
Benchmarking Self-Supervised Video Representation Learning (Supplementary)

Anonymous Author(s)

Affiliation

Address

email

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

- 4 • Section 1: describes challenges and future work based on our study.
- 5 • Section 2: Pretext tasks explanation used in our analysis.
- 6 • Section 3: Training details about architectures, datasets, and, other hyperparameters.
- 7 • Section 4: We show additional CKA maps, more results on HMDB51 dataset and more
8 analysis on noise robustness. We added some tables for Knowledge distillation experiments
9 that were promised in the main paper.

10 1 Challenges and future work

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

15 2 Pretext Tasks Details

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

18 2.1 Spatial Transformation

19 **Rotation Net (13) (RotNet)** applies geometrical transformation on the clips. The videos are rotated
20 by various angles and the network predicts the class which it belongs to. Since the clips are rotated, it
21 helps the network to not converge to a trivial solution.

22 **Contrastive Video Representation Learning (21) (CVRL)** technique applies temporally coherent
23 strong spatial augmentations to the input video. The contrastive framework brings closer the clips
24 from same video and repels the clip from another video. With no labels attached, the network learns
25 to cluster the videos of same class but with different visual content.

26 2.2 Temporal Transformation

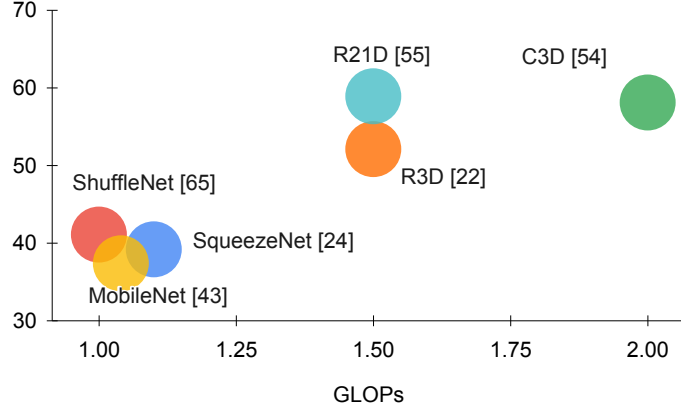


Figure 1: **Architecture Performance Analysis:** Variation in performance for different architectures. X-axis shows the relative floating point operations and Y-axis shows the Top-1 Accuracy.

27 **Video Clip Order Prediction (32) (VCOP)** learns the representation by predicting the permutation
 28 order. The network is fed N clips from a video and then it predicts the order from N! possible
 29 permutations.

30 **Temporal Discriminative Learning (30) (TDL)** In contrast to CVRL, TDL works on temporal
 31 triplets. It looks into the temporal dimension of a video and targets them as unique instances. The
 32 anchor and positive belongs to same temporal interval and has a high degree of resemblance in visual
 33 content compared to the negative.

34 2.3 Spatio-Temporal Transformation

35 **Playback Rate Prediction (33) (PRP)** has two branch, generative and discriminative. Discrimina-
 36 tive focuses on the classifying the clip's sampling rate, whereas, generative reconstructs the missing
 37 frame due to dilated sampling. Thus, the first one concentrates on temporal aspect and second one on
 38 spatial aspect.

39 **Relative Speed Perception Network (5) (RSPNet)** applies contrastive loss in both spatial and
 40 temporal domain. Clips are samples from a same video to analyze the relative speed between them.
 41 A triplet loss pulls the clips with same speed together and pushes clips with different speed apart in
 42 the embedding space. To learn spatial features, InfoNCE loss (29) is applied. Clip from same video
 43 are positives whereas clips from different videos are negatives.

44 3 Implementation Details

45 3.1 Architecture Details

46 Preliminary research has shown that 3D networks (28; 10) have outperformed 2D CNN variants on
 47 video recognition tasks. We looked into three types of capacity - small, medium and big on the basis
 48 of number of trainable parameters. Fig. 1 shows comparison of multiple architectures in terms of
 49 GFLOPs and Top-1 accuracy.

50 **Small capacity networks:** are resource efficient, implying they can be trained in larger batches
 51 within short span of time. The network selection is done on the basis of supervised training scores on
 52 Kinetics(14) and UCF101(15). ShuffleNet V1 2.0X (34) utilizes point-wise group convolutions and
 53 channel shuffling. SqueezeNet (12) reduces the filter size and input channels to reduce the number of

parameters. MobileNet (23) 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.

