Inter-intra Variant Dual Representations for Self-supervised Video Recognition

Lin Zhang[13]
https://lzhangbj.github.io
Qi She†[3]
https://qi-she.net
Zhengyang Shen[23]
shenzhy@pku.edu.cn
Changhu Wang[3]
https://changhu.wang

[1] School of Computer Science
Carnegie Mellon University
Pittsburgh, USA

[2] School of Mathematical Science
Peking University
Beijing, China

[3] ByteDance AI Lab
Beijing, China

Abstract

Contrastive learning applied to self-supervised representation learning has seen a resurgence in deep models. In this paper, we find that existing contrastive learning based solutions for self-supervised video recognition focus on inter-variance encoding but ignore the intra-variance existing in clips within the same video. We thus propose to learn dual representations for each clip which (i) encode intra-variance through a shuffle-rank pretext task; (ii) encode inter-variance through a temporal coherent contrastive loss. Experiment results show that our method plays an essential role in balancing inter and intra variances and brings consistent performance gains on multiple backbones and contrastive learning frameworks. Integrated with SimCLR and pretrained on Kinetics-400, our method achieves **82.0%** and **51.2%** downstream classification accuracy on UCF101 and HMDB51 test sets respectively and **46.1%** video retrieval accuracy on UCF101, outperforming both pretext-task based and contrastive learning based counterparts. Our code is available at https://github.com/lzhangbj/DualVar.

1 Introduction

Labeled data is the fundamental resource in deep learning era however is laborious to acquire. As a result, researchers resort to self-supervised learning to utilize unlabeled data. Recent rapid development of self-supervised learning [5, 6, 10, 12] has been largely benefited from contrastive learning [41]. With InfoNCE loss [33], contrastive learning tries to pull examples from the same instance (positive pairs) close while repelling those from different instances (negative pairs). This has been largely used in image tasks for its effectiveness. Meanwhile, as the most important information source in daily life, video has been actively studied towards various research directions, such as architecture design [38], class incremental learning [37] and multi-model learning [39]. As a result, recent works tried to transplant it into video level [8, 24, 30], i.e. instance discrimination in the video level. Though having

† corresponding author

achieved remarkable performances, we challenge that such learning goal does not conform to the innate inter-intra variance of videos, thus is incomplete for video representation learning. Specifically, different clips sampled from different time spans of a video can exhibit different semantics. For instance, *running* and *jumping* are two different mini-actions though they are both sampled from a video classified as *HighJump*. As contrastive learning enforces features of clips sampled from a video to always be the same, the pretrained encoder is easily overfit to the pretrained dataset. As a result, feature distributions of videos are sparse (supplementary Figure 2) and instance discrimination ability is over strong (Table 2). By considering such intra-variance, pretrained encoder can be more generalizable to downstream tasks thus wins a better transferring ability. Previous works [9, 18, 22, 40, 43] have proposed to learn such temporal differences using frame/clip order verification tasks. However, the order of sub-clips is largely determined by continuity instead of semantic difference between sub-clips and is ambiguous in some repetitive actions.

In this work, we delve into self-supervised video representation learning from the perspective of inter-intra variance encoding. We theoretically and experimentally find out that contrastive learning [33] overemphasizes the learning of inter-variance but ignores intra-variance. Previous works [23, 35] tried to separately encode inter and intra variances by appending an extra projection head to solely solve a pretext task. In contrast, we learn dual representations for each clip, which joinly encodes intra-variance between sub-clips by a shuffle-rank pretext task and inter-variance between videos by a temporal coherent contrastive loss. Besides, we adopt a ranking loss to induce a small margin between sub-clip features to reduce interference between inter and intra variance encoding. We verify its effectiveness by a series of experiments on UCF101 [27] and HMDB51 [17] datasets and show that our method can balance inter-intra feature variances (Table 2) and achieve superior performances on both finetuning and video retrieval task to state-of-the-arts.

In a nutshell, our contributions are 4-fold (i) We propose a shuffle-rank pretext task, which induces a small margin between intra-variant features, alleviating contradiction between inter-video and intra-video discrimination. (ii) We propose temporal coherent contrast on the dual representations to model inter variance learning. (iii) We learn joint inter-intra variant dual representations as opposed to solely inter-variant representation in contrastive learning. We also conduct a series of experiments to validate the effectiveness of our method on inter-intra variance encoding. (iv) The proposed method can be flexibly applied to contrastive learning frameworks, e.g. MoCo and SimCLR, with multiple spatial-temporal backbones and achieves superior performances to state-of-the-art methods.

2 Related work

2.1 Self-supervised video recognition

We classify existing self-supervised video recognition methods into two categories based on type of the supervision signal enforced: pretext task based and contrastive learning based.

