Self-Supervised Learning via Multi-Transformation Classification for Action Recognition

1st Duc-Quang Vu  
Dept. of CSIS  
Thai Nguyen University of Education  
Thai Nguyen, Vietnam  
quangvd@tnue.edu.vn

2nd Ngan Le  
Dept. of CSCE  
University of Arkansas  
Fayetteville, USA  
thile@uark.edu

3rd Jia-Ching Wang  
Dept. of CSIE  
National Central University  
Taoyuan, Taiwan  
jcw@csie.ncu.edu.tw

Abstract—Self-supervised tasks have been utilized to build useful representations that can be used in downstream tasks when the annotation is unavailable. In this paper, we introduce a self-supervised video representation learning method based on the multi-transformation classification to efficiently classify human actions. Self-supervised learning on various transformations not only provides richer contextual information but also enables the visual representation more robust to the transforms. The spatio-temporal representation of the video is learned in a self-supervised manner by classifying seven different transformations i.e. rotation, clip inversion, permutation, split, join transformation, color switch, frame replacement, and noise addition. First, seven different video transformations are applied to video clips. Then the 3D convolutional neural networks are utilized to extract features for clips and these features are processed to classify the pseudo-labels. We use the learned models in pretext tasks as the pre-trained models and fine-tune them to recognize human actions in the downstream task. We have conducted the experiments on UCF101 and HMDB51 datasets together with C3D and 3D Resnet-18 as backbone networks. The experimental results have shown that our proposed framework outperformed other SOTA self-supervised action recognition approaches.

Index Terms—Self-supervised learning, Action Recognition, 3D ResNet, C3D, multi-transformation.

I. INTRODUCTION

Human action recognition is one of the most fundamental research problems in computer vision and machine learning. Besides speech processing [1], [2] and natural language processing [3], action recognition has attracted huge attention over the last decade with the availability of large-scale video datasets ([4], [5]). However, annotating new video datasets is always required to address the problems in new domains. Annotation is time-consuming and labor-intensive; thus, it is useful if we can leverage the unlabeled data. As one of the most widely used datasets for pre-training very deep 2D Convolutional Neural Networks (CNNs), ImageNet ([6]) contains about 1.3 million labeled images covering 1,000 classes. Furthermore, the collection and annotation of video datasets are more expensive than image datasets due to the temporal dimension. The Kinetics-600 dataset ([7]), which consists of 500,000 10-second videos belonging to 600 categories, is mainly used to train networks for video action recognition.

To avoid time-consuming and expensive data annotations, many self-supervised methods ([5], [8]–[10]) have been proposed to learn visual features from large-scale datasets for video recognition tasks based on self-supervised learning without using any human annotations. Self-supervised learning constructs a pre-training or “pretext” task used to extract knowledge from unlabeled data. After training a model on the pretext task, it can then be adapted to the target task through transfer learning. Self-supervised learning approaches usually involve transforming the input data to force the model to predict missing parts of the data or recognize the transformations applied to the data or introduce some information bottleneck. A pretext task with pseudo-labels is automatically generated to exploit data structure. A CNN model is then trained to solve tasks where pseudo-labels can be easily derived from input data without human labors. Some examples of those such tasks are solving a jigsaw puzzle of image patches ([9]), predicting frames order ([10]–[12]), motion and appearance statistics ([8]), image color channel ([13]), etc. The CNN can then be directly applied to other video tasks as a feature extractor or to be used as a weight initialization for downstream tasks.

In this paper, we propose a novel self-supervised learning approach to learn video representations by classifying the transformation that was applied to the video. The success of multi-task self-supervised learning inspires our proposed method ([14]–[17]). Our proposed framework contains two parts corresponding to the pretext task and the downstream task, as shown in Fig.1. The 3D CNN model is learned in the pretext task by applying seven transformations to change the appearance and/or motion in the input video clip. The transformation is then classified using the pseudo-labels where each label is assigned to one transformation. The 3D CNN model is later used as a pre-trained model for the downstream task to recognize human actions. To summarize, we make the following three contributions of this work (i) we have proposed an effective self-supervised framework for action recognition based on the multi-transformation classification. Different from the existing methods which focus on either spatial or temporal domain, our transformer is able to cover both temporal and spatial domains. In the pretext task, the multi-transformation is applied to the input videos to model the spatio-temporal features. (ii) From the ablation study results, we have demonstrated that our multi-transformations enable the model to learn richer contextual spatio-temporal
features. (iii) We have conducted experiments on C3D and 3D ResNet-18 backbone networks. On both benchmarks, our proposed framework robustly exhibits strong performances i.e. outperforms SOTA approaches on the UCF101 and HMDB51 datasets regardless of the backbone networks.

