Supplementary Material for Contextualized Spatio-Temporal Contrastive Learning with Self-Supervision

1. Model Architectures

We illustrate the model architectures that are used in the ConST-CL framework.

1.1. Base network $f(\cdot)$

Table 1 describes the base model architecture that is proposed for reconciling global and local training signals.

Stage	Network	Input from	Output size $T \times S^2$
raw clip	-	-	32×224^2
data	stride **2**, 1^2	raw clip	16×224^2
res1	$\underline{\mathbf{5} \times 7^2}$, 64 stride **2**, 2^2	data	8×112^2
pool1	1×3^2 max stride 1, 2^2	res1	8×56^2
res2	$\begin{bmatrix} 1\times1^2, 64 \\ 1\times3^2, 64 \\ 1\times1^2, 256 \end{bmatrix} \times 3$	pool1	8×56^2
res3	$\begin{bmatrix} 1\times1^2, 128 \\ 1\times3^2, 128 \\ 1\times1^2, 512 \end{bmatrix} \times 4$	res2	8×28^2
res4	$\begin{bmatrix} \underline{3\times1^2, 256} \\ 1\times3^2, 256 \\ 1\times1^2, 1024 \end{bmatrix} \times 6$	res3	8×14^2
res5$_r$	$\begin{bmatrix} \underline{3\times1^2, 512} \\ 1\times3^2, 512 \\ 1\times1^2, 2048 \end{bmatrix} \times 3$	res4	8×7^2
res5$_g$	$\begin{bmatrix} \underline{3\times1^2, 512} \\ 1\times3^2, 512 \\ 1\times1^2, 2048 \end{bmatrix} \times 3$	res4	8×7^2

Table 1. **Base network $f(\cdot)$: a ResNet3D-50 (R3D-50) based encoder.**

1.2. ConST-CL head $g(\cdot, \cdot)$

Table 2 describes the projection head we use for achieving the instance prediction task.

2. Region Generation

In this section, we detail three options to generate region priors that we study for training ConST-CL.

Random boxes. For all of our related experiments, we randomly generate 8 boxes on each frame. The boxes are constrained to have aspect ratio within $[0.5, 2]$ and size within $[0.1, 0.5]$ of the image size.

Stage	Input, Dimension	Network	Output
Linear project	$\boldsymbol{h}$, N×2048	n_nodes=128	Query
Linear project	$\boldsymbol{F}_c'$, M×2048	n_nodes=128	Key
Linear project	$\boldsymbol{F}_c'$, M×2048	n_nodes=128	Value

MHSA	Query, N×128
	Key, M×128
	Value, M×128
		hidden_size=128
		n_heads=3
		n_layers=3
			Hidden

Linear project	Hidden, N×128	n_nodes=2048	$\boldsymbol{z}$

Table 2. **ConST-CL head $g(\cdot, \cdot)$: a transformer-based decoder.** The inputs are the region features $\boldsymbol{h}$ and the context features $\boldsymbol{F}_c'$ and the outputs are the transformed features $\boldsymbol{z}$. N and M are the number of tokens of $\boldsymbol{h}$ and $\boldsymbol{F}_c'$ respectively.

Boxes from low-level image cues. We use the SLIC [1] algorithm to generate 16 superpixels on each frame. Following [10], we alternatively use the graph-based image segmentation method [6] to generate 16 image segments for each frame. We use two scales to generate segments, the scale s and minimum cluster size c, and $s = c \in \{500, 1000\}$ in practice. After the segments generation, we convert each segment into its minimal bounding box and only keep those with width/height between $[0.05, 0.7]$ of the image width/height.

Boxes from detectors. We also use off-the-shelf modern detectors to generate object-centric bounding boxes for weakly supervised learning. A CenterNet-based [20] person detector is employed to generate bounding boxes on persons only. As an alternative, we use a generic object detector, which is based on Cascade RCNN [3].

3. Downstream tasks

3.1. Action Recognition

On all video action recognition datasets, we use the video clip of 32 frames with temporal stride 2 as input. During training, the temporally consistent random data augmentation [15] of cropping, resizing and flipping are applied and the resolution is set to 224 × 224. During evaluation, we densely sample 10 clips with resolution 256 × 256 from each video and apply a 3-crop evaluation following [4].

