# LOOK GLOBALLY AND LOCALLY: INTER-INTRA CONTRASTIVE LEARNING FROM UNLABELED VIDEOS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

State-of-the-art video contrastive learning methods spatiotemporally augment two clips from the same video as positives. By only sampling positive clips from the same video, these methods neglect other semantically related videos that can also be useful. To address this limitation, we leverage nearest-neighbor videos from the global space as additional positives, thus improving diversity and introducing a more relaxed notion of similarity that extends beyond video and even class boundaries. Our "Inter-Intra Video Contrastive Learning" (IIVCL) improves performance and generalization on video classification, detection, and retrieval tasks.

## 1 INTRODUCTION

Recently, contrastive learning works for video such as CVRL (Qian et al., 2021) and $\rho$-MoCo (Feichtenhofer et al., 2021) are competitive with supervised learning. These works learn a representation from unlabeled data by pulling positive pairs closer and pushing negative samples apart in the embedding space. For video, these positive pairs are generated through random augmentations of sub-clips from the *same* video, while clips from other similar videos are *never* used as positives.

By only considering clips that belong to the same video to be positive, works such as CVRL and $\rho$-MoCo neglect other semantically related videos that may also be useful and relevant as positives for contrastive learning. For example, consider a positive pair of two skiing videos and a negative snowboarding video. Snowboarding is semantically related but always will be negative, so this similarity will never be leveraged as an additional signal to the skiing-skiing positive pair.

This raises the question of what constitutes a desirable video representation; by focusing too much on local intra-video semantics, we may miss the larger picture and hierarchy of visual concepts. This might lead to overfitting to tasks that are similar to the pretraining dataset and thus hurt generalization. On the other hand, if we focus too much on global inter-video semantics, we may lose sight of granular details that are also important for video understanding. To balance the two, we propose learning notions of similarity both within the same video and between different videos, by leveraging inter-video nearest-neighbor (NNs) from the global space **in addition to** existing intra-video clips as diverse positive pairs for contrastive learning. Our method "Inter-Intra Video Contrastive Learning" (IIVCL) defines a second positive key as the most similar video found from an evolving queue of randomly sampled videos in the learned representation space. In summary, our contributions are:

(i) Going beyond single-video positives by leveraging globally sampled nearest-neighbors to increase the semantic diversity of positive keys and introduce higher-level notions of similarity.

(ii) IIVCL, a simple yet effective self-supervised video contrastive learning algorithm that plugs into existing work and jointly learns intra and inter-video similarity using only RGB and no clustering.

(iii) Balancing local and global similarity to improve performance on video action recognition, action detection, and video retrieval — even in a few-shot learning setting.

## 2 RELATED WORK

**Self-supervised image representation learning.** The re-emergence of contrastive learning elevated self-supervised image representation learning as a viable alternative paradigm to fully-

supervised learning (He et al., 2020; Chen et al., 2020c;a;b; Grill et al., 2020; Chen & He, 2021). These methods encourage models to be invariant to multiple augmented views of the same image.

Some recent works such as Zhuang et al. (2019); Caron et al. (2020); Li et al. (2020) go beyond single-instance contrastive learning by using clusters to find semantically relevant positive pairs. Our work is similar to NNCLR (Dwibedi et al., 2021) which uses nearest-neighbors for image contrastive learning, but differs in that we combine inter-video nearest-neighbors with intra-video positives for **video** contrastive learning, which brings unique challenges due to the temporal dimension.

**Self-supervised video representation learning.** Several recent works have considered contrastive learning for video. These methods differ in their definition of positives and negatives. Some works use different subclips of equal-length from the same video as positive samples (Qian et al., 2021; Feichtenhofer et al., 2021; Dave et al., 2021), or different frames from the same video (Tian et al., 2020). Recasens et al. (2021) samples two subclips of different length from the same video to encourage generalization to broader context. Other works utilize optical flow in a cross-modal context; Han et al. (2020b) uses optical flow to mine RGB images with similar motion cues as positives, while other works do not mine positives but instead learn from the natural correspondence between optical flow and RGB within the same video (Xiao et al., 2021), or the same frame (Tian et al., 2020).

In contrast, our work goes beyond local definitions of positives from a single-video and expands to globally sampled nearest-neighbor videos, but without using optical flow nor separate training phases like Han et al. (2020b). Unlike Chen et al. (2021a), our work uses the online representation space to pick NNs on the fly instead of pre-computing video clusters. Unlike Chen et al. (2021a); Morgado et al. (2021) which uses audio-visual correspondence, our work only uses RGB frames.

