Appendix for "TransRank: Self-supervised Video Representation Learning via Ranking-based Transformation Recognition"

Haodong Duan[1] Nanxuan Zhao[1,3,4]✉ Kai Chen[2,5] Dahua Lin[1,2,3]
[1]The Chinese University of HongKong [2]Shanghai AI Laboratory
[3]Centre of Perceptual and Interactive Intelligence [4]University of Bath [5]SenseTime Research

Table 10. **The mapping from HMDB51 action classes to five motion types.** The mapping is manually annotated by [**?**].

Linear	Projectile	Local	Oscillatory	Random
brush hair	cartwheel	chew	clap	draw sword
climb	catch	drink	dribble	fall floor
climb stairs	dive	eat	pullup	fencing
punch	flic flac	kiss	pushup	hug
push	golf	laugh	situp	kick
ride bike	handstand	pour		kick ball
ride horse	hit	shake hands		pick
run	jump	shoot gun		sit
shoot bow	shoot ball	smile		stand
walk	somersault	smoke		sword
	swing baseball	talk		sword exercise
	throw	wave		turn

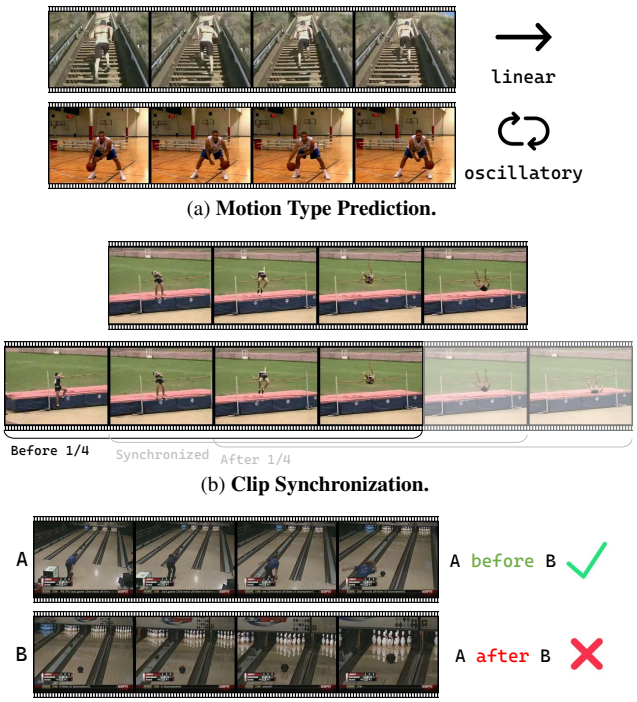
A. Details about Downstream Tasks

In this section, we first describe the three temporal-related downstream tasks in detail. Then we illustrate the concrete practices to transfer the SSL models to each task. Figure 6 illustrates the three temporal-related tasks.

A.1. Temporal-related Tasks

Motion Type Prediction (*Motion*). The task is first proposed in [**?**], aiming at categorizing the motion in a video into five pre-defined categories: linear, projectile, local, oscillatory, and random. To obtain the ground-truth, each class in HMDB51 [**?**] is annotated with one pre-defined motion type, yielding a new dataset mHMDB51 [**?**]. We list the mapping in Table 10. To transfer SSL models to this task, we initialize the backbone of a 5-way classifier with pre-trained weights and finetune on the mHMDB51. We report the Top1 accuracy on the test set (with 1530 videos).

Synchronization (*Sync*). The synchronization task is first proposed in RTT [**?**], serving as a pseudo task to illustrate the motion modeling capability of SSL models. For this task, two temporally overlapping clips a, b are sampled from a video and separately fed to the pre-trained backbone ψ to extract features $\psi(a)$, $\psi(b)$ at the res_4 layer. We consider 7 overlapping patterns $\{-3/4, -1/2, ..., +3/4\}$ (*e.g.*, $-1/2$ denotes a is ahead of b with $1/2$ overlapping, $+3/4$ denotes a is behind of b with $3/4$ overlapping). Following [**?**], we adopt an MLP on top of the fused feature $\psi(a) - \psi(b)$ to recognize the overlapping pattern. The backbone parameters are fixed during training.



