A Large-Scale Analysis on Contextual Self-Supervised Video Representation Learning

Supplementary Material

Here, we explain things in details about pretext task, architecture setup, provide some more results and include more visual analysis. We also include tables which we were not able to include in main paper due to space limitations.

- Section 7: describes challenges and future work based on our study.
- Section 8: Pretext tasks explanation used in our analysis.
- Section 9: Training details about architectures, datasets, and, other hyperparameters.
- Section 10: We show additional CKA maps, more results on HMDB51 dataset and more analysis on noise robustness. We added some tables for Knowledge distillation experiments that were promised in the main paper.
- Section 11: We extend the main table and compare with previous state-of-the-art results on HMDB51 dataset.

7. Challenges and future work

There are several key challenges in video SSL and we believe 1) long-term video understanding, 2) multi-modal learning, and 3) robust learning are some of the less studied aspects. The novel insights in our study regarding training dataset size, model architectures, and robustness will play a crucial role in guiding future work on these research directions.

8. Pretext Tasks Details

In this section, we go through each pretext task in more detail that are used in our main work for analysis.

8.1. Spatial Transformation

Rotation Net [34] (RotNet) applies geometrical transformation on the clips. The videos are rotated by various angles and the network predicts the class which it belongs to. Since the clips are rotated, it helps the network to not converge to a trivial solution.

Contrastive Video Representation Learning [56] (CVRL) technique applies temporally coherent strong spatial augmentations to the input video. The contrastive framework brings closer the clips from same video and repels the clip from another video. With no labels attached, the network learns to cluster the videos of same class but with different visual content.

8.2. Temporal Transformation

Video Clip Order Prediction [85] (VCOP) learns the representation by predicting the permutation order. The network is fed N clips from a video and then it predicts the order from N! possible permutations.

Temporal Discriminative Learning [78] (TDL) In contrast to CVRL, TDL works on temporal triplets. It looks into the temporal dimension of a video and targets them as unique instances. The anchor and positive belongs to same temporal interval and has a high degree of resemblance in visual content compared to the negative.

8.3. Spatio-Temporal Transformation

Playback Rate Prediction [88] (PRP) has two branch, generative and discriminative. Discriminative focuses on the classifying the clip's sampling rate, whereas, generative reconstructs the missing frame due to dilated sampling. Thus, the first one concentrates on temporal aspect and second one on spatial aspect.

Relative Speed Perception Network [10] (RSPNet) applies contrastive loss in both spatial and temporal domain. Clips are samples from a same video to analyze the relative speed between them. A triplet loss pulls the clips with same speed together and pushes clips with different speed apart in the embedding space. To learn spatial features, InfoNCE loss [76] is applied. Clip from same video are positives whereas clips from different videos are negatives.

Video MAE [73] (V-MAE) applies a spatio-temporal tube masking to the input video. The pretext task is to reconstruct those missing tubes. Mean-squared error loss is applied between the masked tokens and the reconstructed tokens.

9. Implementation Details

9.1. Architecture Details

Preliminary research has shown that 3D networks [27, 75] have outperformed 2D CNN variants on video recognition tasks. We looked into three types of capacity - small, medium and big on the basis of number of trainable parameters. The architecture details of all networks are mentioned in supplementary.

Small capacity networks: are resource efficient, implying they can be trained in larger batches within short span

of time. The network selection is done on the basis of supervised training scores on Kinetics[35] and UCF101[38]. ShuffleNet V1 2.0X [89] utilizes point-wise group convolutions and channel shuffling. SqueezeNet [31] reduces the filter size and input channels to reduce the number of parameters. MobileNet [61] has ResNet like architecture. With its depthwise convolution, there's a reduction in model size and the network can go more deep.

Medium capacity networks: Following the conventional 3D architectures for self-supervised learning approaches C3D, R21D and R3D are used in this study.

Big Capacity networks: Comparing across four transformer architectures, ViViT [5] Timesformer [8], VideoSwin [47] and MViT [18], we selected VideoSwin, because it outperforms others on Kinetics 400 dataset.