Pretext tasks Pretext task based solutions design handcrafted tasks to solve. Verifying frame and clip order [9, 18, 22, 40, 43] can provide useful order information for downstream transferring and is proved to be effective. Utilizing spatial and temporal information [14, 21, 34] has also achieved remarkable performances. For example, Wang et al. [34] proposed to learn spatial-temporal features by designing multiple spatial-temporal statistics prediction tasks which however introduces more complexity. Recently, exploring speedness

in videos have become very popular [23, 25, 26, 35, 44] and also achieved state-of-the-art performance [23]. In this paper, we also propose a shuffle-rank pretext task to encode temporal intra-variance between sub-clips. It is essentially different from order verification in that (i) We aim to learn a variety of intra-variance between clips by comparing sub-clip feature similarities instead of simply predicting clip order, which is ambiguous when clip changes are small; (ii) Unlike order classification, our ranking loss induces a small margin between sub-clip features, alleviating contradiction between inter and intra variance encoding.

Contrastive learning Contrastive learning tries to distinguish same instances from different ones. MoCo [12] designed a negative queue to store more negatives. SimCLR [6] proved that large batchsize is crucial to achieve superior performance. On the video level, based on MoCo, Tian et al. [30] built a temporarily decayed negative queue to model temporal variance. Rui et al. [24] conducted sufficient experiments to study video-level SimCLR's performance. Kong et al. [16] learned feature proximity between video and frame features. Feichtenhofer et al. [8] systematically analyzed four self-supervised learning frameworks on videos. However, all these contrastive learning methods aim to learn inter-video variance and intra-video invariance. Differently, Tao et al. [28] proposed an inter-intra contrastive learning framework by creating different positive and negative pairs but achieved little performance gains. In contrast, our work makes use of dual features to encode temporal differences between sub-clips and jointly utilized pretext tasks to achieve much better performance.

2.2 Intra-class and inter-class variance

Balancing inter and intra class variance has been a critical research field in various areas. Bai et al. [3] leveraged intra-class variance in metric learning to improve the performance of fine-grained image recognition. Liu et al. [19] found out that negative margins in softmax loss results in lower intra-class variance and higher inter-class variance for novel classes in few shot image classification. To alleviate long-tailed distribution, Liu et al. [20] proposed to increase intra-variance of tail classes by augmenting it with feature distributions of head classes. In this paper, we instead treat each video as an individual class and jointly encode instance-wise intra and inter variances in unlabeled videos. We experimentally study the effect of our model on inter-intra variance learning in section 5.3.

2.3 Ranking measure

Approximating ranking measures using functions has been studied by multiple previous works. Burges et al. [4] investigated using gradient descent methods to approximate ranking functions and proposed RankNet for pairwise ranking. Chen et al. [7] concluded that an essential loss is both an upper bound of the measure-based ranking errors and a lower bound of the loss functions. Recently, Andrew et al. [2] approximated ranking-based metric (Average Precision) using logistic functions and proposed Smooth-AP. Ali et al. [1] further applied such idea into self-supervised learning, formulating it as a ranking problem. In this work, we also use logistic function for ranking approximation. Differently, we propose to rank sub-clip features for sub-clip discrimination thus learn the intra clip variances.

3 Preliminary

In this section, we first introduce inter-intra variances in video data, and then explain the disadvantage of video contrastive learning which only encodes inter-variance. This leads to our motivation of learning inter-intra variant dual representations in section 4.

3.1 Video data distribution with inter-intra variances

Suppose we have a collection of N unlabeled videos $\{V_i\}_{i=1}^{N}$. Limited by memory, we sample a total of M clips $\{c_i\}_{i=1}^{M}$ from videos, with $\frac{M}{N}$ clips per video. During self-supervised pretraining, a clip c_i is sampled and encoded by our model f into a normalized feature vector z_i, i.e. $z_i = f(c_i)$. The goal of self-supervised pretraining is to learn a good encoder f that can be well transferred to downstream video action recognition. Therefore, firstly, we should distinguish different videos based on their very different contents, which is characterized as inter-variance (σ_{inter}). Secondly, semantics of clips from the same video vary a lot, e.g. *running* and *jumping* are two different mini-actions at different time spans of a video classified as *HighJump*. Our motivation is that an encoder learning on both clip-level and sub-clip-level has a more generalized transfer ability in downstream tasks. We thus aim to produce inter and intra variant embedded features $\{z\}$.

3.2 Self-supervised contrastive representation learning

Contrastive learning expects clips from the same video to attract each other and repel those from different videos. Formally, the clip-feature based contrastive loss is denoted as :

$$L_c = -\frac{1}{M}\sum_{i=1}^{M}\log\frac{\exp(z_i \cdot z_{i^+}/\tau)}{\sum_{k=1}^{M} 1_{[k\neq i]}\exp(z_i \cdot z_k/\tau)} \tag{1}$$

where z_{i^+} is a positive $(+)$ clip feature sampled from the same video of z_i and τ is a temperature parameter. We can easily extend the analysis in [31] to find that such contrastive learning has an objective of persistently increasing σ_{inter} and decreasing σ_{intra}, leading to insignificant intra-variance (see supplementary section 6). In this work, we propose a shuffle-rank pretext task to compensate for lack of intra-variance and a temporal coherent contrast loss between sub-clip representations to encode σ_{inter}.

