Supplementary Material
Learning Tracking Representations from Single Point Annotations

Qiangqiang Wu Antoni B. Chan
Department of Computer Science, City University of Hong Kong
qiangqwu2-c@my.cityu.edu.hk, abchan@cityu.edu.hk

In this supplementary material, we provide additional quantitative comparisons, qualitative visualizations and ablation studies. Section A provides the comparison between our SoCL-TransT and TransT [3] under the same annotation time cost. Section B shows the comparison between our SoCL-Siam and Box-Siam with the same annotation time cost and same training videos. Section C contains the ablation study on the usage of projection head in both Siamese and correlation filter trackers. We then introduce the detailed preprocessing step in Section D in order to obtain smoother target objectness prior (TOP) maps for more effective representation learning. Finally, Section E shows the qualitative visualization of the soft sample generation, including both global soft template (GST) and soft negative sample (SNS) generation.

A. Comparison with Same Annotation Time Cost using TransT

In this section, we compare the proposed SoCL-TransT to its fully supervised baseline (i.e.,TransT [3]) trained with bounding boxes under the same time cost of annotation. Note that SoCL-TransT is trained on the whole GOT-10k dataset with point annotations, and its total annotation time cost is about 1.2K hours. We randomly sample training videos with bounding box annotations from GOT-10k to meet the same annotation time requirement (1.2K hours), and then use these videos to train TransT. As illustrated in Table 1, our SoCL-TransT achieves better performance than TransT in terms of all the metrics on the three large-scale tracking datasets. For example, SoCL-TransT achieves favorable AUC, P_{Norm} and P on the LaSOT dataset by respectively improving 7.1%, 8.9% and 10.1% compared to TransT, which demonstrates the effectiveness of the proposed new annotation schema for training scale regression-based trackers.

Table 1. Comparison of SoCL-TransT and TransT trained using the same annotation time cost (i.e., 1.2K hours) on GOT-10k [5], TrackingNet [6] and LaSOT [4]. The best results are highlighted.

Trackers	GOT-10k			TrackingNet			LaSOT		
	AO	$SR_{0.5}$	$SR_{0.75}$	AUC	P_{Norm}	P	AUC	P_{Norm}	P
TransT [3]	59.1	68.0	51.9	72.4	76.4	67.1	48.9	50.1	46.8
SoCL-TransT	**62.2**	**72.4**	**52.5**	**75.0**	**80.5**	**71.1**	**56.0**	**59.0**	**56.9**

Table 2. Comparison of Box-Siam and SoCL-Siam trained using the same annotation time costs (i.e., hours) and the same number of training videos in terms of AUC on OTB-13. The best results are highlighted.

Annotation time cost	110h	220h	440h	880h
Box-Siam	52.2	54.3	58.1	58.8
SoCL-Siam	**56.7**	**58.2**	**59.8**	**60.9**

B. Comparison with Same Annotation Time Cost and Same Training Videos

In Section 4.3 of the main paper, under the same annotation time cost, the training sets for baselines or SoCL are selected by sampling whole videos to ensure the same time cost. However, this means that the baseline methods are trained on fewer videos (possibly seeing less backgrounds and less objects) compared to SoCL. In this section, we guarantee that SoCL-Siam and Box-Siam use the same number of training videos and same annotation cost. Specifically, for each video, SoCL-Siam uses all its video frames while Box-Siam uses about 22.2% (i.e., 1/4.5) of the frames. The comparison is shown in Table 2. We can see that our SoCL-Siam still significantly outperforms Box-Siam under various annotation time costs, which shows the superiority of our SoCL and demonstrates that SoCL can also learn effective temporal correspondences from soft representations for visual tracking.

C. Ablation Study on Projection Head

The usage of a projection head has been well explored in the contrastive learning community [2, 7]. Commonly, using a projection head for end-to-end contrastive learning can