A. Action Recognition

Achieving state-of-the-art performance is the most important task in action recognition. Various approaches have been proposed to address this task such as knowledge-distillation-based methods [18]–[20], novel backbones [21], [22], etc.

Different from these mentioned fully supervised methods. Self-supervised learning aims at learning visual features from unlabeled data in pretext tasks. The learned visual representation model in the pretext task is then transferred to the downstream task. The objective of self-supervised learning focuses on extracting good feature representations without annotation; thus, it targets designing an effective pretext task component [23].

Villegas et al. [24] proposed a deep neural network that uses both optical flow frames and the RGB frame to predict one future frame. With the input as a tuple of frames order, Misra et al. [10] proposed a method that allows verifying whether the temporal order is correct or not. To solve this pretext task, the authors proposed a ConvNet model which all input video frames are passed through the model. The objective of the model figure out whether the frames are in the correct order or not. In doing so, the model learns not just spatial features but also takes into account temporal features.

Inspired by the frames reordering task, Kim et al. [25] introduced a self-supervised task called Space-Time cubic puzzles. Given a randomly permuted sequence of 3D spatio-temporal pieces cropped from a video clip. The 3D CNN is used to learn both spatial and temporal relations from the input video frames and predict their original arrangement. Fernando et al. [5] presented a self-supervised CNN called O3N to predict an odd video from a set of otherwise related input videos. The goal of O3N is to predict an odd video from a set of otherwise related input videos. The network’s input is a tuple of videos where one of the videos has the wrong temporal order of frames while the other ones have the correct temporal order.

Unlike the above methods, a model based on deep reinforcement learning is introduced in [26]. In this work, the deep reinforcement model is proposed to learn a policy that proposes best-suited permutations from errors the model has made when recovering frame order. Wang et al. [8] presented a pretext task to predict motion and appearance statistics. Each video frame is first divided into several spatial regions, and it is then predicted by selecting the largest motion and direction.

Far apart from the previous self-supervised learning approaches, which focus only on a single transformation during training pretext tasks, our proposed network inherits the advantages of multiple transformations, including frame rotation, color switching, video clip inversion, noise addition to frames, video joining, and splitting, permutation, and frame replacement. Learning various transformations not only provides richer contextual information but also enables the visual representation to be more robust to the transformations.

A. Pretext Task

In this task, we apply multi-transformations to an input video. Let \( x' = G(x, m) \) be the transformed video where \( x \) denotes the original video and \( m \) indicates the transformation types. According to our choice of seven transformations, \( m \) is the vector in \( \mathbb{R}^7 \), each element in \( m \) set as 1 or 0 depending on if a transformation is applied or not. In this task, we adopt a 3D CNN (either C3D ([27]) or 3D ResNet18 ([28])) as a feature extractor. The feature extractor network is parameterized by \( \theta \) and denoted as \( F(x'|\theta) \). The feature extracted from \( F \) is then predicted what transformation kinds have been applied to the original video. Given a video \( x \), the network parameters \( \theta \) are learned by minimizing the following objective function:

\[
\mathcal{L}(x|\theta) = - \sum_{z=0}^{M} (z \log F(x'|\theta) + (1 - z) \log(1 - F(x'|\theta))
\]

where \( M = 7 \) corresponds to the total number of transformations. In our approach, we randomly select a few transforms among \( M (M = 7) \) transforms and apply them to one video. As shown in Eq 1, we use the cross-entropy loss for the binary classification to predict if a single transformation \( m \) is applied

![Diagram of self-supervised learning via multi-transformation classification.](image)

Fig. 1. Overview of self-supervised learning via multi-transformation classification.
We propose 7 different transformations as follows:

The vector presents the probability of applying transformation \( m \). Given a set of \( N \) training videos, the overall training loss function in the pre-text task is as follows:

\[
L_1 = \min_{\theta} \frac{1}{N} \sum_{i=0}^{N} L(x_i|\theta) \tag{2}
\]

where \( L(x_i|\theta) \) is defined in Eq.1. The overview of the pretext task module in our proposed network is given in Fig. 1.