## 3 Inter-Intra Video Contrastive Learning

**Intra-Video Contrastive Learning.** We use the contrastive loss $\mathcal{L}^{\text{NCE}}$ (Oord et al., 2018). Contrastive learning methods for video such as CVRL (Qian et al., 2021) and $\rho$-MoCo (Feichtenhofer et al., 2021) use the embeddings of two subclips $z_1$ and $z_2$ from the same video as positives. Given a queue $Q$ of randomly sampled embeddings, the intra-video contrastive loss is then:

$$\mathcal{L}_{\textit{Intra}}(z_1, z_2,\ Q) = \mathcal{L}^{\text{NCE}}(z_1, z_2,\ Q) \tag{1}$$

**Nearest-Neighbor Contrastive Learning.** We maintain a queue $Q$ that is updated with embeddings from each forward pass, which allows us to directly compute cosine similarities between the input video and queue. Given an embedded input video $x$ and queue $Q$ of randomly sampled embeddings across the dataset, we use the nearest-neighbor of $x$ as its positive:

$$\text{NN}(x, Q) = \underset{z \in Q}{\text{argmax}}(x \cdot z) \tag{2}$$

Let $z_1$ and $z_2$ be the embeddings of two subclips from the same video. We use $Q$ for both selecting the NN as a positive key and providing negatives (excluding the NN). Using the nearest-neighbor operation in Eq. 2 to select the positive key for $z_1$ as $\text{NN}(z_2, Q)$, and removing it from $Q$ to form $Q^- = Q \setminus \text{NN}(z_2, Q)$, we have the NN contrastive loss:

$$\mathcal{L}_{\textit{NN}}(z_1, z_2,\ Q) = \mathcal{L}^{\text{NCE}}(z_1, \text{NN}(z_2, Q),\ Q^-) \tag{3}$$

**Combined Intra and Inter Training Objective**
The final training objective is $\lambda_{\textit{Intra}} \cdot \mathcal{L}_{\textit{Intra}} + \lambda_{\textit{NN}} \cdot \mathcal{L}_{\textit{NN}}$. See A.3 and A.4 for more details.

## 4 Experiments

### 4.1 Baselines

$\rho$-MoCo (Feichtenhofer et al., 2021) is a leading contrastive learning work that samples intra-video clips. We primarily compare against $\rho$-MoCo for $\rho$=2 (two clips per video), pretrained for 200 epochs on unlabeled K400, and call this baseline *$\rho$-MoCo*. Feichtenhofer et al. (2021) does not test *$\rho$-MoCo* on all downstream datasets, so we rerun all experiments for fair comparison. We distill the effect of improved positive diversity and balanced global-local context on downstream performance.

| Method | Date | Backbone | Pretrain Data (duration) | Pretrain Epochs | Pretrain Input Size | UCF | HMDB | K400 |
|---|---|---|---|---|---|---|---|---|
| Supervised | | R3D-50 | scratch | | $8 \times 224^2$ | 68.8 | 22.7 | 74.7 |
| DPC (Han et al., 2019) | 2019 | R2D-3D34 | K400 (28d) | 110 | $40 \times 224^2$ | 75.7 | 35.7 | - |
| DynamoNet (Diba et al., 2019) | 2019 | STCNet | YT8M-1 (58d) | - | $32 \times 112^2$ | 88.1 | 59.5 | - |
| SpeedNet (Benaim et al., 2020) | 2020 | S3D-G | K400 (28d) | - | $16 \times 224^2$ | 81.1 | 48.8 | - |
| MemDPC (Han et al., 2020a) | 2020 | R2D-3D34 | K400 (28d) | - | $40 \times 224^2$ | 86.1 | 54.5 | - |
| VideoMoCo (Pan et al., 2021) | 2021 | R(2+1)D18 | K400 (28d) | 200 | $32 \times 224^2$ | 78.7 | 49.2 | - |
| TECVRL (Jenni & Jin, 2021) | 2021 | R3D-18 | K400 (28d) | 200 | $16 \times 128^2$ | 87.1 | 63.6 | - |
| **IIVCL** | | R3D-18 | K400 (28d) | 200 | $8 \times 128^2$ | 89.4 | 60.2 | 59.2 |
| VTHCL (Yang et al., 2020) | 2020 | R3D-50 | K400 (28d) | 200 | $8 \times 224^2$ | 82.1 | 49.2 | 37.8 |
| CVRL (Qian et al., 2021) | 2020 | R3D-50 | K400 (28d) | 1000 | $16 \times 224^2$ | 92.2 | 66.7 | 66.1 |
| $\rho$-MoCo$^\dagger$ (Feichtenhofer et al., 2021) | 2021 | R3D-50 | K400 (28d) | 200 | $8 \times 224^2$ | 91.1 | 65.3 | 65.4 |
| **IIVCL** | | R3D-50 | K400 (28d) | 200 | $8 \times 224^2$ | 92.6 | 65.8 | 65.7 |

Table 1: **Comparison with state-of-the-art self-supervised approaches**. Reported results are top-1 accuracy under finetune protocol (UCF, HMDB) and linear protocol (K400). $^\dagger$ refers to our reimplementation (Sec. 4.1).