Based on [38], we probed into the performance comparison of several versions of these architectures. We choose 3D-ShuffleNet V1 2.0X, 3D-SqueezeNet, and 3D-MobileNet V2 1.0X networks based on their performance on Kinetics and UCF-101 dataset

3D-ShuffleNet V1 2.0X [89]: It utilize point-wise group convolutions and channel shuffling and has 3 different stages. Within a stage, the number of output channel remains same. As we proceed to successive stage, the spatiotemporal dimension is reduced by a factor of 2 and the number of channels are increased by a factor of 2. V1 denotes version 1 of ShuffleNet and 2.0X denotes the 2 times number of channels compared to original configuration.

3D-SqueezeNet [31]: It uses different alteration to reduce the number of parameters as compared to the 2D version which employs depthwise convolution. Those three modifications are: 1) Change the shape of filters from 3x3 to 1x1, 2) Input channels to 3x3 filters is reduced, and, 3) to maintain large activation maps high resolution is maintained till deep layers.

3D-MobileNet V2 1.0X [61]: This network employs skip connections like ResNet architecture in contrast to version 1. It helps the model in faster training and to build deeper networks. There are also linear bottlenecks present in the middle of layers. It helps in two ways as we reduce the number of input channels: 1) With depthwise convolution, the model size is reduced, and 2) at inference time, memory usage is low. V2 denotes version 2 of mobilenet and 1.0X uses the original parameter settings.

The architectures of medium capacity networks are described as follows:

C3D [74]: This follows a simple architecture where two dimensional kernels have been extended to three dimensions. This was outlined to capture spatiotemporal features from videos. It has 8 convolutional layers, 5 pooling layers and 2 fully connected layers.

R3D [27]: The 2D CNN version of ResNet architecture is recasted into 3D CNNs. It has skip connections that helps make the gradient flow better as we build more deeper networks.

R(2+1)D [75]: In this architecture, 3D convolution is broken down into 2D and 1D convolution. 2D convolution is in spatial dimension and 1D convolution is along the temporal dimension. There are two benefits of this decomposition: 1) Increase in non-linearity as the number of layers have increased, and, 2) Due to factorization of 3D kernels, the optimization becomes easier.

VideoSwin [47] It is an inflated version of original Swin [46] transformer architecture. The attention is now spatio-temporal compared to previous which is only spatial. 3D tokens are constructed from the input using patch partition and sent to the network. The architecture includes four stages of transformer block and patch merging layers.

9.2. Original and Noise Datasets

We have shown the examples of each dataset used in the paper in Fig. 6.

The test datasets have different number of videos for different levels and types of noises. For Gaussian noise, we manipulated all 3783 samples. For noise level 1, apart from Gaussian, we had roughly 400 samples and all other levels of severity, we have approximately 550 samples. An example of each type of noise is shown in Fig. 7.

9.3. Pretext Tasks Configurations

Here, we briefly describe the configurations used in our training. For VCOP, RotNet and PRP, we just manipulated the type of augmentation from the original work. We applied Random Rotation, Resizing, Random Crop, Color Jittering and Random Horizontal Flipping to the input clip. CVRL has some extra data augmentation compare to the previous ones we mentioned. It includes grayscale and gamma adjustment as well. RSPNet also uses some temporal augmentation. For finetuning the augmentations are Resize and Center Crop for all the approaches.

The k-value for Momentum contrastive network is 16384 for RSPNet, it's 500 for TDL.

9.4. Datasets

Here we discuss datasets in detail. We use Kinetics-400 (K400) [35] and Something-Something V2 [24] for our pretraining. For the downstream task evaluation, we perform our experiments on UCF-101 [67], HMDB-51 [40], and Diving48 [45]. Since, the pretraining and finetuning datasets are different, the performance variation will provide us a better picture about how much meaningful spatiotemporal features are learned by these networks. K400 has approximately 240k videos distributed evenly across 400 classes respectively. The approximate number of videos in finetuning datasets are: 1) UCF101-10k, 2) HMDB51-7k, and,