We propose 7 different transformations as follows:

- **Frame rotation**: Different rotation angles i.e. 90°, 180°, or 270° are applied into frames.
- **Color switching**: Besides the actual color channel order as R-G-B, the color channel is inverted to other orders such as B-G-R.
- **Noise addition**: In this transformation, a Gaussian distribution with zero mean, and standard deviation is randomly set from 0.1 to 0.3 is added into frames.
- **Frame replacement**: In this transformation, a frame from the given video clip is replaced by a noise frame that is generated from a uniform distribution.
- **Clip inversion**: Given an original video clip \( X(t) \) where \( t = 1, 2, ..., N \) and \( N \) is the length of the video clip, the inverted clip is expressed as \( X'(t) \) where \( t = N, N - 1, ..., 1 \).
- **Splitting and joining**: In this transformation, a video clip is divided into two parts and one of them is replaced by another part from another video clip which has the same length/dimension.
- **Permutation**: All frames in the given clip are randomly shuffled.

We apply seven transformations to the video clip and send these transformed images to the pretext task network to predict what sort of transformation was applied to the video clip and the network simply performs an 8-class classification to predict the transformation.

Fig.2 illustrates examples of transformations together with the pseudo labels. The original video clip is given in the first row, and its pseudo label is zero. In contrast, the remaining rows are transformed video clips, and their pseudo labels are assigned from 1 to 7, corresponding to different transformations.

**B. Downstream Task**

In our self-supervised learning framework, the downstream task performs action recognition to evaluate the quality of features learned by the pretext task. Given a set of \( N \) samples is denote as \((x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\) such that \( x_i \) is a video clip and \( y_i \) is its label (i.e., action). In this stage, the model \( F \) is fine-tuned to learn a mapping \( F : x_i \rightarrow y_i \). Let \( p_i \) be a probability distribution over labels, where \( p_i = F(x_i, \theta') \) and \( \theta' \) is the set of trainable parameters. The correctness of the prediction is measured using cross-entropy as follows:

\[
L_{DT}(y_i, p_i) = -\frac{1}{N} \sum_{i=0}^{N} y_i \log(p_i) \tag{3}
\]

The loss \( L_{DT} \) is then backpropagated to optimize the whole framework. When the model is trained to predict the action classes, the 3D CNN is trained to extract clips’ meaningful features. A fully connected layer with the softmax function is applied over to output the final prediction.

**IV. Experiment**

**A. Datasets and Implementation**

We have conducted experiments on two datasets including HMDB51 ([29]) and UCF101 ([30]).

**HMDB51**: is a small dataset including 6,766 videos from 51 human action classes. The average duration of each video is about 3 seconds. Three train/test splits (70% training and 30% testing) are provided in this dataset.

**UCF101**: is similar to HMDB51. UCF101 includes 13,320 action instances from 101 human action classes. The average duration of each video is about 7 seconds. Three train/test splits also are provided in this dataset.

**Fig. 2.** An illustration of seven different transformations that are applied to the original video clip given in the 1st row. Each transformation is assigned one pseudo label. Label 0: original video clip, label 1: frame rotation, label 2: color switch, label 3: noise addition, label 4: frame replacement, label 5: clip inversion, label 6: split and join, label 7: frames permutation.

| Label | Transformation |
|-------|----------------|
| 0     | Original clip  |
| 1     | Frame rotation |
| 2     | Color switching|
| 3     | Noise addition |
| 4     | Frame replacement |
| 5     | Clip inversion |
| 6     | Split and join |
| 7     | Frames permutation |
Training the pretext task: We split the videos in the datasets into many clips with 16 contiguous frames of length. Each frame of the clip is scaled with a shorter edge of 256, and the other edge is calculated so that it still maintains the frame aspect ratio. Then, frames are randomly cropped using window sizes of $224 \times 224$ (center cropped for the testing process). To create a transformed video clip from the original, we randomly chose the transformations described above and applied them to the video clip. The pseudo-labels are generated correspondingly for each transformation. We set the mini-batch size to 16. We use the stochastic gradient descent (SGD) optimizer with an initial learning rate of 0.01 and a momentum of 0.9. The training process is done in 100 epochs.