4 Methodology

4.1 Dual representations

In contrastive learning, an n-frame clip is typically encoded into a single feature for representation without considering contrast between the inner sub-clips. We instead use dual feature vectors for representing two halves of the input clip (Figure 1).

Formally, in addition to the original clip projector, we add another projection head, denoted as dual projection head. A sampled clip c is projected by the dual projection head into dual features $r = (q^1, q^2)$. Our goal is then to jointly encode the inter and intra variances into r, i.e. differences between two sub-clips of c and between c and other videos.

4.2 Shuffle-rank

To encode differences between two sub-clips into dual representations, we propose a shuffle-rank pretext task to align raw sub-clips and learned dual features. In this section, we first

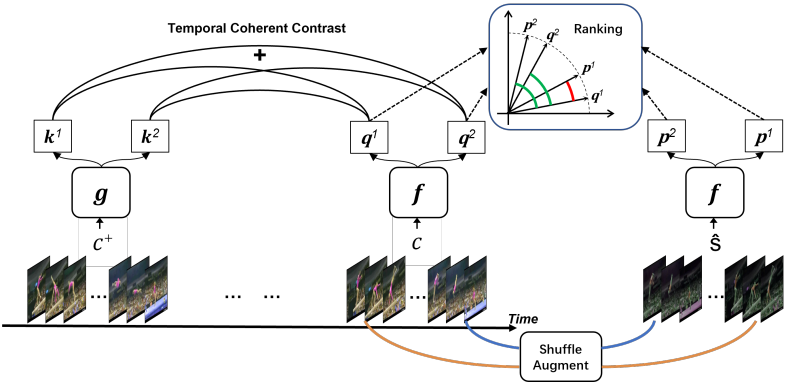


Figure 1: Method Overview. During the intra-variance learning stage, we shuffle and augment clip c into $\hat{s}$ and encode it into dual representations p^1 and p^2, which correspond to the unshuffled dual representations q^1 and q^2 respectively. If we treat q^1 as anchor, then representation of the same sub-clip (Red) should be ranked higher than that of different sub-clips (Green). During the inter-variance learning stage (left-hand side), the temporarily variant dual representations should however keep coherence in that all dual representations of clips sampled from the same video (c, c^+) should be regarded as positive pairs (+) in contrastive learning. Encoder g is momentum updated by f in MoCo but equals to f in SimCLR.

describe our method, then explain the differences between our method and order prediction, which refers to simply predicting sequential order of clips and has been extensively studied before [14, 22, 43].

Overall, shuffle-rank consists of two stages, sub-clip shuffling and representation ranking. In the sub-clip shuffling stage, the sub-clips of an input clip c is shuffled and augmented into $\hat{s}$. Both clips will then be projected into dual representations for sub-clips. In the representation ranking stage, dual features of c and $\hat{s}$ will be pairwisely ranked to achieve correspondence between sub-clips and dual features through a ranking loss. As a result, the predicted dual features can genuinely reflect the intra-variance between sub-clips.

Sub-clip shuffling We first uniformly divide clip c into two sub-clips c^1 and c^2. By applying data augmentation on c, we get a new augmented clip $\hat{c} = (\hat{c}^1, \hat{c}^2)$, where *hat* refers to augmentation. We further shuffle $\hat{c}$ to get its shuffled version $\hat{s} = (\hat{c}^2, \hat{c}^1)$. Both c and $\hat{s}$ are then projected into dual representations r and $\hat{r}^s$ through the dual projection head.

$$c = (c^1, c^2) \tag{2}$$

$$\hat{c} = (\hat{c}^1, \hat{c}^2) = augment(c^1, c^2), \quad \hat{s} = (\hat{c}^2, \hat{c}^1) = shuffle(\hat{c}) \tag{3}$$

$$r = (q^1, q^2) = f(c), \quad \hat{r}^s = (p^2, p^1) = f(\hat{s}) \tag{4}$$

where f is the encoding function containing a backbone and a projection layer.