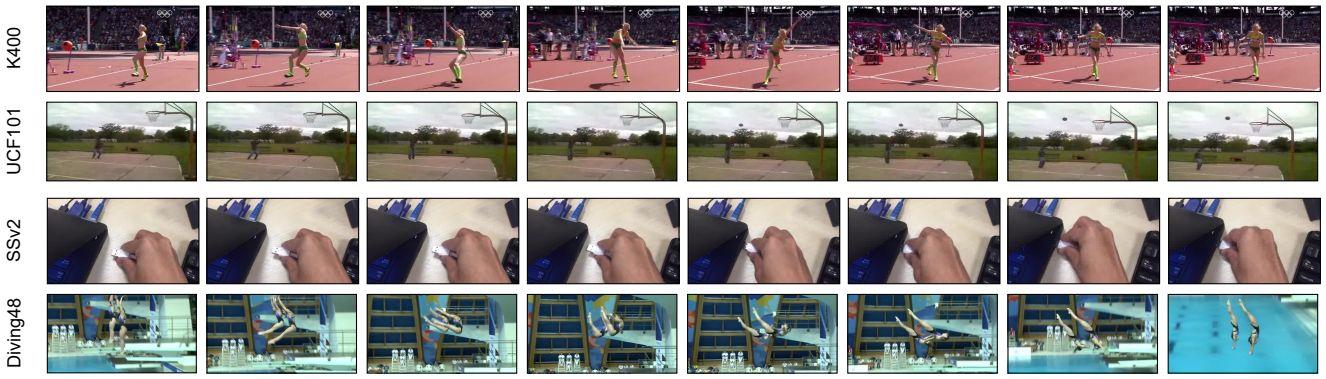

Figure 6. **Example** sample from each dataset.

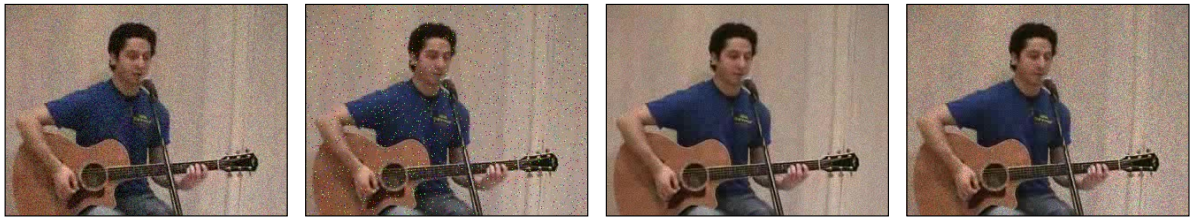

Figure 7. **Example frame sample for each noise** Gaussian, Impulse, Shot and Speckle from left to right. Sample clips are provided in supplementary.

3)Diving48-16k. The datasets can be categorized into two ways:

Appearance-based: Kinetics, UCF101 and HMDB51 comes under this category [12, 29]. Kinetics videos length are generally 10s centered on human actions. It mainly constitutes singular person action, person-to-person actions and person-object action. For pre-training, we select a random subset of videos and maintain equal distribution from each class. Unless otherwise stated, pre-training is done on K400-50k subset for all experiments.

Temporal-based: In Kinetics, we can estimate the action by looking at a single frame [12, 29]. From Fig. 6, top two rows, we can see the person with a javelin and basketball. This information helps in class prediction. Looking at bottom two rows (SSv2 and Diving48 respectively), we can't describe the activity class until we look into few continuous frames. It shows that temporal aspect plays an important role for these datasets, that's why we categorize them into temporal-based datasets.

UCF-101 [67] : It's an action recognition dataset that spans over 101 classes. There are around 13,300 videos, with 100+ videos per class. The length of videos in this dataset varies from 4 to 10 seconds. It covers five types of categories: human-object interaction, human-human interaction, playing musical instruments, body motion and sports.

HMDB-51 [40] : The number of videos in this dataset is 7000 comprising 51 classes. For each action, at least 70 videos are for training and 30 videos are for testing. The actions are clubbed into five categories: 1) General facial actions, 2) Facial actions with body movements, 3) General body movements, 4) Body movements with object interaction, and, 5) Body movements for human interaction.

10. Additional Results

Here, we will talk about some additional results, to further strengthen the claims made in the main paper.

10.1. Preliminary Experiments

Pretext tasks evaluation Figure 8 depicts the hidden representations of R21D network pretrained on different pretext tasks. Here the 50k subset of K-400 was used for pretraining, and finetuned on UCF-101.

Linear Probing vs Finetuning Firstly, we discuss linear probing (LP) vs finetuning (FT) results for different pretext tasks and different architectures. From Table 9, we can see that FT outperforms LP by a margin of approximately 20% and 40% on ShuffleNet and R21D respectively. Thus, we perform finetuning for all of our analysis.