Training the downstream task: When the pre-training stage with the pretext task is done, we transfer the model to the downstream task. We set the mini-batch size of 16 and the initial learning rate of 0.001. We used the SGD optimizer with a momentum of 0.9. Each video is split into different 16-frame clips in the testing, and the class scores are averaged over all the video clips.

B. Performance and Comparison

To demonstrate the quality of the learned video features from our self-supervised models, we fine-tune our models on the action recognition datasets. As shown in Table I, all results are top-1 accuracy in action recognition on two standard datasets. The results in the table contain three parts.

| Method              | Backbone | UCF101 (%) | HMDB51 (%) |
|---------------------|----------|------------|------------|
| Random Init         | C3D      | 45.4       | 19.7       |
| Random Init         | 3D Res-18| 46.5       | 17.1       |
| Shuffle & Learn [10]| AlexNet  | 50.9       | 19.8       |
| OPN [31]            | VGG-M-2048| 59.8       | 23.8       |
| O3N - Sum-of-diff. [5]| AlexNet | 54.3       | 25.9       |
| O3N - Dynamic image [5]| AlexNet | 53.2       | 26.0       |
| O3N - Stack-of-Diff. [5]| AlexNet | 60.0       | 32.5       |
| Geometry [32]       | 3D Res-18| 54.1       | 22.6       |
| CrossLearn [33]     | CaffeNet | 58.7       | 27.2       |
| Video Jigsaw* [9]   | C3D      | 55.4       | 27.0       |
| Appearance [8]      | C3D      | 48.6       | 20.3       |
| Motion [8]          | C3D      | 57.8       | 29.9       |
| Motion & Appearance [8]| C3D   | 58.8       | 32.6       |
| Motion & Appearance* [8]| C3D | 61.2       | 33.4       |
| DPC [34]            | 3D Res-18| 60.6       | -          |
| Geometry [32]       | CaffeNet | 55.1       | 23.3       |
| CMC [35]            | Res-50   | 59.1       | 26.7       |
| Multi-Transforms (Ours) | C3D | **62.8**   | **34.2**   |
| Multi-Transforms (Ours) | 3D Res-18| **63.2**   | **35.9**   |

The first part includes the accuracy of both networks (C3D and 3D ResNet-18) that were trained from scratch on the UCF101 and HMDB51 datasets.

The second part shows the performance of state-of-the-art self-supervised methods regardless of backbone networks. All previous methods are pre-trained on different pretext tasks. When transferring to the downstream task (action recognition in this case), these models have different performance that depends on the features the models learned from the pretext task.

The third part is our method’s performance on C3D and 3D ResNet-18 networks. As can be seen in Table I, the accuracy of both networks is significantly improved compared with training from scratch (increase 16.7% and 14.5% on the UCF101 and HMDB51, respectively). This demonstrates that human action recognition can be significantly improved with self-supervised learning regardless of backbone networks, thanks to the contextual feature representation learned through the pretext tasks.

Also in Table I, we compare our results to state-of-the-art self-supervised methods using the RGB video data such as O3N ([5]), Video Jigsaw ([9]), Motion & Appearance* ([8]), DPC ([34]), Geometry ([32]), and so on. Compare to the existing approaches on various backbone networks and different pretext tasks, our proposed method obtains state-of-the-art performance in terms of accuracy on both datasets. In particular, comparing with several methods that used the Kinetics dataset in the pretext task, we achieved 62.8% on the UCF101 dataset. This result outperforms Video Jigsaw ([9]) and Motion & Appearance methods ([8]) by a margin of 2.0% with the same backbone.

C. Ablation Study

Fig. 3 shows the validation loss for each transformation at 100 epochs in the pretext task. As can be seen in Fig. 3, the transformation recognition network reaches a steady-state when learning the transformation tasks after 80 epochs. However, the different transformations reach their steady states at different epochs. Moreover, clear differences among the steady-state loss of the different transformations are observed, pointing to varied difficulties in the self-supervised training tasks.