Large Capacity networks: ViViT (2) Timesformer (3), VideoSwin (19) and MViT (7) fall under this category.

In small capacity networks, based on (15), 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.

3D-ShuffleNet V1 2.0X (34): It utilizes point-wise group convolutions and channel shuffling and has 3 different stages. Within a stage, the number of output channels 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 (12): 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 (23): 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 (27): 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 (10): 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 (28): 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.

The architectures of large capacity networks are described as follows:

VideoSwin (19): It is an inflated version of original Swin (18) 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.

The performance across different architectures are shown in Table 1 and Figure 1. ShuffleNet and R21D performs the best across small and medium capacity networks in most of the pretext tasks. Thus, we choose ShuffleNet and R21D for our benchmark analysis.

3.2 Clip Length

Different pretext tasks take 16 or 32 frames as input clip length. We experimented with both 16 and 32 clips length and observe that 32 frames mostly provide better performance. However, to maintain

Table 1: **Comparison of FLOPs** and trainable parameters for each network on UCF101 dataset. [†] - pretraining on Kinetics 700 (4) (%).

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 2: **Downstream accuracy** classification on UCF-101 dataset. FT: Finetuning LP: Linear Probing (%).

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

consistency with most of the approaches and reduce computation costs, we use 16 frames in our experiments.

3.3 Linear probing vs Finetuning

For linear probing, we train only the linear layers attached for classification while freezing other network weights, whereas in finetuning the whole network is trained end-to-end. In our preliminary experiments we use Kinetics-400 for pretraining and UCF-101 as the target dataset. From Table 2, on several pretext tasks, we observe an average drop of 25% (ShuffleNet) and 40% (R21D) in performance when comparing linear probe with finetuning. However, we do not usually observe this significant drop when both the pretraining and target datasets are the same (24). It indicates that *finetuning is important for the model to adapt to downstream dataset* in case it is different. Therefore, some of the existing works (26) rely on finetuning when the source and target datasets are different. Since we are interested in cross-dataset learning, we perform finetuning on all our downstream datasets.

3.4 Pretext Tasks

Configurations: 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.

Evaluations: A comparison of pretext tasks on two different backbones is shown in Table 3. We observe that most of the *contrastive* tasks outperform *non-contrastive* tasks when they are trained under different constraints (row 3). However, that is not the case when we compare them under the same constraints (row 1-2). Similarly, *spatial* and *spatio-temporal* tasks have a similar performance from reported results. However, *spatio-temporal* pretext tasks outperform spatial ones by a large margin when we keep pre-training constraints similar. This supports our hypothesis that it is important to experiment under similar constraints for a fair evaluation of different approaches.

Table 3: **Comparison across different pretext tasks** pre-train on K400-50k subset and finetuned on UCF101 dataset against *reported* results in the original paper (%).

	Non-Contrastive			Contrastive		
	Rot	VCOP	PRP	CVRL	TDL	RSP
	(S)	(T)	(ST)	(S)	(T)	(ST)
Shuffle	16.6	40.8	21.9	62.3	12.4	68.8
R21D	41.2	51.5	46.2	61.2	31.7	78.0
<i>Reported</i> *	72.1	68.4	72.4	94.4	84.9	93.7

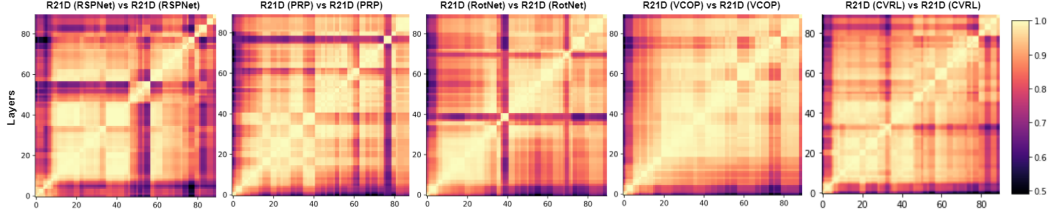


Figure 2: **Pretext tasks CKA maps** for RSPNet, PRP, RotNet, VCOP, CVRL on K-400 50k subset using R21D network (Left to right). R21D pretrained on K400 shows a semi-block structure for VCOP, indicating near-saturation condition of the network on this pretext task. It shows a more prominent grid-based structure on CVRL and RSPNet instead. These observations corroborate the quantitative results, where pretraining on K400 for both CVRL and RSPNet gives better performance.