Network Parameters We have shown the performance across different architectures in Table 10. ShuffleNet and R21D performs the best across small and medium capacity

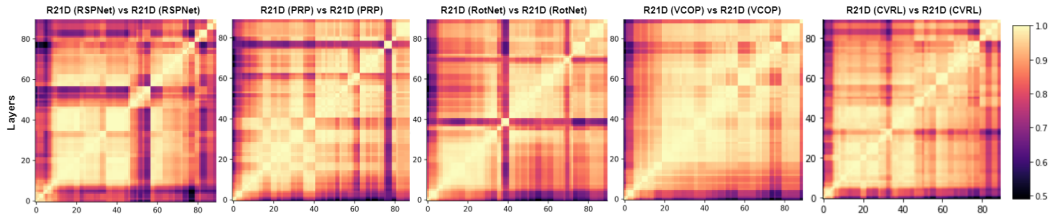

Figure 8. **Pretext tasks CKA maps** for RSPNet, PRP, RotNet, VCOP, CVRL on K-400 50k subset using R21D network (Left to right). R21D pretrained on K400 shows a semi-block structure for VCOP, indicating near-saturation condition of the network on this pretext task. It shows a more prominent grid-based structure on CVRL and RSPNet instead. These observations corroborate the quantitative results, where pretraining on K400 for both CVRL and RSPNet gives better performance.

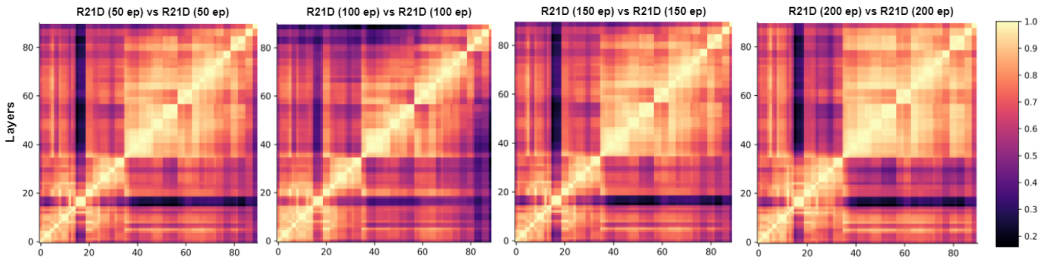

Figure 9. **Training time CKA maps** on 50, 100, 150, 200 epochs of R21D network on RSPNet pretext for K-400 10k subset (Left to right). The block structure is visible from 50 epochs itself, which then darkens and becomes prominent by 200 epochs. With 10k subset, the saturation starts hitting at 100 epochs.

	Non-contrastive			Contrastive		
Epochs	VCOP	Rot	PRP	CVRL	TDL	RSPNet
10k	18.9	15.0	9.2	22.2	9.9	30.2
30k	19.3	11.7	11.5	25.0	10.1	37.3
50k	17.3	12.2	10.2	29.3	9.5	40.2

Table 8. **Evaluation of different pretext tasks** on different subset size on R21D network on HMDB51 dataset.

Network	LP	FT	RotNet	VCOP	PRP
Shuffle	✓		4.3	12.3	2.8
		✓	16.6	40.8	21.9
R21D	✓		2.7	12.2	4.6
		✓	41.2	51.5	46.2

Table 9. **Downstream accuracy** classification on UCF-101 dataset. FT: Finetuning LP: Linear Probing

networks in most of the pretext tasks. Thus, we choose ShuffleNet and R21D for our benchmark analysis.

10.2. Effect of dataset size

In Table 8, we extend results for different pretext tasks on HMDB51 dataset. Similar to UCF101, *the scale in subset size doesn't reciprocate to gain in performance* for all pretext tasks on HMDB51 dataset. From Figures 10 and 11, we see that performance increase for Swin by a good margin, whereas in case of ShuffleNet and R21D it's relatively less beyond 50k subset.

Training time Table 11 shows VideoSwin saturates at 150 epochs on UCF101 whereas CNN architectures saturates earlier (100 epochs) which reflects limitation of model capacity. Figure 9 shows the emergence of block structures for R21D network trained on RSPNet for K400 10k. The saturation point has reached earlier around 100 epochs which supports the hypothesis in main work that CNN architectures mostly saturates around 100 epochs. We see similar pattern even after increasing the dataset size.