Linear Evaluation. On action recognition datasets, we train a linear classifier with fixed backbone weights using the SGD optimizer with momentum of 0.9. On Kineitcs400 [12], the linear classifier is trained for 100 epochs with learning rate of 32 and batch size of 1024. On UCF101 [16] and HMDB51 [13], the linear classifier is trained for 50 epochs with learning rate of 0.84 and batch size of 128. No dropout and weight decay are applied.

Fine-tuning. On UCF101 [16] and HMDB51 [13], we use the pre-trained models to initialize the network and fine-tune all layers for 50 epochs. We use batch size of 128, weight decay of 1e-5 and dropout rate of 0.5 during fine-tuning. The learning rate is set to 0.72 and 0.84 for UCF101 and HMDB51 respectively.

3.2. Spatio-temporal Action Localization

We use the same action transformer head as in [7, 14] to our R3D-50 backbone and follow the setting in [14]. The model is fine-tuned with batch size 256 for 50k steps, which is around 36 epochs on AVA-Kinetics [14]. The input has 32 frames with resolution 400 and temporal stride 2. We use the SGD optimizer with momentum 0.9 during the fine-tuning. On AVA-Kinetics, the learning rate is 1e-2 and the weight decay is 1e-7. On AVA [9], the learning rate is set to 3e-2 and the weight decay is 1e-4. During evaluation, we use the same set of detected boxes in [14] for AVA-Kinetics and in [5] for AVA v2.2 for a fair comparison.

3.3. Object Tracking

To evaluate on OTB2015 [17] dataset, we follow the same practice as in [8, 18, 19] to adopt the SiameseFC [2] as the tracker. Specifically, we modify the spatial stride and dilation rate to be $(1, 2)$ and $(1, 4)$ in the first layer of the res_4 and res_5 blocks. These modifications allows us to increase the feature map resolution without impacting on the pre-trained model. We fine-tune the tracker on GOT-10K [11] dataset using the SGD optimizer with momentum of 0.9. We use batch size of 256, learning rate of 0.1 and weight decay of 1e-4 and the tracker is fine-tuned for 20 epochs.

4. Visualization

4.1. Attention

We visualize the learned attention map during the training in Figure 1. For visualization purpose only, we use the boxes from the object detector to pool the region features in the source views to generate the attention maps. The model is trained with the randomly generated boxes as described in the paper. In Figure 1, we visualize the center frames in the source and the target views and the source frames are superimposed with one box for visualization. The zoomed-in thumbnails are presented in the second column. Given the context (features from the target views), we use these thumbnails' region feature as the query to generate the attention maps shown in the fourth column. It is interesting to observe that the model learns to attend to not only the corresponding instance in the target view, but also to some other semantically meaningful objects the instance potentially interacts with.

4.2. Visual Object Tracking

We provide some qualitative results on visual object tracking on OTB2015 [17] in Figure 2. The results show that our tracker could robustly track objects under different scenarios.