127 Additionally, Figure 2 depicts the hidden representations of R21D network pretrained on different
 128 pretext tasks. Here the 50k subset of K-400 was used for pretraining, and finetuned on UCF-101.

129 3.5 Datasets

130 Here we discuss datasets in detail. We use Kinetics-400 (K400) (14) and Something-Something V2
 131 (9) for our pre-training. For the downstream task evaluation, we perform our experiments on UCF-101
 132 (25), HMDB-51 (16), and Diving48 (17). Since, the pretraining and finetuning datasets are different,
 133 the performance variation will provide us a better picture about how much meaningful spatiotemporal
 134 features are learned by these networks. K400 has approximately 240k videos distributed evenly
 135 across 400 classes respectively. The approximate number of videos in finetuning datasets are: 1)
 136 UCF101-10k, 2) HMDB51-7k, and, 3) Diving48-16k. The datasets can be categorized into two ways:

137 **Appearance-based:** Kinetics, UCF101 and HMDB51 comes under this category (6; 11). Kinetics
 138 videos length are generally 10s centered on human actions. It mainly constitutes singular person
 139 action, person-to-person actions and person-object action. For pre-training, we select a random subset
 140 of videos and maintain equal distribution from each class. Unless otherwise stated, pre-training is
 141 done on K400-50k subset for all experiments.

142 **Temporal-based:** In Kinetics, we can estimate the action by looking at a single frame (6; 11). From
 143 Fig. 3, top two rows, we can see the person with a javelin and basketball. This information helps in
 144 class prediction. Looking at bottom two rows (SSv2 and Diving48 respectively), we can't describe
 145 the activity class until we look into few continuous frames. It shows that temporal aspect plays an
 146 important role for these datasets, that's why we categorize them into temporal-based datasets.

147 **UCF-101 (25) :** It's an action recognition dataset that spans over 101 classes. There are around
 148 13,300 videos, with 100+ videos per class. The length of videos in this dataset varies from 4 to
 149 10 seconds. It covers five types of categories: human-object interaction, human-human interaction,
 150 playing musical instruments, body motion and sports.

151 **HMDB-51 (16) :** The number of videos in this dataset is 7000 comprising 51 classes. For each
 152 action, at least 70 videos are for training and 30 videos are for testing. The actions are clubbed
 153 into five categories: 1) General facial actions, 2) Facial actions with body movements, 3) General



Figure 3: **Example** sample from each dataset.

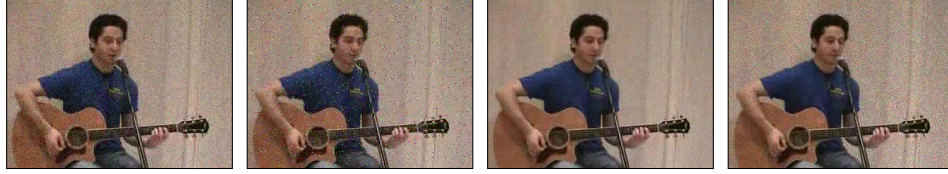


Figure 4: **Example frame sample for each noise** Gaussian, Impulse, Shot and Speckle from left to right. Sample clips are provided in supplementary.

Table 4: **Evaluation of different pretext tasks** on different subset size on R21D network on HMDB51 dataset.

Epochs	Non-contrastive			Contrastive		
	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

body movements, 4) Body movements with object interaction, and, 5) Body movements for human interaction.

3.6 Noisy Datasets

We have shown the examples of each dataset used in the paper in Fig. 3.

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. 4.

4 Additional Results

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

4.1 Effect of dataset size

Diminishing returns Looking across different architectures in Figure 6, there’s a minimal gain for R21D and ShuffleNet beyond increasing dataset size from 30k subset against VideoSwin which improves with an increase in dataset size which relates to similar behavior like image models discussed in (8).