10.3. Impact of task complexity

Figures 12 shows for ShuffleNet dark patterns with increase in complexity. R21D shows staggering grids. It supports our hypothesis that *model capacity* plays an important role to learn meaningful features and always increasing the complexity doesn't reciprocate to *better spatio-temporal features*.

10.4. Effect of data distrbituion

Figure 14 illustrates CKA maps for networks pretrained on *different source datasets* - for R21D pretrained on K400-50k

Networks	Parameters	GFLOPs	Rot†	VCOP †	PRP†	RSPNet
ShuffleNet	4.6M	1.1	**42.2**	**41.6**	**41.1**	**68.8**
MobileNet	3.1M	1.1	38.0	40.0	37.4	63.1
SqueezeNet	1.9M	1.8	41.3	41.4	39.2	62.9
C3D	27.7M	77.2	**57.7**	54.5	58.1	67.6
R3D	14.4M	39.8	51.1	50.7	52.1	62.1
R(2+1)D	14.4M	42.9	46.9	**56.8**	**58.9**	**78.0**

Table 10. **Comparison of FLOPs** and trainable parameters for each network on UCF101 dataset. † - pretraining on Kinetics 700 [9].

Epochs	Shuffle				R21D				Swin			
	10k	30k	50k	100k	10k	30k	50k	100k	10k	30k	50k	100k
50	59.1	66.3	68.1	68.9	66.8	71.1	75.0	77.2	-	40.4	44.9	52.0
100	60.3	67.6	68.7	69.0	69.5	75.2	76.1	80.0	37.2	44.3	49.6	58.5
150	61.8	66.7	69.4	69.7	69.5	76.6	76.5	78.8	37.9	46.2	50.7	61.3
200	61.5	68.2	68.5	69.9	69.6	76.6	77.4	78.3	36.8	46.3	52.5	61.5

Table 11. RSPNet with different subset size on ShuffleNet/R21D/VideoSwin on UCF101 dataset.

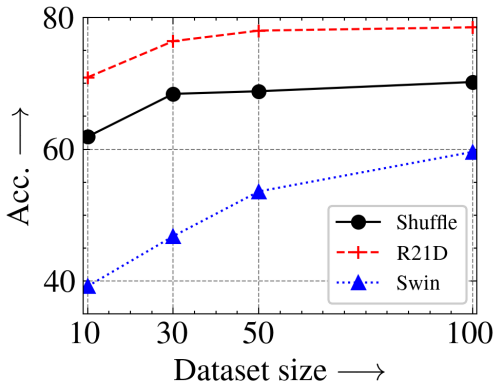

Figure 10. **Multiple architectures and data subsets on UCF101.** Pretext task is RSPNet. (x-axis: subset size, y-axis: Top-1 Accuracy) Here, 10 means 10k dataset subset, 30 means 30k and so on.

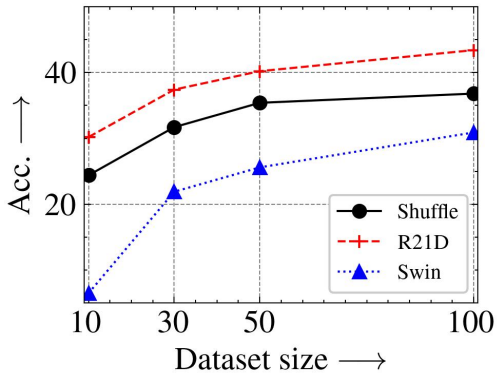

Figure 11. **Multiple architectures and data subsets on HMDB51.** Pretext task is RSPNet. (x-axis: subset size, y-axis: Top-1 Accuracy) Here, 10 means 10k dataset subset, 30 means 30k and so on.

	Non-contrastive			Contrastive		
	RotNet	VCOP	PRP	CVRL	TDL	RSP
No Noise	41.2	51.5	46.2	61.2	31.7	78.0
Gaussian	40.9	47.0	14.6	12.7	28.0	16.7
Impulse	38.1	30.5	5.4	3.5	18.8	8.5
Shot	33.4	45.1	20.9	26.4	21.5	45.1
Speckle	34.7	43.9	14.4	13.1	24.7	27.0

Table 12. Analysis of all pretext tasks with noise severity level 1 on R21D network on UCF101 dataset.

on VCOP and CVRL respectively. The stark difference in semi-block structure of *spatial (VCOP)* vs grid-like structure of *spatio-temporal (CVRL)* shows spatio-temporal outperforms spatial pretext task.