![Fig. 3. Individual validation losses for each transformation and average loss (ave) versus epoch are presented for the video transformations recognition task.](Image)
As illustrated in Fig. 3, the combination of the proposed transforms aims to generalize the model better. Besides, compared to individual transform, the combination of seven transforms produces much more stable loss during validation.

| Method                     | UCF101  |
|----------------------------|---------|
| Frame rotation             | 52.2    |
| Color switching            | 47.1    |
| Noise addition             | 46.9    |
| Frame replacement          | 47.3    |
| Clip inversion             | 49.6    |
| Splitting and joining      | 46.6    |
| Permutation                | 51.8    |
| **Multi-Transforms (Ours)** | **62.8** |

To demonstrate the benefit from multi-transformations to action recognition, we conducted the ablation study on each transformation’s effectiveness and compared it to multi-transformations. The Table. II shows top-1 accuracy on the UCF101 dataset. Each transformation is pre-trained with the respective pretext task and then transfer to action recognition. We can see that multi-transformations’ performance is significantly improved (increase at least 10.6%) compared to single-transformation. With more transformations and many transformations being applied to an input video, it is tough for the model to predict which transformations are used. This is the motivator to help the model learn more spatio-temporal features based on transformations. Moreover, many transformations can be proposed for pseudo-label in the pretext task and data augmentation. This is one of the most advantages of the proposed multi-transformation method.

V. CONCLUSION

This paper introduced a novel approach for self-supervised spatio-temporal video representation learning by predicting a set of video transformations. The task is very suitable for 3D CNNs, which can model the spatio-temporal information. From the experimental results, we found that our method achieves state-of-the-art performance in self-supervised video action recognition on the UCF101 and HMDB51 datasets. Our method outperforms some methods that leverage the much larger-scale Kinetics dataset. These results demonstrate the efficacy of our proposed method to predict video transformations. We suggest our model as a powerful feature extractor for other tasks.

REFERENCES

[1] Ha Minh Tan, Duc-Quang Vu, and Jia-Ching Wang, “Selinet: a lightweight model for single channel speech separation,” in ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2023, pp. 1–5.

[2] Ha Minh Tan, Duc-Quang Vu, Duyen Nguyen Thi, and Trang Phung T Thu, “Voice separation using multi learning on squash-norm embedding matrix and mask,” in International Conference on Advances in Information and Communication Technology. Springer, 2023, pp. 327–333.

[3] Cao Hong Nga, Duc-Quang Vu, Huong Hoang Luong, Chien-Lin Huang, and Jia-Ching Wang, “Cyclic transfer learning for mandarin-english code-switching speech recognition,” IEEE Signal Processing Letters, 2023.

[4] Joao Carreira and Andrew Zisserman, “Quo vadis, action recognition? a new model and the kinetics dataset,” in CVPR. IEEE, 2017, pp. 6299–6308.

[5] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould, “Self-supervised video representation learning with odd-one-out networks,” in CVPR, 2017, pp. 3636–3645.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in CVPR. Ieee, 2009, pp. 248–255.

[7] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman, “A short note about kinetics-600,” arXiv preprint arXiv:1808.01340, 2018.

[8] Jiangliu Wang, Jianbo Jiao, Linao Bao, Shengfeng He, Yunhui Liu, and Wei Liu, “Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics,” in CVPR, 2019, pp. 4006–4015.

[9] Unaiza Ahsan, Rishi Madhok, and Irfan Essa, “Video jigsaw: Unsupervised learning of spatiotemporal context for video action recognition,” in WACV. IEEE, 2019, pp. 179–189.

[10] Ishan Misra, C Lawrence Zitnick, and Martial Hebert, “Shuffle and learn: unsupervised learning using temporal order verification,” in ECCV. Springer, 2016, pp. 527–544.

[11] Aalaeeldin El-Nouby, Shuangfei Zhai, Graham W Taylor, and Joshua M Susskind, “Skip-clip: self-supervised spatiotemporal representation learning by future clip order ranking,” arXiv preprint arXiv:1910.12770, 2019.

[12] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang, “Self-supervised spatiotemporal learning via video clip order prediction,” in CVPR, 2019, pp. 10334–10343.

[13] Richard Zhang, Phillip Isola, and Alexei A Efros, “Colorful image colorization,” in ECCV. Springer, 2016, pp. 649–666.

[14] Carl Doersch and Andrew Zisserman, “Multi-task self-supervised visual learning,” in CVPR, 2017, pp. 2051–2060.