Representation ranking Shuffle-rank alone can not guarantee the correspondence between sub-clip and dual features. Therefore, we apply a ranking measure [4, 7] to learn the subtle temporal intra-variances between sub-clips by enforcing the sub-clip feature correspondence, i.e. $\{c^1, \hat{c}^1\} \Rightarrow \{q^1, p^1\}$ and $\{c^2, \hat{c}^2\} \Rightarrow \{q^2, p^2\}$. Formally, if we regard q^1 as anchor, then p^1 should be ranked before both q^2 and p^2 while the ranking between q^2 and p^2 is unknown, as shown in Figure 1. Penalties should be heavily imposed when such ranking is wrong and stay zero when the ranking is correct. However, directly applying such discrete loss would harm the stability of training. In order to have a smoother gradient backpropagation, we adopt the logistic loss function [4]. For a sub feature $x \in \{q^1, p^1, q^2, p^2\}$, we denote

its leave-self-out set of dual representations as x^+ and its unpaired representation set as x^-:

$$q^{1+} = \{q^1, p^1\} \setminus \{q^1\} = \{p^1\}, \quad q^{1-} = \{q^2, p^2\} \tag{5}$$

$$q^{2+} = \{q^2, p^2\} \setminus \{q^2\} = \{p^2\}, \quad q^{2-} = \{q^1, p^1\} \tag{6}$$

$$p^{1+} = \{q^1, p^1\} \setminus \{p^1\} = \{q^1\}, \quad p^{1-} = \{p^2, q^2\} \tag{7}$$

$$p^{2+} = \{q^2, p^2\} \setminus \{p^2\} = \{q^2\}, \quad p^{2-} = \{p^1, q^1\} \tag{8}$$

Let S be a function mapping two clips to their dual features set, i.e. $S(c_i, \hat{s}_i) = \{q_i^1, q_i^2, p_i^1, p_i^2\}$, then the ranking loss between *unaugmented* original clips $\{c_i\}$ and their shuffled and augmented clips $\{\hat{s}_i\}$ is:

$$\mathcal{L}_{rank}^{unaug} = \sum_{i=1}^{M} \sum_{x \in S(c_i, \hat{s}_i)} \sum_{y \in x^+, z \in x^-} \log(1 + \exp(\frac{\text{sim}(x,z) - \text{sim}(x,y)}{\theta})) \tag{9}$$

where θ is a temperature parameter. In practice, for augmentation, we also compute ranking loss between *augmented* clips $\{\hat{c}_i\}$ and $\{\hat{s}_i\}$, denoted as $\mathcal{L}_{rank}^{aug}$:

$$\mathcal{L}_{rank}^{aug} = \sum_{i=1}^{M} \sum_{x \in S(\hat{c}_i, \hat{s}_i)} \sum_{y \in x^+, z \in x^-} \log(1 + \exp(\frac{\text{sim}(x,z) - \text{sim}(x,y)}{\theta})) \tag{10}$$

Final ranking loss $\mathcal{L}_{rank} = 0.5 * \mathcal{L}_{rank}^{unaug} + 0.5 * \mathcal{L}_{rank}^{aug}$. In Figure 1, we only demonstrate the computing of $\mathcal{L}_{rank}^{unaug}$ for simplicity.

The adopted ranking loss is advantageous over order prediction in two aspects: (i) Order only reflects very little information of intra-video variance, whereas in our case, by comparing the pairwise similarities between sub-clip representations, a larger variety of intra-variance can be encoded. (ii) Softmax cross entropy loss based order prediction induces large margin between intra-video features [15], thus decreases the margin between inter-video features and disturbs inter-variance encoding. Instead, ranking loss only requires a small margin between similarity of positive intra pairs (x and $y \in x^+$) and negative intra pairs (x and $z \in x^-$). Such a loss is also safer since sub-clip differences vary a lot from video to video, e.g. frames in a *Typing* video seldom changes, exhibiting smaller intra-variance, while frames in a *ClipDiving* change very fast, resulting in large intra-variance.

In section 5.3, we compare our shuffle-rank task with a common order prediction task. We also show that the temperature θ plays an important role in modeling such ranking effect and brings obvious improvement when θ is small enough.

4.3 Temporal coherent contrastive learning

We want to further encode the inter-variance into the dual features. To do so, coherence between dual features should be maintained in that dual features from clips in the same video should be closer to each other in feature space than those from different videos, since inter-variance is much larger than intra-variance. We thus extend clip contrast to temporal coherent contrast by using sub-clip similarity instead of clip similarity. In particular, we denote similarity between two dual representations r_i and r_j as $\text{tc-sim}(r_i, r_j) = \frac{1}{4} \sum_{x \in r_i, y \in r_j} x \cdot y$ where r_i and r_j correspond to clips c_i and c_j respectively. Then the temporal coherent contrastive loss is written as:

$$\mathcal{L}_{tc} = -\frac{1}{M} \sum_{i=1}^{M} \log \frac{\exp(\text{tc-sim}(r_i, r_{i^+})/\tau_{tc})}{\sum_{k=1}^{M} 1_{[k \neq i]} \exp(\text{tc-sim}(r_i, r_k)/\tau_{tc})} \tag{11}$$

where τ_{tc} is a temperature parameter and i^+ indexes the i-th clip's positive pair. Though simple, the temporal coherent contrastive learning further increases the inter-instance variances and instance discrimination ability of self-supervised learned models, and consistently

improves the performance upon intra-variance encoded representations, as Table 1 shows.