(a) **Motion Type Prediction.**

(b) **Clip Synchronization.**

(c) **Temporal Order Prediction.**

Figure 6. **Three temporal-related downstream tasks.**

Temporal Order Prediction (*Order*). The order task is also proposed in RTT [**?**] to evaluate the temporal modeling capability of SSL methods. For this task, a single input clip x (with 16 frames) is constructed by sampling two non-overlapping sub-clips (each has 8 frames) x_1, x_2, where x_1 comes before x_2. The network inputs are then either (x_1, x_2) for class "before" or (x_2, x_1) for class "after". Following [**?**], we train an MLP on top of the res_4 feature

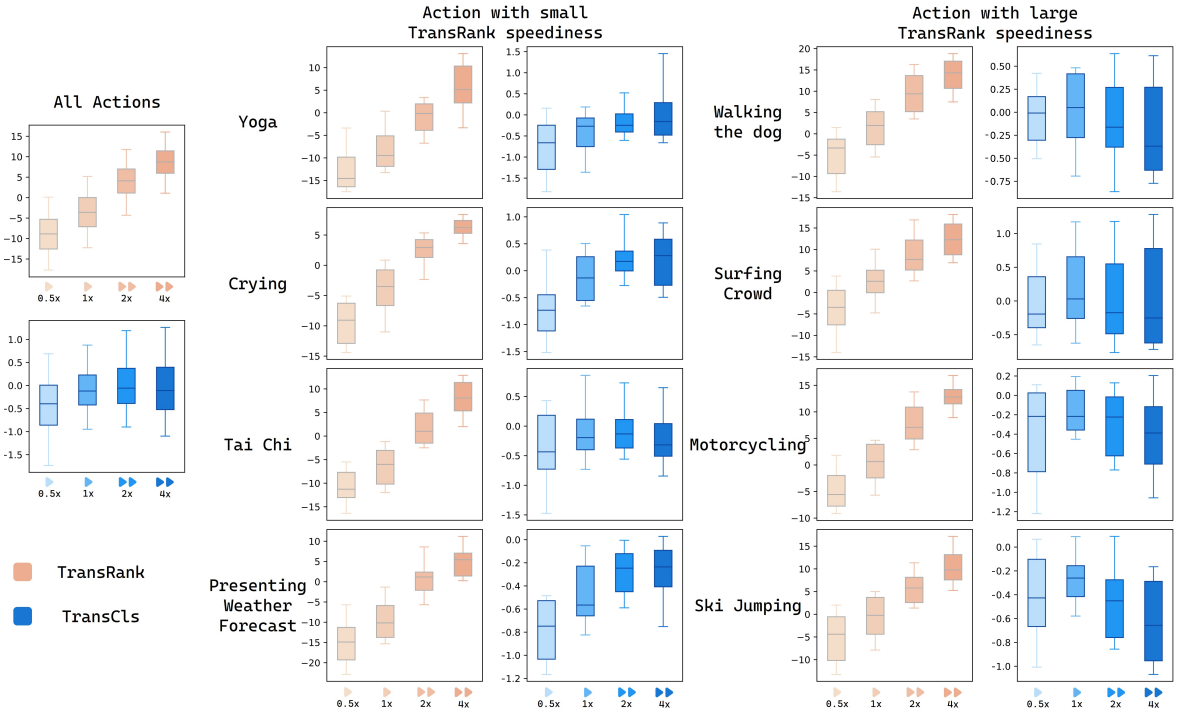


Figure 7. **Class-specific visualization for speediness score (unnormalized) distributions.** TransRank can accurately capture the relative speediness of different playback rates, while TransCls cannot. Besides, the absolute TransRank speediness score can reveal the speediness of an action category to some extent.

of x. The backbone parameters are fixed during training.

A.2. Practices for Transfer Learning.

In this section, we introduce the transfer learning practices we adopted for all pretext tasks in the preliminary study (main paper Sec 3). For all downstream experiments, we use SGD as the optimizer. The hyper-parameter configuration is listed in Table 11. We adopt five initial learning rates and report the best results.

Semantic-related Tasks. For *Nearest Neighbor Evaluation,* we extract and average features (res_4) of 10 clips to obtain a single 512-d feature vector for each video. Cosine similarity is used as the metric to determine the nearest neighbors. For each clip in the testing split, we query clips in the training split to get N nearest neighbors (N = 1, 5, 10) and report the corresponding recalls. For *Linear Evaluation* and *Finetuning*, we adopt a fully-connected head on top of the backbone and re-train it for action recognition. We fix all backbone parameters for *Linear Evaluation*.

Temporal-related Tasks. We adopt the SSL weights to initialize the backbone and finetune all parameters for *Motion*. For *Sync* and *Order*, we fix all backbone parameters and re-train the MLP heads. We adopt a 2-layer MLP with a hidden layer 512.

