the last convolution from 3 to 7, RF changes from 91-16 to 91+16, and OFS changes from 7 to 3.

[4] The building block for Incep.-22 consists of two branches: a residual block and a 1×1 convolution. Here, we do not adopt convolutions with large kernel size, such as the 5×5 convolution in Fig. 1, since that it makes RF too large. In the residual block, the padding in bottleneck layer is removed, while a cropping operation is inserted behind the 1×1 convolution. This modification ensures the feature sizes of two branches are compatible before concatenation. To tune the size of RF and OFS, we vary the kernel size k from 1 to 7.



Inception A **Inception B**

Figure 1: Inception modules used in Incep.-16 and Incep.-22. These two modules have the same structure as in GoogLeNet[5], but the number of output feature channels is different. The 1×1 convolutions are used for decreasing feature channels.

Table III: Network architectures used in Tab. 8 of the main manuscript.

	Stage	CIResNet-16[1]	CIResNet-19[2]	CIResNet-22[3]	CIResIncep.-22[3]
Basic Layers	Conv1	7×7, 64, stride 2			
	Conv2	2×2 max pool, stride 2			
		$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$ $\begin{bmatrix} 1\times1, 64 \end{bmatrix} \times 3$
	Conv3	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 1$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 3$ $\begin{bmatrix} 1\times1, 128 \end{bmatrix} \times 3$
Adjustment Layers		$\begin{bmatrix} 1\times1, 128 \\ k\times k, 128 \\ 1\times1, 512 \end{bmatrix} \times 1$	$\begin{bmatrix} 1\times1, 128 \\ k\times k, 128 \\ 1\times1, 512 \end{bmatrix} \times 1$	$\begin{bmatrix} 1\times1, 128 \\ k\times k, 128 \\ 1\times1, 512 \end{bmatrix} \times 1$	$\begin{bmatrix} 1\times1, 128 \\ k\times k, 128 \\ 1\times1, 512 \end{bmatrix} \times 1$ $\begin{bmatrix} 1\times1, 128 \end{bmatrix} \times 1$
		cross correlation			

[1] For CIResNet-16, RF ranges from 93-16 to 93+24 when k ranges from 5 to 10.

[2] For CIResNet-19, RF ranges from 93-16 to 93+24 when k ranges from 3 to 8.

[3] For CIResNet-22 and CIResIncep.-22, RF ranges from 93-16 to 93+24, when k ranges from 1 to 6.

Exemplar Image Size	103	111	127-8	127	127+8	143	151	159
Ratio (RF=93)	0.90	0.84	0.78	0.73	0.69	0.65	0.62	0.58
SiamFC-AUC	0.58	0.61	0.63	0.67	0.67	0.66	0.65	0.64
SiamRPN-AUC	0.61	0.63	0.64	0.67	0.65	0.65	0.64	0.63

Table IV: Influence of different exemplar image sizes. Here, CIResNet-22 servers as the backbone network for SiamFC [1] and SiamRPN [3]. The results are evaluated on OTB-13 dataset.

References

[1] 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*, pages 850–865. Springer, 2016. 1, 2, 3

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[3] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu. High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8971–8980, 2018. 3

[4] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[5] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 1, 3