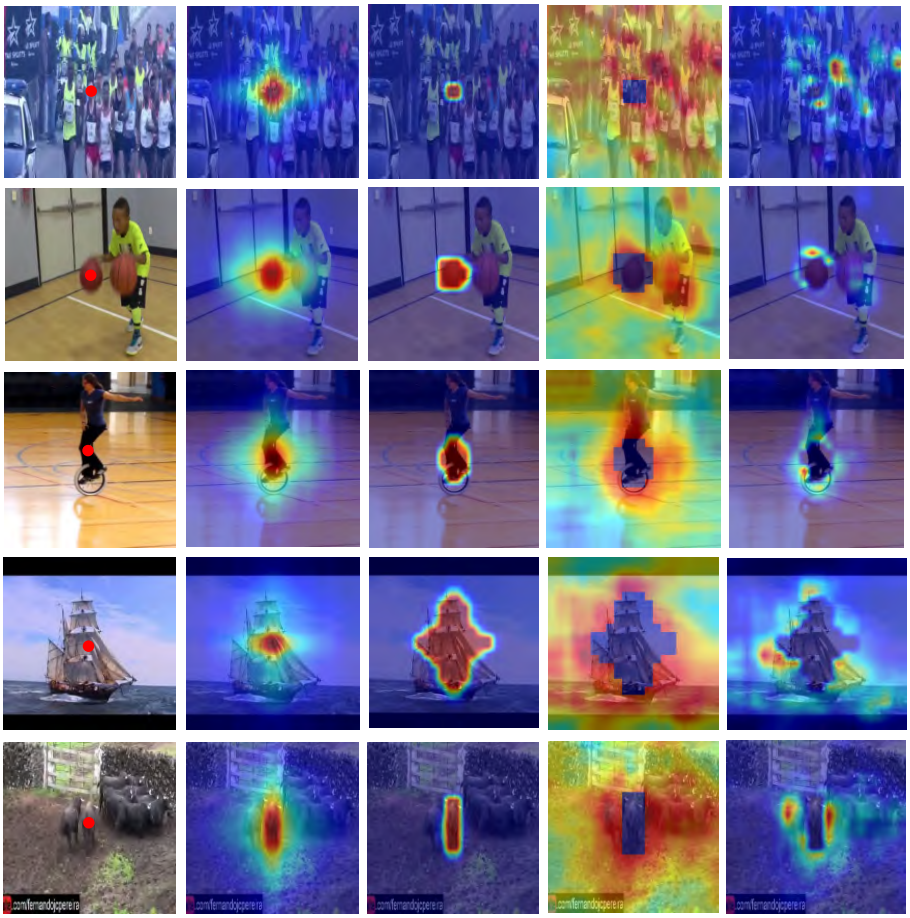

Figure 1. Qualitative visualization of global soft template (GST) and soft negative sample (SNS) generation in the proposed SoCL framework. (a) Input search images that contain both target and background regions. (b) Score maps generated via proposal generation and objectness measurement [1]. (c) TOP maps generated by applying the softmax function on the score maps. (d) the similarity map with target responses masked out using the background selection function. (e) applying the softmax to (d) yields the hard negative map, which is used to select features for the SNS.

Table 3. Ablation study on projection heads: AUCs obtained by using different trackers w/ and w/o projection heads on OTB-13. The best results are highlighted.

	w/ proj. head	w/o proj. head
SoCL-Siam	49.5	**60.9**
SoCL-CF	**69.6**	68.8

learn better feature representations for some typical downstream tasks, e.g., image classification. However, there is no empirical study to explore its usage on visual tracking, i.e., whether it is beneficial for learning robust tracking representations. Note that the projection head we use is implemented as a three-layer perceptrons with a single hidden layer of K units, where K is set to the dimension of features extracted from the backbone. The output of the final perceptron is a 64-dimensional vector.

We conduct this ablation study on two different types of tracking frameworks: offline learning-based Siamese and online learning-based correlation filter (CF) trackers. Specifically, we put the projection head after the backbone networks used in SoCL-Siam and SoCL-CF[1] to further extract features for contrastive learning.

The results are presented in Table 3. SoCL-Siam with the projection head degrades the performance, while SoCL-CF with the projection head achieves better performance than its variant without a projection head. The main reason is that SoCL-Siam directly uses the extracted backbone features for online tracking without further updating. The learning of SoCL-Siam without the projection head is consistent with its online tracking process, thus leading to bet-

[1]For the ResNet-18 backbone used in SoCL-CF, we remove its average pooling and fully-connected layers, and modify its stride in *Layer4* to 1, so that the final output in the feature space can have a relatively large spatial size, which is more beneficial for soft sample generation.

Table 4. AUCs obtained by using various η on OTB-13. The best results are highlighted.

η	5%	10%	15%	20%
SoCL-Siam	60.0	**60.9**	59.5	59.1

ter performance. Moreover, SoCL-Siam with the projection head treats the backbone network as the intermediate layers, which facilitates the backbone to learn to encode more detailed and rich information into features. These features are not good for offline learning-based trackers without further online updating. Compared with SoCL-Siam, SoCL-CF can benefit from these features due to its powerful online updating mechanism.

D. Preprocessing of TOP Map

The target objectness prior (TOP) maps are generated by applying the softmax function on the score maps, which are calculated via the generated proposals (see Sec. 3.1 of the main paper). In practical implementation, we find that the TOP maps may have extremely large peaks on the annotated locations, which makes the generation of GSTs excessively focus on these locations. This is because there are large peaks in the score maps, and the softmax function assigns too much weights on these locations. To alleviate this problem, we use a simple max clip operation to clip maximum values in score maps. Specifically, given a score map, we firstly set an adaptive clip threshold η. Then we calculate the mean score of the top-η scores in the score map. The calculated mean score is used to perform the max clip in the score map, so that the scores with large values will have the same value, and thus more locations will be selected by applying the softmax function.

Table 4 shows the performance obtained by using various η for the clip threshold. The optimal performance is achieved by setting $\eta = 10\%$. Setting η to larger values (e.g., 15% and 20%) may cause the generated TOP maps to excessively focus on background regions, thus degrading the performance.

E. Qualitative Visualization

Fig. 1 shows the qualitative visualization of GST and SNS generation. The generation of SNSs tends to aggregate features from discriminative regions, e.g., target boundary regions (see the third and fourth rows) and hard negative counterparts (see the first, second and fifth rows). Note that all the maps in Fig. 1 are interpolated to the input image size for visualization.

References

[1] B. Alexe, D. Thomas, and F. Vittorio. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2189–2202, 2012. 2

[2] T. Chen, S. Kornblith, and M. Norouzi. A simple framework for contrastive learning of visual representations. In *arXiv:2002.05709*, 2020. 1

[3] X. Chen, B. Yan, and J. Zhu. Transformer tracking. In *CVPR*, 2021. 1

[4] H. Fan, L. Lin, and F. Yang. Lasot: A high-quality benchmark for large-scale single object tracking. In *CVPR*, pages 5374–5383, 2019. 1

[5] L. Huang, X. Zhao, and K. Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 1

[6] M. Muller, A. Bibi, and Giancola S. Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In *ECCV*, pages 300–317, 2018. 1

[7] Y. Tian, S. Kornblity, and K. Swersky. Big self-supervised models are strong semi-supervised learners. In *Neurips*, 2020. 1