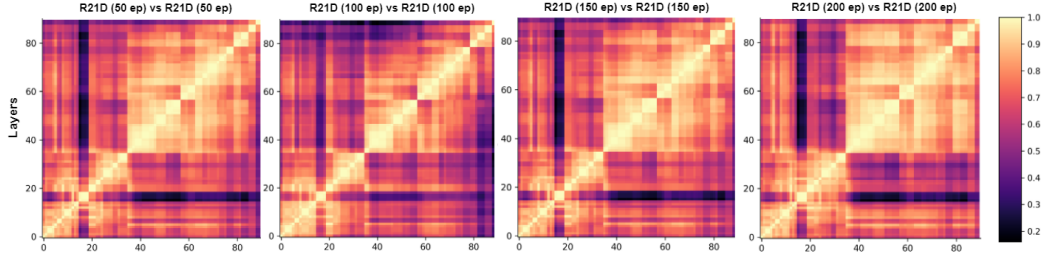


Figure 5: **Training time CKA maps** on 50, 100, 150, 200 epochs of R21D network on RSPNet pretext for K-400 10k subset (Left to right). The block structure is visible from 50 epochs itself, which then darkens and becomes prominent by 200 epochs. With 10k subset, the saturation starts hitting at 100 epochs.

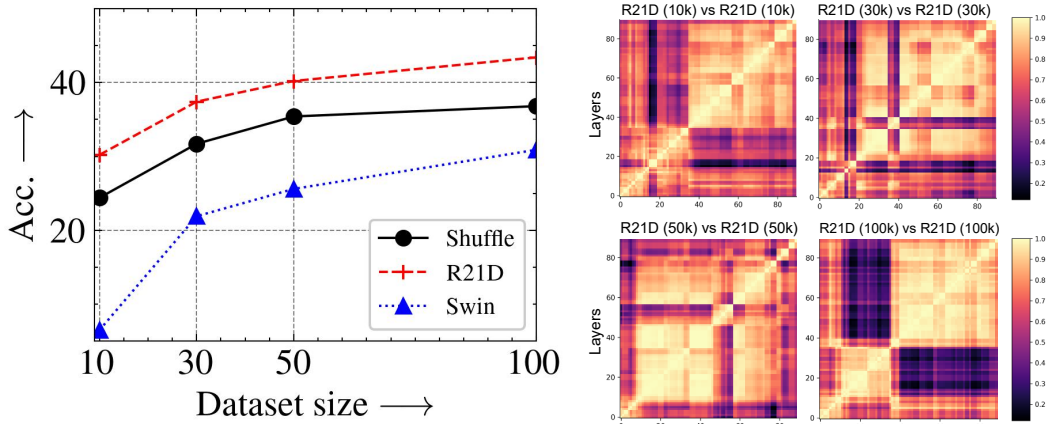


Figure 6: Left: **Dataset subset** performance for three different architectures on RSPNet pretext task (x-axis: subset size, y-axis: Top-1 Accuracy). Here, 10 means 10k dataset subset, 30 means 30k, and so on. Right: **CKA maps** for RSPNet on different subsets with R21D backbone.

Table 5: RSPNet with different subset size on ShuffleNet/R21D/VideoSwin on UCF101 dataset.

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

170 **HMDB51** In Table 4, we extend results for different pretext tasks on HMDB51 dataset. Similar
171 to UCF101, *the scale in subset size doesn't reciprocate to gain in performance* for all pretext tasks
172 on HMDB51 dataset. From Figures 7 and ??, we see that performance increase for Swin by a good
173 margin, whereas in case of ShuffleNet and R21D it's relatively less beyond 50k subset.

174 **Training time** Table 5 shows VideoSwin saturates at 150 epochs on UCF101 whereas CNN
175 architectures saturates earlier (100 epochs) which reflects limitation of model capacity. Figure 5
176 shows the emergence of block structures for R21D network trained on RSPNet for K400 10k. The
177 saturation point has reached earlier around 100 epochs which supports the hypothesis in main work
178 that CNN architectures mostly saturates around 100 epochs. We see similar pattern even after
179 increasing the dataset size.