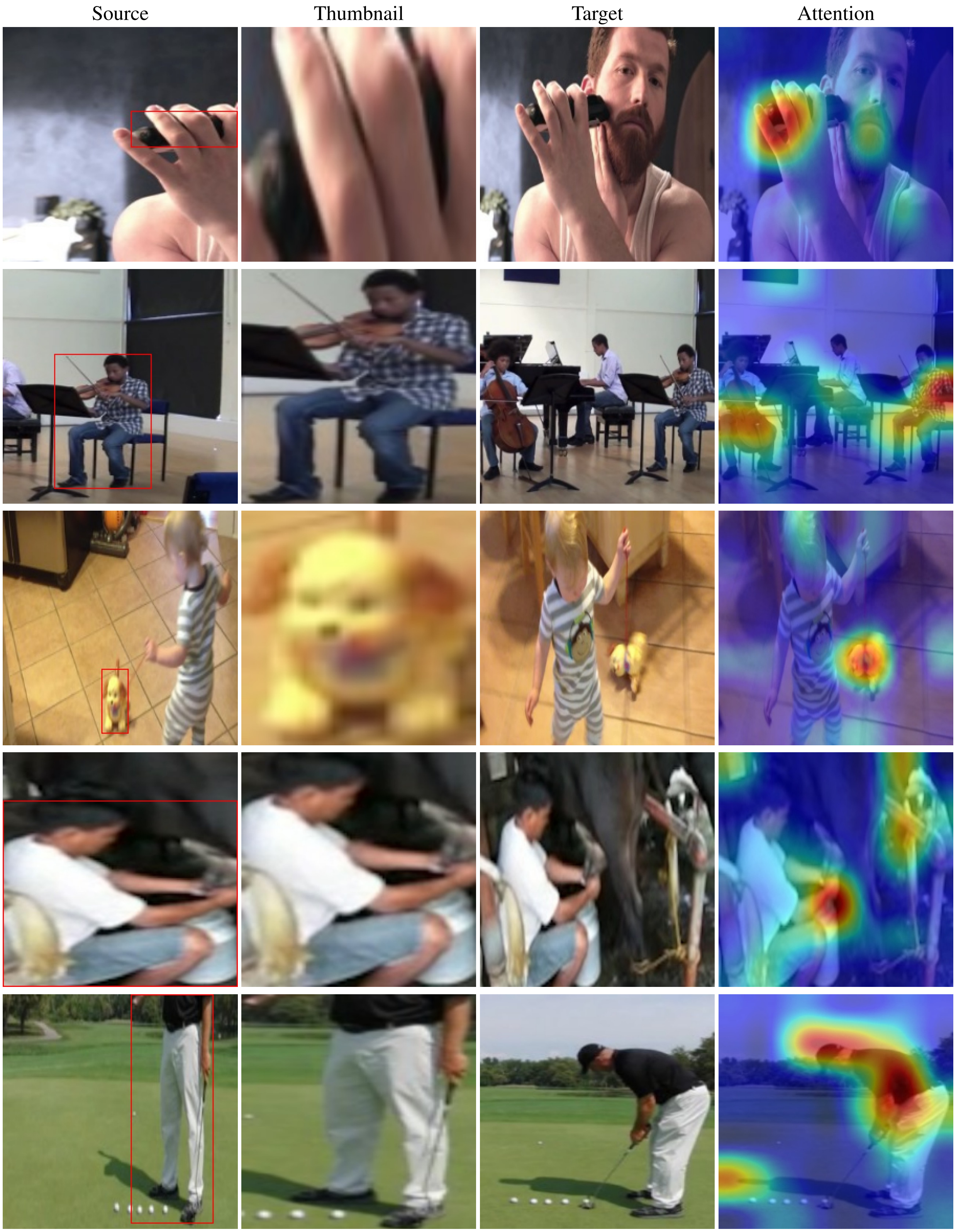

Figure 1. **Visualization of the attention during the training.** We use boxes from the object detector to pool region features from the source view as the query in order to generate the attention maps given the context (features from the target view). Interestingly, the model learns to attend to not only the corresponding instance in the target frame, but also to some other semantically meaningful objects the instance potentially interacts with.