Our final loss is the sum of clip contrastive loss, ranking loss and temporal coherent contrastive loss $\mathcal{L} = \mathcal{L}_c + \lambda_1 * \mathcal{L}_{rank} + \lambda_2 * \mathcal{L}_{tc}$, where λ_1 and λ_2 are hyperparameters.

5 Experiments

We conduct experiments on two contrastive learning frameworks (MoCo [12], SimCLR [6]) and three backbones (R3D [32], R(2+1)D [32], S3D-G [42]). We apply our method in pretraining stage and evaluate performance on two tasks: finetuning and video retrieval.

5.1 Datasets

Kinetics400 Kinetics400 [13] is a large-scale video action dataset with 400 classes and more than 400 videos for each class. All the videos are clips from Youtube and persist around 10 seconds. We are only able to obtain 218,846 videos due to invalid links.

UCF101 UCF101 [27] is a medium-scale human action video dataset with 13,320 videos classified into 101 classes. All the videos have a fixed frame rate of 25 FPS and a resolution of 320× 240. It provides 3 train-test splits. We use split 1 in all our experiments.

HMDB51 HMDB51 [17] is a human action video dataset with 6849 videos in 51 classes. The videos are scaled to a height of 240 pixels and 30 FPS. We use its split 1 in experiments.

5.2 Implementations

We briefly introduce implementations and provide more details in supplementary section 1.

Self-supervised pretrain In self-supervised pretraining stage, we randomly resize and crop clips to size of $16 \times 112 \times 112$ in a temporal consistent way with temporal stride of 4. Color jittering, horizontal flipping and gaussian blurring are applied. We pretrain the model for 200 epochs with an SGD optimizer with an initial learning rate of 0.003 and batch size of 64 on 8 Tesla V100 GPUs. We pretrain on UCF101 training split in ablation study and on large-scale Kinetics400 for performance comparison with counterparts. We set τ, τ_{tc}, θ, λ_1, λ_2 to 0.07, 0.5, 0.05, 1.0 and 1.0, respectively.

Supervised finetuning We replace the nonlinear projection head during pretraining with a classification linear layer and initialize the backbone with the pretrained weights. We finetune all layers for 150 epochs on UCF101 and HMDB51 training splits with a batchsize of 64 and learning rate of 0.05. We then test classification accuracy on test splits.

Video retrieval To evaluate the representation ability of pretrained model, we use videos in test set to retrieve videos in training set. Specifically, we average features of 10 clips uniformly sampled from each video using the pretrained backbone. We conduct video retrieval on UCF101 and calculate the top-k accuracy ($k = 1, 5, 10, 20, 50$).

5.3 Ablation study

Effectiveness of proposed method We first show the effectiveness of our method by conducting experiments on both MoCo and SimCLR frameworks and three spatio-temporal

	R3D		R(2+1)D		S3D-G	
	UCF101	HMDB51	UCF101	HMDB51	UCF101	HMDB51
MoCo	71.72	41.04	77.64	45.70	68.41	38.08
MoCo+SR	74.28$^{+2.56}$	44.06$^{+3.02}$	78.67$^{+1.03}$	46.09$^{+0.39}$	70.79$^{+2.38}$	40.12$^{+2.04}$
MoCo+SR+TC	74.65$^{+2.93}$	44.45$^{+3.41}$	78.46$^{+0.82}$	47.47$^{+1.77}$	72.19$^{+3.78}$	41.56$^{+3.48}$
SimCLR	71.90	39.79	71.61	31.78	66.27	20.16
SimCLR+SR	72.24$^{+0.34}$	40.38$^{+0.59}$	76.82$^{+5.21}$	40.05$^{+8.27}$	70.82$^{+4.55}$	34.73$^{+14.57}$
SimCLR+SR+TC	72.69$^{+0.79}$	43.01$^{+3.32}$	79.01$^{+7.40}$	45.37$^{+13.59}$	71.32$^{+5.05}$	35.33$^{+15.17}$

Table 1: Experiments on MoCo and SimCLR with R3D, R(2+1)D and S3D-G backbones. Models are pretrained on UCF101 train split 1. SR refers to shuffle-rank. TC refers to temporal coherent contrast. Improvement upon baseline is marked as Red superscripts.

backbones R3D, R(2+1)D and S3D-G. In Table 1, consistent performance gains on multiple backbones can be observed. On SimCLR with R(2+1)D, shuffle-rank increases baseline accuracy on UCF101 and HMDB51 by 5.21% and 8.27% while the integrated method increases it by 7.40% and 13.59% respectively. It can be observed that performance improvements differ on different backbones, which might be due to both the internal structure of architecture and baseline performance, e.g. a strong baseline performance means smaller space for improvement. However, even on a pretty strong baseline such as MoCo with R(2+1)D backbone, our integrated method can still improve accuracy on UCF101 and HMDB51 by 0.82% and 1.77% respectively. Moreover, as the model is pretrained on UCF101, improvement is generally larger on HMDB51, e.g. 15.17% versus 5.05% with S3D-G and 13.59% versus 7.40% with R(2+1)D on SimCLR, verifying our model is more generalizable.