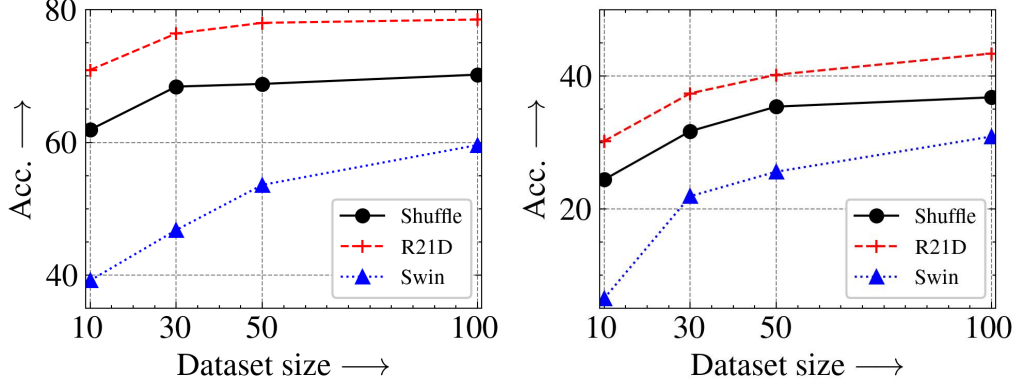


Figure 7: **Multiple architectures and data subsets.** Pretext task is RSPNet. (x-axis: subset size, y-axis: Top-1 Accuracy) Here, 10 means 10k dataset subset, 30 means 30k and so on. Left: UCF-101, right: HMDB51.

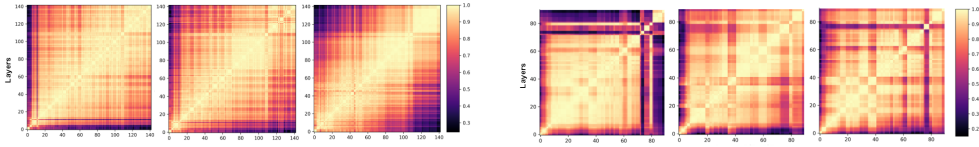


Figure 8: **Complexity CKA maps** PRP ShuffleNet (Left) and R21D (Right) network increasing complexity from 2 to 4 (Left to right). ShuffleNet has lower performance than R21D, and it shows darkest patterns when complexity is increased from 3 to 4. For both of these complexities, R21D shows staggering grids.

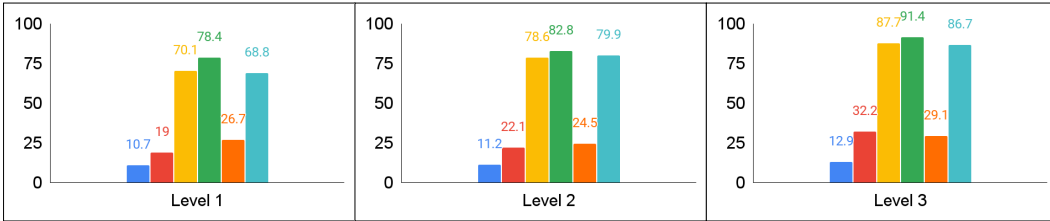


Figure 9: **Relative decrease in performance** at three different severity levels in increasing order from left to right. The pretext tasks is depicted by following colors - RotNet, VCOP, PRP, CVRL, TDL, RSPNet.

180 4.2 Impact of task complexity

181 Figures 8 shows for ShuffleNet dark patterns with increase in complexity. R21D shows staggering
 182 grids. It supports our hypothesis that *model capacity* plays an important role to learn meaningful
 183 features and always increasing the complexity doesn't reciprocate to *better spatio-temporal features*.

184 4.3 Effect of data distribution

185 Figure 10 illustrates CKA maps for networks pretrained on *different source datasets* - for R21D
 186 pretrained on K400-50k on VCOP and CVRL respectively. The stark difference in semi-block
 187 structure of *spatial* (VCOP) vs grid-like structure of *spatio-temporal* (CVRL) shows spatio-temporal
 188 outperforms spatial pretext task.