Table 11. **The hyper parameters for transfering.**

Hyper Parameter	Value (Choices)
Batch Size	128
Total Epochs	100
Initial Learning Rate	$\{0.01, 0.02, 0.04, 0.08, 0.16\}$
Learning Rate Decay	0.1, decays at the 60_{th}, 80_{th} epoch
Momentum	0.9
Weight Decay	10^{-4}
Dropout Ratio	0.5

B. Per-Class Speediness Distribution

In Figure 7, we visualize the distributions of speediness scores (unnormalized) for different action categories in MiniKinetics. For comparison, we first visualize the unnormalized speediness distribution (TransRank and TransCls) for all MiniKinetics validation videos on the left side. Figure 7 shows that for each single action category, For each action, TransRank can accurately capture the relative speediness of different playback rates, while TransCls cannot. Besides, the absolute TransRank speediness score can reveal some characteristics of the action. The $1\times$ clips with lower TransRank speediness scores belong to the 'still' actions, like Yoga, Crying, Tai Chi; while for actions like Motorcycling and Ski Jumping, the TransRank speediness scores of $1\times$ clips are much larger.

Table 12. **Additional Results for Video Retrieval on the split 1 of UCF101 and HMDB51.** For CoCLR, we report numbers obtained with the released checkpoint and codebase (https://github.com/TengdaHan/CoCLR). For dual-modality video retrieval, we average the similarity of both modalities to obtain the new similarity matrix. TransRank-ST achieves impressive retrieval performance with both RGB and the cheap RGBDiff modalities. With two modalities combined, TransRank-ST outperforms the previous state-of-the-art CoCLR, which adopts the much more expensive modality optical flow.

Method	Backbone	Modality	Pre-train Data	UCF101					HMDB51				
				R@1	R@5	R@10	R@20	R@50	R@1	R@5	R@10	R@20	R@50
CoCLR	S3D	RGB	UCF101	53.2	69.2	76.5	82.2	88.8	21.9	42.1	54.4	68.0	83.9
CoCLR	S3D	Flow	UCF101	49.2	68.2	75.7	82.0	88.4	22.2	46.1	56.1	69.3	84.1
CoCLR	S3D	RGB + Flow	UCF101	54.5	71.2	76.8	82.6	89.0	23.6	46.6	57.9	70.1	85.0
CoCLR	S3D	RGB	K400	46.3	62.8	69.5	76.7	84.5	20.6	43.0	54.0	66.3	81.2
CoCLR	S3D	Flow	K400	23.7	46.1	58.1	70.1	82.8	11.4	33.1	48.0	64.3	84.3
CoCLR	S3D	RGB + Flow	K400	40.3	60.6	69.5	77.6	86.9	19.3	43.6	54.3	68.9	84.8
TransRank-ST	R3D-18	RGB	UCF101	46.5	63.7	72.8	82.0	90.0	19.4	45.4	59.1	74.0	86.9
TransRank-ST	R3D-18	RGBDiff	UCF101	43.7	62.7	72.8	82.2	91.2	17.6	41.2	56.4	71.3	87.1
TransRank-ST	R3D-18	RGB + Diff	UCF101	48.1	66.2	75.0	83.0	91.5	19.7	47.2	60.1	74.0	86.6
TransRank-ST	R3D-18	RGB	K200	54.0	71.8	79.6	86.4	92.5	25.5	**52.3**	65.8	**78.4**	89.6
TransRank-ST	R3D-18	RGBDiff	K200	52.9	72.7	81.6	87.6	93.6	21.8	50.0	62.8	75.4	**90.9**
TransRank-ST	R3D-18	RGB + Diff	K200	**56.7**	**74.2**	**82.1**	**88.3**	**93.9**	**27.3**	52.1	**66.9**	77.7	90.6

C. Experiments

C.1. Training Details.

Pre-training Details. In pre-training, we sample N clips from each video, applying different temporal transformations ($1\times$, $2\times$, rev, *etc.*), spatial transformations (like RandomRotate, only for TransRank-ST), and strong clip-wise spatial & temporal augmentations to each clip. With transformations applied, each input clip consists of 16 frames with spatial size 112×112. We use SGD as the optimizer with a mini-batch size 64. We adopt the CosineAnnealing scheduler to update the learning rate (lr), while the initial lr is set to 0.1. We train the model for 100 epochs by default.

Finetuning Details. We finetune TransRank on UCF101, HMDB51, and SthV1 for action recognition. We finetune 100 epochs on UCF101, HMDB51; 50 epochs on SthV1 (the dataset is much larger). SGD is used as the optimizer with the MultiStepLR scheduler (decay the learning rate by $1/10$ after $3/5$ and $4/5$ training epochs finished). By default, we use 0.16 as the finetuning LR and set the batch size to 128. We find that large finetuning LR is critical for the success of RecogTrans-based SSL approaches.

C.2. Multi-modality Retrieval Results

In Table 12, we report multi-modality video retrieval results. Besides RGB, TransRank-ST can also achieve impressive retrieval performance with the RGBDiff modality. Moreover, we find that RGB and RGBDiff are two complementary modalities. The ensemble can outperform each individual modality. Integrated with the cheap modality RGBDiff, TransRank-ST outperforms the contrastive-based approach CoCLR [**?**] trained with the much more expensive modality optical flow. Besides, we find a clear trend for TransRank-ST: more pre-training data $\rightarrow$ better retrieval performance. However, this trend doesn't hold true for the contrastive-based CoCLR.