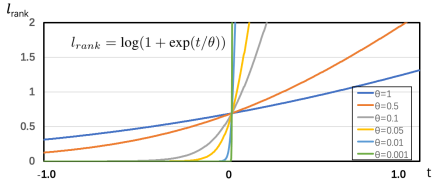
Effect on inter and intra variance encoding To analyze the effect of our method on variance encoding, we explicitly calculate inter-intra variance of video features produced by pretrained backbone on UCF101 test set. Specifically, we uniformly sample 10 clips for each video temporarily, then calculate σ_{inter}, σ_{intra} and instance discrimination factor $\sigma_{inter}/\sigma_{intra}$ according to the formulas defined in supplementary section 5. As shown in Table 2, shuffle-rank can always increase σ_{intra} by a large margin, e.g. 14 times on R(2+1)D from 0.0084 to 0.1123. After further adding temporal coherent contrast, σ_{inter} is increased to 0.0797 and σ_{intra} is decreased to 0.3798. Our method balances the instance discrimination ability from a super high level 33.60 to a medium value 4.77. This general phenomenon on all three backbones (R3D, R(2+1)D, S3D-G) verifies our motivation, i.e. using shuffle-rank to encode intra-variance and temporal coherent contrast to strengthen inter-variance encoding. It also supports our statement in section 3 that encoding intra-variance can be beneficial. More experiment results on HMDB51 dataset can be see in supplementary section 3.

Comparison to shuffling order prediction Following our discussion in Section 4.2, we compare shuffle-rank to a non-trivial shuffling order prediction baseline. Our method is more friendly in encoding inter-intra variances by using ranking loss. Our finetuning accuracy improves upon order prediction baseline from 75.39% and 32.38% to 76.82% and 40.05% on UCF101 and HMDB51 respectively. We report details in the supplementary section 2.

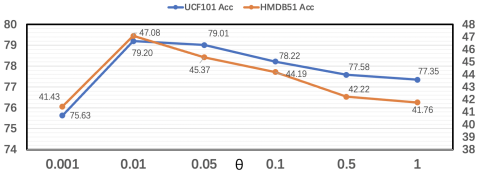
Effect of ranking loss parameter θ Following our discussion in Section 4.2, in Figure 2, we validate our statement that inducing a smaller margin between intra positive and neg-

	R3D			R(2+1)D			S3D-G		
	inter-v.	intra-v.	discrim.	inter-v.	intra-v.	discrim.	inter-v.	intra-v.	discrim.
SimCLR	28.8	0.8	34.3	28.2	0.8	33.6	73.1	3.0	24.7
+SR	22.6	11.3$^{\uparrow 13.6\times}$	2.0$^{\downarrow 17.2\times}$	23.0	11.2$^{\uparrow 13.4\times}$	2.0$^{\downarrow 16.8\times}$	51.7	11.7$^{\uparrow 4.0\times}$	4.4$^{\downarrow 5.6\times}$
+SR+TC	42.3	5.8$^{\downarrow 1.9\times}$	7.3$^{\uparrow 3.7\times}$	38.0	8.0$^{\downarrow 1.4\times}$	4.8$^{\uparrow 2.4\times}$	78.7	4.9$^{\downarrow 2.4\times}$	16.2$^{\uparrow 3.7\times}$

Table 2: Comparison of inter-instance variance, intra-instance variance and instance discrimination factor on UCF101 test set. Results are multiplied by 100 for demonstration. Each cell's increasing (Red) and decreasing (Green) times are compared to the cell above it.



(a) Graph of ranking loss. t is the difference between negative and positive sub-clip pairs, i.e. $t = \text{sim}(x,z) - \text{sim}(x,y)$, where $z \in x^-$ and $y \in x^+$.



(b) Finetuning accuracies on UCF101 and HMDB51 test set.

Figure 2: Under different θ values, we (a) plot ranking loss graph (b) investigate effect of θ on downstream classification accuracy. Decreasing θ induces a smaller margin ($|t|$) under the same l_{rank} and improves finetuning performance.

ative pairs brings larger benefits. In Figure 2 (a), we plot ranking loss (l_{rank}) graph under different θ. When the difference between similarities of intra negative and positive pairs ($t = \text{sim}(x,z) - \text{sim}(x,y)$, where $z \in x^-$ and $y \in x^+$) is zero, a fixed penalty of $log2$ is enforced as the representation is not discriminative on intra-variance. Definition of x^+ and x^- is in section 4.2. As θ becomes larger, derivative at $t = 0$ keeps increasing and the penalty quickly increases when the ranking measure is wrong ($t > 0$) and decreases when it is correct. Besides, when the ranking is correct, penalties enforced are close to zero as long as t is smaller than a margin value that is monotonically increasing with θ. As shown in Figure 2 (b), as θ decreases from 1.0 to 0.01, model performance keeps increasing, validating our hypothesis that a small enough margin is more beneficial. However, when θ is too small as 0.001, l_{rank} is too sensitive at $t = 0$, leading to unstable training. One thing need to mention here is that a smaller θ (0.01) can further increase our reported performances under $\theta = 0.05$.