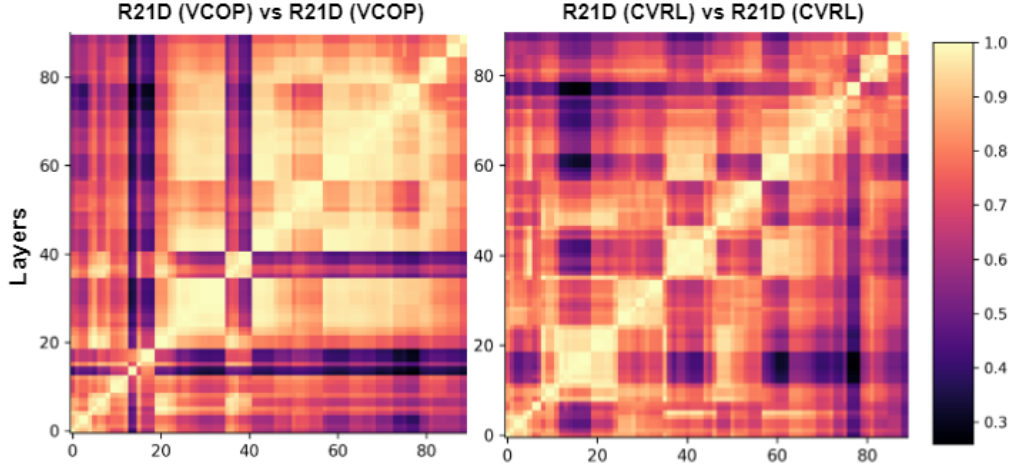


Figure 10: **Out-of-distribution CKA maps:** on VCOP and CVRL for R21D Network (Left to right). The semi-block structure of VCOP contrasts sharply with the grid-like structure of CVRL.

Table 6: Analysis of all pretext tasks with noise severity level 1 on R21D network on UCF101 dataset.

	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

189 4.4 Robustness of SSL tasks

190 Table 6 shows performance of each pretext on each type of noise for severity level 1. Fig. 9 shows a
 191 relative decrease in performance for three different severity level on UCF101 dataset. *Non-contrastive*
 192 tasks are more robust than *contrastive* on average even at different severity levels.

193 4.5 Feature Analysis

194 We employ knowledge distillation to evaluate how complementary information from different datasets
 195 can be used to train a student model that could take advantage of this in terms of performance gain and
 196 training time reduction. Here we show the numbers quantitatively. Table 7 shows smaller architecture
 197 leans complementary information whereas bigger architecture depends on pretext task. Table 8 shows
 198 that for each pretext task, we learn *complementary information* from two *different source* datasets.
 199 Thus, student always outperforms the teachers. Table 9 shows that distilling knowledge from a
 200 *spatial* and a *temporal* task outperforms the standalone *spatio-temporal* task in both *contrastive* and
 201 *non-contrastive* case.

202 4.6 Study on Video Foundation Models

203 In Table 11, we show performance of different ViFMs zero shot accuracy on UCF101.

204 References

- 205 [1] Shahzad Ahmad, Sukalpa Chanda, and Yogesh S Rawat. Ez-clip: Efficient zeroshot video action recognition.
 206 *arXiv preprint arXiv:2312.08010*, 2023.

Table 7: **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.

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

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

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

Table 10: Top K Clip Retrieval on HMDB51/UCF101 across different architectures for RSPNet.

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

- [2] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lucic, and Cordelia Schmid. Vivit: A video vision transformer. *ArXiv*, abs/2103.15691, 2021.
- [3] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *ArXiv*, abs/2102.05095, 2021.
- [4] J. Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *ArXiv*, abs/1907.06987, 2019.
- [5] Peihao Chen, Deng Huang, Dongliang He, Xiang Long, Runhao Zeng, Shilei Wen, Minghui Tan, and Chuang Gan. Rspnet: Relative speed perception for unsupervised video representation learning. In *AAAI*, 2021.
- [6] Jinwoo Choi, Chen Gao, Joseph C.E. Messou, and Jia-Bin Huang. Why can’t i dance in the mall? learning to mitigate scene bias in action recognition. In *NeurIPS*, 2019.
- [7] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. *ArXiv*, abs/2104.11227, 2021.
- [8] Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. *arXiv preprint arXiv:1905.01235*, 2019.
- [9] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fründ, Peter N. Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thureau, Ingo Bax, and Roland Memisevic. The “something something” video database for learning and evaluating visual common sense. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5843–5851, 2017.
- [10] K. Hara, H. Kataoka, and Y. Satoh. Learning spatio-temporal features with 3d residual networks for action recognition. *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages

Table 11: **Selected ViFMs** based on type, pretraining objective, pretraining data and clip length x sample rate. Zero-shot classification accuracy on UCF-101 from all ViFMs.