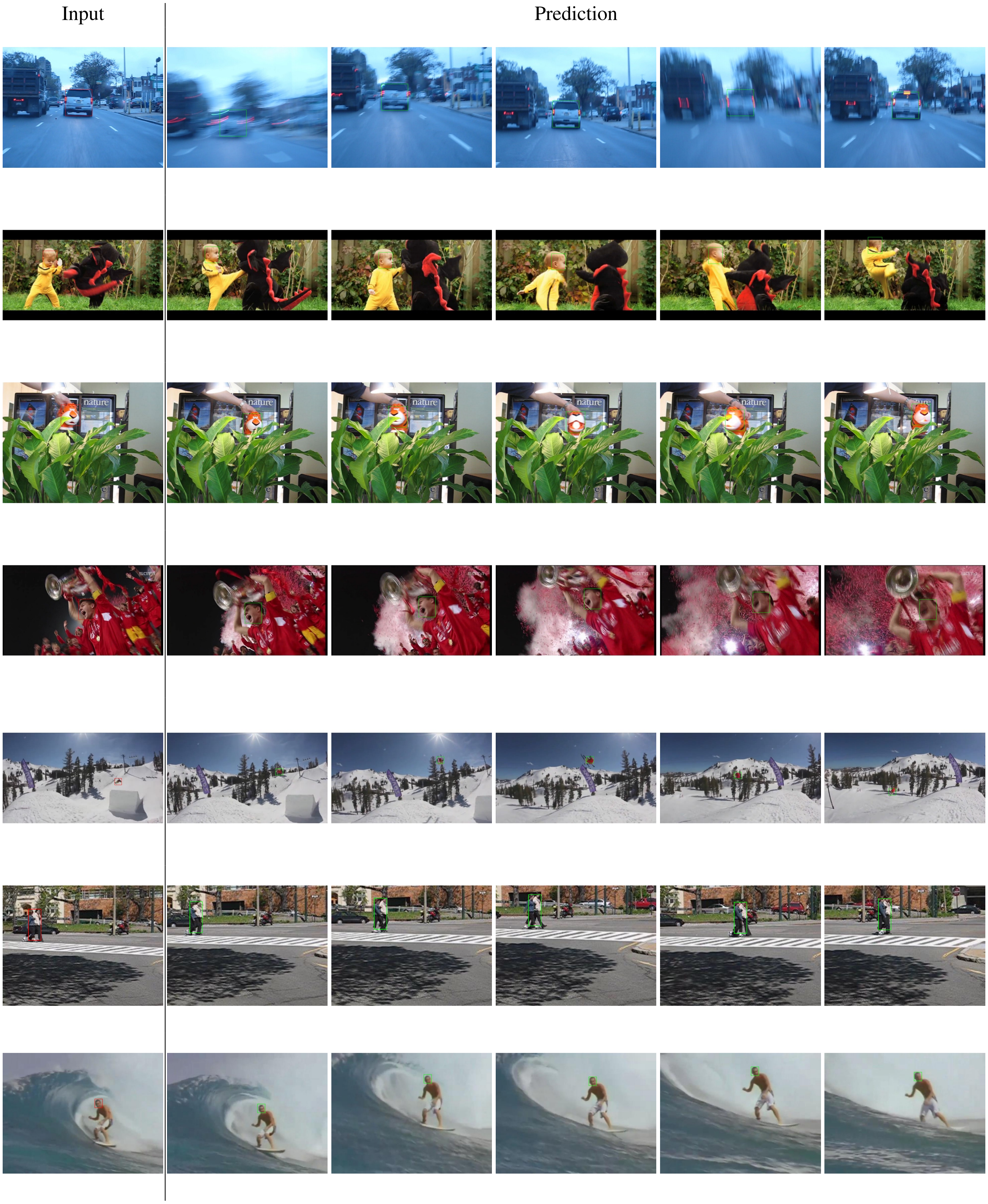

Figure 2. Qualitative results for visual object tracking on OTB2015 [17]. Best view in color.

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *TPAMI*, 2012. 1

[2] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *ECCV*, 2016. 2

[3] Xianzhi Du, Barret Zoph, Wei-Chih Hung, and Tsung-Yi Lin. Simple training strategies and model scaling for object detection. *arXiv preprint arXiv:2107.00057*, 2021. 1

[4] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019. 2

[5] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *CVPR*, 2021. 2

[6] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004. 1

[7] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *CVPR*, 2019. 2

[8] Daniel Gordon, Kiana Ehsani, Dieter Fox, and Ali Farhadi. Watching the world go by: Representation learning from unlabeled videos. *arXiv preprint arXiv:2003.07990*, 2020. 2

[9] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *CVPR*, 2018. 2

[10] Olivier J Hénaff, Skanda Koppula, Jean-Baptiste Alayrac, Aaron van den Oord, Oriol Vinyals, and João Carreira. Efficient visual pretraining with contrastive detection. In *ICCV*, 2021. 1

[11] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *TPAMI*, 2021. 2

[12] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2

[13] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, 2011. 2

[14] Ang Li, Meghana Thotakuri, David A Ross, João Carreira, Alexander Vostrikov, and Andrew Zisserman. The ava-kinetics localized human actions video dataset. *arXiv preprint arXiv:2005.00214*, 2020. 2

[15] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *CVPR*, 2021. 1

[16] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 2

[17] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online object tracking: A benchmark. In *CVPR*, 2013. 2, 4

[18] Jiarui Xu and Xiaolong Wang. Rethinking self-supervised correspondence learning: A video frame-level similarity perspective. In *ICCV*, 2021. 2

[19] Ting Yao, Yiheng Zhang, Zhaofan Qiu, Yingwei Pan, and Tao Mei. Seco: Exploring sequence supervision for unsupervised representation learning. In *AAAI*, 2021. 2

[20] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 1