Performance comparison We compare our method with previous works on both supervised finetuning and video retrieval tasks. In Table 3, we classify previous methods into 3 categories. Hybrid means combination of pretext tasks and contrastive learning. We do not compare to recent methods [8, 24, 29] as they use either much larger backbones and input sizes or optical flow. We outperform methods based on two mainstream pretext tasks: temporal order [18, 22, 43] and pace [25, 26, 35]. SpeedNet [25] and TempTrans [26] achieved superior performance due to large input size or backbones. MemDPC [11] predicted future states and applies spatial-temporal contrastive loss on features however relies on huge input size. Our model achieves higher performance than MoCo based method BE [36] and VideoMoCo [30]. Our model surpasses RSPNet [23], which is the state-of-the-art improving upon Pace [35] by predicting relative speedness, by 0.9% and 6.6% on UCF101 and HMDB51 test set, respectively. Even pretrained on much smaller UCF101 training data, our model still exhibits excellent performance with 0.8% higher HMDB51 accuracy upon Kinetics400

Method	Input Size	Arch	#param.	pretrain	UCF101	HMDB51
Pretext Task						
Shuffle&Learn[22]	$3 \times 256 \times 256$	AlexNet	58.3M	UCF101	50.2	18.1
OPN[18]	$4 \times 80 \times 80$	VGG	8.6M	UCF101	59.8	23.8
VCP[21]	$16 \times 112 \times 112$	R(2+1)D	14.4M	UCF101	66.3	32.2
VCOP[43]	$16 \times 112 \times 112$	R(2+1)D	14.4M	UCF101	72.4	30.9
PRP[44]	$16 \times 112 \times 112$	R(2+1)D	14.4M	UCF101	72.1	35.0
SpeedNet[25]	$64 \times 224 \times 224$	S3D-G	9.6M	K400	81.1	48.8
TempTrans[26]	$16 \times 112 \times 112$	R(2+1)D-18	33.2M	UCF101	81.6	46.4
Contrastive						
MemDPC[11]	$40 \times 224 \times 224$	3D-ResNet34	32.4M	K400	78.1	41.2
VideoMoCo[30]	$32 \times 112 \times 112$	R(2+1)D	14.4M	K400	78.7	49.2
BE(MoCo)[36]	$16 \times 112 \times 112$	C3D	27.7M	UCF101	72.4	42.3
IIC[28]	$16 \times 112 \times 112$	R3D	14.4M	UCF101	74.4	38.3
Hybrid						
Pace[35]	$16 \times 112 \times 112$	R(2+1)D	14.4M	K400	77.1	36.6
RSPNet[23]	$16 \times 112 \times 112$	R(2+1)D	14.4M	K400	81.1	44.6
Ours(MoCo)	$16 \times 112 \times 112$	R(2+1)D	14.4M	UCF101	78.5	47.5
Ours(SimCLR)	$16 \times 112 \times 112$	R(2+1)D	14.4M	UCF101	79.0	45.4
Ours(SimCLR)	$16 \times 112 \times 112$	R(2+1)D	14.4M	K400	**82.0**	**51.2**

Table 3: Finetuning performance comparison.

	Arch	Top-k				
		k=1	k=5	k=10	k=20	k=50
PRP[44]	C3D	23.2	38.1	46.0	55.7	68.4
Pace[35]	R(2+1)D	25.6	42.7	51.3	61.3	74.0
TempTrans[26]	3D-ResNet18	26.1	48.5	59.1	69.6	82.8
RSPNet[23]	3D-ResNet18	41.1	59.4	68.4	77.8	**88.7**
Ours	R(2+1)D	**46.7**	**63.1**	**69.7**	**78.0**	87.8

Table 4: Video retrieval performance comparison.

pretrained RSPNet. On video retrieval task in Table 4, our method also exhibit robust performance. Our top-1 retrieval accuracy reaches 46.7%, improving upon RSPNet by 5.6%. This shows our model has a well learned discrimination ability.

6 Conclusion

In this paper, we approach self-supervised video representation learning from the perspective of inter-intra variance. We find that existing contrastive learning solution over-learns instance discrimination ability on pretrained dataset, thus has difficulty in generalization. Therefore, we propose to learn dual representations which encodes inter-intra variancesby a shuffle-rank pretext task and a temporal coherent contrast that wins a higher transferring power. It surpasses both pretext-task based and contrastive learning based counterparts on classification and video retrieval tasks on UCF101 and HMDB51 dataset.

References

[1] Varamesh Ali, Diba Ali, Tuytelaars Tinne, and Luc Van Gool. Self-supervised ranking for representation learning. In *(NeurIPS Workshop*, 2020.