ViFM	Type.	Pretraining Data	Frames x Rate	Accuracy
ViFi-CLIP (22)	image-based	Kinetics-400	32 x 2	77.329
X-CLIP (20)	image-based	Kinetics-400	8 x 8	71.614
EZ-CLIP (1)	image-based	Kinetics-400	8 x 8	70.573
ViCLIP (31)	video-based	InternVid-10M-FLT	8 x 8	75.542
LanguageBind (35)	video-based	VIDAL-10M	8 x 8	69.973

- 3154–3160, 2017.
- [11] De-An Huang, Vignesh Ramanathan, Dhruv Mahajan, Lorenzo Torresani, Manohar Paluri, Li Fei-Fei, and Juan Carlos Niebles. What makes a video a video: Analyzing temporal information in video understanding models and datasets. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7366–7375, 2018.
- [12] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. SqueezeNet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size, 2016. cite arxiv:1602.07360Comment: In ICLR Format.
- [13] Longlong Jing, Xiaodong Yang, Jingen Liu, and Y. Tian. Self-supervised spatiotemporal feature learning via video rotation prediction. *arXiv: Computer Vision and Pattern Recognition*, 2018.
- [14] W. Kay, J. Carreira, K. Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, F. Viola, T. Green, T. Back, A. Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *ArXiv*, abs/1705.06950, 2017.
- [15] Okan Köpüklü, Neslihan Kose, Ahmet Gunduz, and Gerhard Rigoll. Resource efficient 3d convolutional neural networks. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 1910–1919. IEEE, 2019.
- [16] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. Hmdb: A large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563, 2011.
- [17] Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards action recognition without representation bias. In *ECCV*, 2018.
- [18] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021.
- [19] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3192–3201, 2022.
- [20] Bolin Ni, Houwen Peng, Minghao Chen, Songyang Zhang, Gaofeng Meng, Jianlong Fu, Shiming Xiang, and Haibin Ling. Expanding language-image pretrained models for general video recognition. In *European Conference on Computer Vision*, pages 1–18. Springer, 2022.
- [21] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, H. Wang, Serge J. Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6960–6970, 2021.
- [22] Hanoona Rasheed, Muhammad Uzair Khattak, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Fine-tuned clip models are efficient video learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6545–6554, 2023.
- [23] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [24] Madeline C Schiappa, Yogesh S Rawat, and Mubarak Shah. Self-supervised learning for videos: A survey. *ACM Computing Surveys*.
- [25] K. Soomro, A. Zamir, and M. Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *ArXiv*, abs/1212.0402, 2012.
- [26] Fida Mohammad Thoker, Hazel Doughty, Piyush Bagad, and Cees G. M. Snoek. How severe is benchmark-sensitivity in video self-supervised learning? In *ECCV*, 2022.
- [27] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV ’15, page 4489–4497, USA, 2015. IEEE Computer Society.
- [28] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri. A closer look at spatiotemporal convolutions for action recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018.
- [29] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *ArXiv*, abs/1807.03748, 2018.

- 281 [30] Jinpeng Wang, Yiqi Lin, Andy Jinhua Ma, and Pong Chi Yuen. Self-supervised temporal discriminative
282 learning for video representation learning. *ArXiv*, abs/2008.02129, 2020.
- 283 [31] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinyuan Chen, Yaohui Wang, Ping
284 Luo, Ziwei Liu, Yali Wang, Limin Wang, and Yu Qiao. Internvid: A large-scale video-text dataset for
285 multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*, 2023.
- 286 [32] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal
287 learning via video clip order prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision*
288 *and Pattern Recognition (CVPR)*, June 2019.
- 289 [33] Yuan Yao, Chang Liu, Dezhao Luo, Yu Zhou, and Qixiang Ye. Video playback rate perception for self-
290 supervised spatio-temporal representation learning. *2020 IEEE/CVF Conference on Computer Vision and*
291 *Pattern Recognition (CVPR)*, pages 6547–6556, 2020.
- 292 [34] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional
293 neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and*
294 *Pattern Recognition (CVPR)*, June 2018.
- 295 [35] Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiayi Cui, Wang HongFa, Yatian Pang, Wenhao Jiang, Junwu
296 Zhang, Zongwei Li, Cai Wan Zhang, Zhifeng Li, Wei Liu, and Li Yuan. Languagebind: Extending
297 video-language pretraining to n-modality by language-based semantic alignment, 2023.