10.5. Robustness of SSL tasks

Table 12 shows performance of each pretext on each type of noise for severity level 1. Fig. 13 shows a relative decrease in performance for three different severity level on UCF101 dataset. *Non-contrastive* tasks are more robust than *contrastive* on average even at different severity levels.

10.6. Feature Analysis

We employ knowledge distillation to evaluate how complementary information from different datasets can be used to train a student model that could take advantage of this in terms of performance gain and training time reduction. Here we show the numbers quantitatively. Table 13 shows smaller architecture leans complementary information whereas bigger architecture depends on pretext task. Table 14 shows that

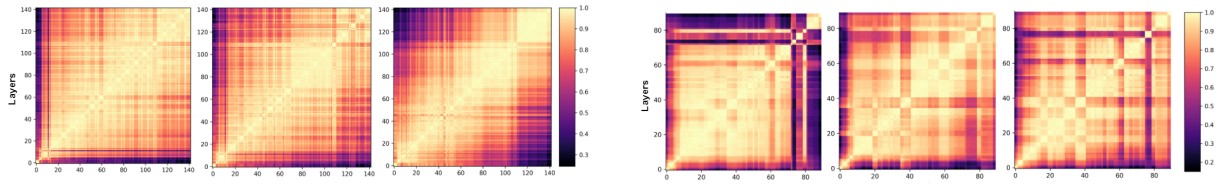

Figure 12. **Complexity CKA maps** PRP ShuffleNet (Left) and R21D (Right) network increasing complexity from 2 to 4 (Left to right). ShuffleNet has lower performance than R21D, and it shows darkest patterns when complexity is increased from 3 to 4. For both of these complexities, R21D shows staggering grids.

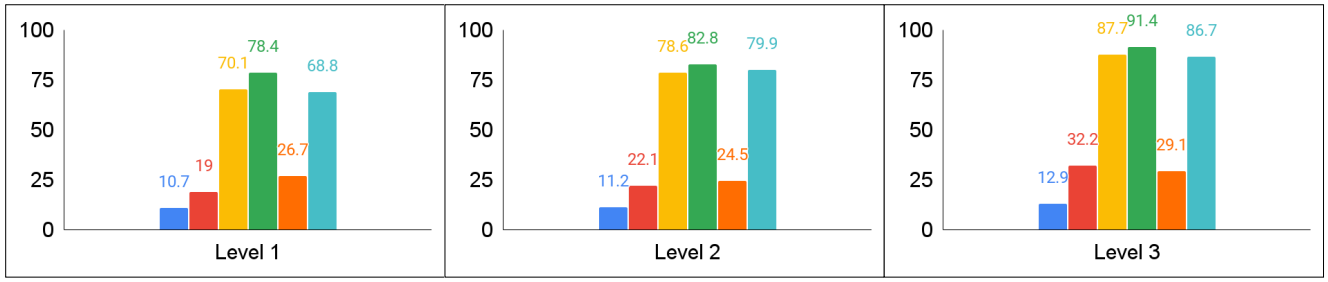

Figure 13. **Relative decrease in performance** at three different severity levels in increasing order from left to right. The pretext tasks is depicted by following colors - RotNet, VCOP, PRP, CVRL, TDL, RSPNet.

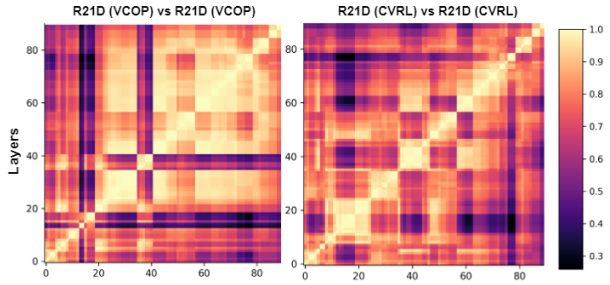

Figure 14. **Out-of-distribution CKA maps:** on VCOP and CVRL for R21D Network (Left to right). The semi-block structure of VCOP contrasts sharply with the grid-like structure of CVRL.

for each pretext task, we learn *complimentary information* from two *different source* datasets. Thus, student always outperforms the teachers. Table 15 shows that distilling knowledge from a *spatial* and a *temporal* task outperforms the standalone *spatio-temporal* task in both *contrastive* and *non-contrastive* case.