[2] Brown Andrew, Xie Weidi, Kalogeiton Vicky, and Zisserman Andrew. Smooth-ap: Smoothing the path towards large-scale image retrieval. In *ECCV*, 2020.

[3] Yan Bai, Feng Gao, Yihang Lou, Shiqi Wang, Tiejun Huang, and Ling-Yu Duan. Incorporating intra-class variance to fine-grained visual recognition. In *ICME*, 2018.

[4] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *ICML*, 2005.

[5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[7] Wei Chen, Tie-yan Liu, Yanyan Lan, Zhi-ming Ma, and Hang Li. Ranking measures and loss functions in learning to rank. In *NeurIPS*, 2009.

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

[9] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, 2017.

[10] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *arxiv:2006.07733*, 2020.

[11] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *ECCV*, 2020.

[12] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[13] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *arXiv:1705.06950*, 2017.

[14] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2018.

[15] Takumi Kobayashi. Large margin in softmax cross-entropy loss. In *BMVC*, 2019.

[16] Quan Kong, Wenpeng Wei, Ziwei Deng, Tomoaki Yoshinaga, and Tomokazu Murakami. Cycle-contrast for self-supervised video representation learning. In *NeurIPS*, 2020.

[17] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *ICCV*, 2011.

[18] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequence. In *ICCV*, 2017.

[19] Bin Liu, Yue Cao, Yutong Lin, Qi Li, Zheng Zhang, Mingsheng Long, and Han Hu. Negative margin matters: Understanding margin in few-shot classification. In *ECCV*, 2020.

[20] Jialun Liu, Yifan Sun, Chuchu Han, Zhaopeng Dou, and Wenhui Li. Deep representation learning on long-tailed data: A learnable embedding augmentation perspective. In *CVPR*, 2020.

[21] Dezhao Luo, Chang Liu, Yu Zhou, Dongbao Yang, Can Ma, Qixiang Ye, and Weiping Wang. Video cloze procedure for self-supervised spatio-temporal learning. In *AAAI*, 2020.

[22] Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. Shuffle and learn: Unsupervised learning using temporal order verification. In *ECCV*, 2016.

[23] Chen Peihao, Huang Deng, He Dongliang, Long Xiang, Zeng Runhao, Wen Shilei, Tan Mingkui, and Gan Chuang. Rspnet: Relative speed perception for unsupervised video representation learning. In *AAAI*, 2021.

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

[25] Benaim Sagie, Ephrat Ariel, Lang Oran, Mosseri Inbar, William T. Freeman, Rubinstein Michael, Irani Michal, and Dekel Tali. Speednet: Learning the speediness in videos. In *CVPR*, 2020.

[26] Jenni Simon, Meishvili Givi, and Favaro Paolo. Video representation learning by recognizing temporal transformations. In *ECCV*, 2020.

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

[28] Li Tao, Xueting Wang, and Toshihiko Yamasaki. Self-supervised video representation learning using inter-intra contrastive framework. In *ACMMM*, 2020.

[29] Han Tengda, Xie Weidi, and Zisserman Andrew. Coclr: Self-supervised co-training for video representation learning. In *NeurIPS*, 2020.

[30] Pan Tian, Song Yibing, Yang Tianyu, Jiang Wenhao, and Liu Wei. Videomoco: Contrastive video representation learning with temporally adversarial examples. In *CVPR*, 2021.

[31] Wang Tongzhou and Isola Phillip. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, 2020.

[32] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.

[33] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2019.

[34] Jiangliu Wang, Jianbo Jiao, Linchao 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.

[35] Jiangliu Wang, Jianbo Jiao, and Yunhui Liu. Self-supervised video representation learning by pace prediction. In *ECCV*, 2020.

[36] Jinpeng Wang, Yuting Gao, Ke Li, Yiqi Lin, Andy J Ma, Hao Cheng, Pai Peng, Rongrong Ji, and Xing Sun. Removing the background by adding the background: Towards background robust self-supervised video representation learning. In *CVPR*, 2021.

[37] Zhengwei Wang, Qi She, Tejo Chalasani, and Aljosa Smolic. Catnet: Class incremental 3d convnets for lifelong egocentric gesture recognition. 2020.

[38] Zhengwei Wang, Qi She, and Aljosa Smolic. Action-net: Multipath excitation for action recognition. In *CVPR*, 2021.

[39] Zhengwei Wang, Qi She, and Aljosa Smolic. Team-net: Multi-modal learning for video action recognition with partial decoding. In *BMVC*, 2021.

[40] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *CVPR*, 2018.

[41] Zhirong Wu, Yuanjun Xiong, X Yu Stella, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.

[42] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *ECCV*, 2018.

[43] 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.

[44] Yuan Yao, Chang Liu, Dezhao Luo, Yu Zhou, and Qixiang Ye. Video playback rate perception for self-supervised spatio-temporal representation learning. In *CVPR*, 2020.