C.3. Qualitative Results for Video Retrieval

Figure 8 visualizes a query video clip and its Top1 retrievals obtained with both TransRank-ST and CoCLR (both use RGB modality). We find that compared to CoCLR, TransRank-ST focuses less on static cues and more on human motions. With TransRank-ST features, one can obtain high-quality retrieval results, robust to changes in background scene or illumination.

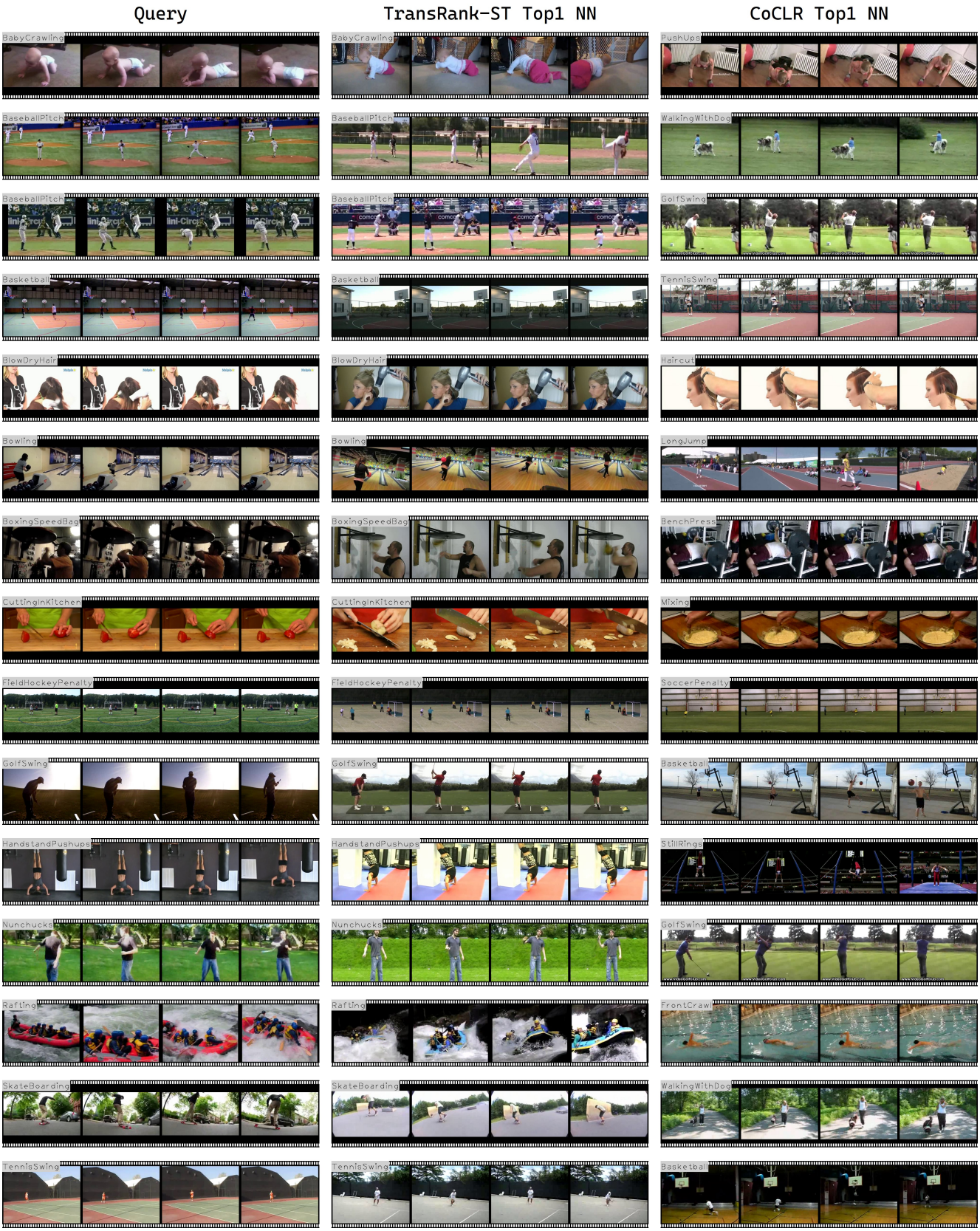


Figure 8. **Qualitative results for video retrieval.** The representation learned by TransRank-ST can retrieve videos with the same action categories. It focuses on human motion and is less vulnerable to changes in background scene or illumination.