10.7. Clip retrieval

In Table 16, we show clip retrieval across different architectures on HMDB51 and UCF101 dataset. Amongst small capacity networks, ShuffleNet outperforms others and in medium-capacity R21D outperforms.

TC↓	RotNet	VCOP	PRP
T1	20.1/48.3	41.6/**56.8**	24.2/38.9
T2	20.2/**58.3**	41.8/54.8	18.1/44.4
T3	16.6/41.2	40.6/55.6	21.9/46.2
S	**75.0**/56.6	**75.4**/43.5	**76.1/61.0**

Table 13. **Complexity variation** with at three levels as teachers (T1, T2, T3) for all three pretext tasks. TC: Task complexity. Results are shown on UCF101 with ShuffleNet/R21D as backbones.

	K400 (T1)	SSV2(T2)	Student
RotNet	36.2	42.5	59.8
VCOP	50.4	59.7	67.6
CVRL	56.9	34.7	66.6
RSPNet	76.4	69.5	80.2

Table 14. **Out-of-Distribution** settings on UCF101 dataset using R21D network with teachers as different *source* datasets.

	S (T1)	T(T2)	Student
Non-Contrastive	RotNet	VCOP	61.1
Contrastive	CVRL	TDL	70.3

Table 15. **Knowledge distillation across different pretext tasks.** Teachers: ShuffleNet; Student: ShuffleNet.

Network	Top@1	Top@5
Squeeze	15.9/38.5	37.6/56.5
Mobile	16.2/37.4	36.5/55.6
Shuffle	19.3/43.1	42.0/62.1
C3D	19.9/43.2	43.4/61.6
R3D	19.3/40.4	42.5/60.2
R21D	18.2/42.7	40.1/62.8

Table 16. Top K Clip Retrieval on HMDB51/UCF101 across different architectures for RSPNet.

11. Main Table

In this section, we firstly expand the Table 6 (main paper) including results on HMDB51 dataset (Table 17). Knowledge distilled network discussed in the main paper still shows competitive performance on HMDB51. Going in depth, the works outperforming us are AVTS[39], GDT [54] in multimodal and VIMPAC [70], VideoMAE [73], TCLR [13] and CVRL [56] in single modality. AVTS and GDT uses two modalities, have more number of frames and AVTS also uses a bigger spatial size. Coming to Generative-based, both VIMPAC and VideoMAE uses a bigger backbone architecture. CVRL uses a longer temporal sequence and bigger frame resolution compared to ours and TCLR utilize 64 effective frames. Thus, the performance on HMDB51 is still competitive.

Approach	Venue	NxW/H	Backbone	Pre-training	UCF101	HMDB51
Generative						
VIMPAC [70]	-	10x256	ViT-L	HTM	92.7	65.9
VideoMAE [73]	NeurIPS'22	16x224	ViT-B	K400	91.3	62.6
VideoMAE * [73]	NeurIPS'22	16x112	R21D-18	K400	76.2	45.4
Context						
PacePred [83]	ECCV'20	16x112	R21D-18	K400	77.1	36.6
TempTrans [32]	ECCV'20	16x112	R3D-18	K400	79.3	49.8
STS [79]	TPAMI-21	16x112	R21D-18	K400	77.8	40.5
VideoMoCo [53]	CVPR'21	16x112	R21D-18	K400	78.7	49.2
RSPNet [10]	AAAI'21	16x112	R21D-18	K400	81.1	44.6
TaCo [6]	-	16x224	R21D-18	K400	81.8	46.0
TCLR[13]	CVIU'22	16x112	R21D-18	K400	88.2	60.0
CVRL† [56]	CVPR'21	32x224	R21D-18	K400	92.9	67.9
TransRank [17]	CVPR'22	16x112	R21D-18	K200	87.8	60.1
Multi-Modal						
AVTS [39]	NeurIPS'18	25x224	I3D	K400	83.7	53.0
GDT [54]	-	32x112	R21D	IG65M	95.2	72.8
XDC [4]	NeurIPS'20	32x224	R21D	K400	84.2	47.1
Ours *	-	16x112	R21D-18	K400-30k	97.3	51.5

Table 17. **Comparison with previous approaches** pre-trained on K400. Ours (* best performing) is RSPNet pretrained on 30k subset of K400. † modified backbone.