[15] Pratam Sarkar and Ali Etemad, “Self-supervised learning for eeg-based emotion recognition,” in ICASSP. IEEE, 2020, pp. 3217–3221.

[16] Mirco Ravanelli, Jianyan Zhang, Santiago Pascual, Paweł Swietojanski, Joao Monteiro, Jan Trmal, and Yoshua Bengio, “Multi-task self-supervised learning for robust speech recognition,” in ICASSP. IEEE, 2020, pp. 6989–6993.

[17] Aaqib Saeed, Tanir Ozcelebi, and Johan Luukkien, “Multi-task self-supervised learning for human activity detection,” in Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol. 3, no. 2, pp. 1–30, 2019.

[18] Duc-Quang Vu, Ngan Le, and Jia-Ching Wang, “Teaching yourself: A self-knowledge distillation approach to action recognition,” IEEE Access, vol. 9, pp. 105711–105723, 2021.

[19] Duc-Quang Vu, Jia-Ching Wang, et al., “A novel self-knowledge distillation approach with siamese representation learning for action recognition,” in VCIP. IEEE, 2021, pp. 1–5.

[20] Duc-Quang Vu, Ngan TH Le, and Jia-Ching Wang, “(+ 1) d distilled shufflenet: A lightweight unsupervised distillation network for human action recognition,” in ICPR. IEEE, 2022, pp. 3197–3203.

[21] Piergiovanni AJ, Anelia Angelova, Alexander Toshov, and Michael S Ryoo, “Evolving space-time neural architectures for videos,” in Proceedings of the IEEE international conference on computer vision, 2019, pp. 1793–1802.

[22] Boyuan Jiang, MengMeng Wang, WeiKao Gan, Wei Wu, and Junjie Yan, “Sim: Spatiotemporal and motion encoding for action recognition,” in ICCV. IEEE, 2019, pp. 2000–2009.

[23] Thi Thu Trang Phung, Thi Hong Thu Ma, Duc Quang Vu, et al., “Self-supervised learning for action recognition by video denoising,” in IJRVF. IEEE, 2021, pp. 1–6.

[24] Ruben Villegas, Jimei Yang, Seunghoon Hong, Xianyu Lin, and Honglak Lee, “Decomposing motion and content for natural video sequence prediction,” arXiv preprint arXiv:1706.08033, 2017.
[25] Dahun Kim, Donghyeon Cho, and In So Kweon, “Self-supervised video representation learning with space-time cubic puzzles,” in AAAI, 2019, vol. 33, pp. 8545–8552.

[26] Uta Buchler, Biagio Brattoli, and Bjorn Ommer, “Improving spatiotemporal self-supervision by deep reinforcement learning,” in ECCV, 2018, pp. 770–786.

[27] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri, “Learning spatiotemporal features with 3d convolutional networks,” in CVPR, 2015, pp. 4489–4497.

[28] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh, “Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?,” in CVPR, 2018, pp. 6546–6555.

[29] Hildegard Kuehne, Hueihan Jhuang, Estibaliz Garrote, Tomaso Poggio, and Thomas Serre, “Hmdb: a large video database for human motion recognition,” in ICCV, IEEE, 2011, pp. 2556–2563.

[30] Khurram Soomro, Amir Roshan Zamir, and M Shah, “A dataset of 101 human action classes from videos in the wild,” Center for Research in Computer Vision, vol. 2, 2012.

[31] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang, “Unsupervised representation learning by sorting sequences,” in CVPR, 2017, pp. 667–676.

[32] Chuang Gan, Boqing Gong, Kun Liu, Hao Su, and Leonidas J Guibas, “Geometry guided convolutional neural networks for self-supervised video representation learning,” in CVPR, 2018, pp. 5589–5597.

[33] Nawid Sayed, Biagio Brattoli, and Björn Ommer, “Cross and learn: Cross-modal self-supervision,” in Pattern Recognition, Cham, 2019, pp. 228–243, Springer International Publishing.

[34] Tengda Han, Weidi Xie, and Andrew Zisserman, “Video representation learning by dense predictive coding,” in CVPRW, 2019, pp. 0–0.

[35] Yonglong Tian, Dilip Krishnan, and Phillip Isola, “Contrastive multiview coding,” 2020.