| Method | Backbone | Pretrain Data | Top-1 Acc |
|---|---|---|---|
| Supervised (Feichtenhofer et al., 2019) | R3D-50 | K400 | 52.8 |
| *$\rho$-MoCo* (Feichtenhofer et al., 2021) | R3D-50 | K400 | 53.6 |
| **IIVCL** | R3D-50 | K400 | 53.8 |

Table 2: **Action recognition on Something-Something.** We finetune on SSv2 and report top-1 accuracy.

| Method | Pretrain Data | Top-1 Acc |
|---|---|---|
| Supervised (Feichtenhofer et al., 2019) | K400 | 21.9 |
| CVRL (Qian et al., 2021) | K400 | 16.3 |
| *$\rho$-MoCo* (Feichtenhofer et al., 2021) | K400 | 18.6 |
| **IIVCL** | K400 | 19.0 |

Table 3: **Action detection on AVA.** We finetune on AVA using a clip size of $8 \times 8$ and report mAP@0.5 IOU.

### 4.2 Action Recognition

Unless otherwise noted, we train IIVCL on unlabeled K400 (Kay et al., 2017) (240K videos) for 200 epochs, then transfer to downstream tasks. We use two popular evaluation protocols for self-supervised representations: **(i) Linear evaluation** freezes the backbone and trains a linear classifier, and **(ii) Finetuning** trains the entire network end-to-end. We report top-1 accuracy on UCF101 (Soomro et al., 2012), HMDB51 (Kuehne et al., 2011), and Something-Something v2 (Goyal et al., 2017) with finetuning, and on Kinetics-400 with linear eval.

In Table 1, we compare IIVCL against state-of-the-art self-supervised methods that use only RGB frames. Compared to the *$\rho$-MoCo* baseline, IIVCL outperforms by 1.5% on UCF, 0.5% on HMDB, and 0.3% on K400. Consistent improvements show the effectiveness of adding nearest-neighbor positives. To fairly compare against other works which use a smaller backbone, we also present results for IIVCL trained with R18 backbone and input resolution of 128x128. We outperform all methods in this setting including VideoMoco (Pan et al., 2021) which uses larger input resolution.

We further evaluate on Something-Something v2 (Goyal et al., 2017) (SSv2) which is a challenging benchmark focused on understanding fine-grained motions. IIVCL slightly outperforms the *$\rho$-MoCo* baseline, showing that our method can consistently generalize to different domains.

### 4.3 Action Detection on AVA

To test whether our method can also generalize to new downstream tasks, we evaluate IIVCL on action detection which not only requires classifying the action but also localizing the person performing the action, using the AVA dataset (Gu et al., 2018). More details in A.5.2. IIVCL also outperforms the *$\rho$-MoCo* baseline on action detection by 0.4%. IIVCL also outperforms CVRL (Qian et al., 2021) despite CVRL being trained for 5x more epochs and using 2x more pretraining frames.

### 4.4 Ablation: Task Generalization of Intra and NN Weights

In Table 6, we summarize the above results and also ablate our choice of $\lambda_{NN}$. Note that ($\lambda_{Intra}$=0.0, $\lambda_{NN}$=1.0) corresponds to a pure NN sampling strategy that uses no intra-video pairs, aka a video-

| | UCF Finetune | | | K400 Linear | |
|---|---|---|---|---|---|
| Method | 1% | 5% | 20% | 1% | 10% |
| *ρ-MoCo* Feichtenhofer et al. (2021) | 41.8 | 68.0 | 84.7 | 34.3 | 53.3 |
| **IIVCL** | 44.3 | 68.9 | 85.0 | 34.9 | 54.2 |

Table 4: **Few-shot learning on UCF101 and K400.** Rows indicate different pretrained models on K400. Columns vary the % of UCF training data used for finetuning and % of K400 training data used for linear eval.

| Epochs | $\rho$ | UCF | HMDB | K400 | SSv2 |
|---|---|---|---|---|---|
| 200 | 2 | 92.6 | 65.8 | 65.7 | 53.8 |
| 200 | 4 | 93.3 | 67.8 | 66.6 | 54.6 |
| 400 | 2 | 93.3 | 68.1 | 67.1 | 54.2 |

Table 5: **More pretraining epochs and NNs.** Data is unlabeled K400.

| Model | | Action Recognition | | | | Action Detection | |
|---|---|---|---|---|---|---|---|
| | | Finetune | | | Linear | Finetune | **Avg.** |
| $\lambda_{Intra}$ | $\lambda_{NN}$ | UCF | HMDB | SSv2 | K400 | AVA | **Rank** |
| 1.0 | 0.0 | 91.1 (#3) | 65.3 (#3) | 53.6 (#2) | 65.4 (#2) | 18.6 (#2) | 2.4 |
| 1.0 | 1.0 | **92.6** (#1) | 65.8 (#2) | **53.8** (#1) | **65.7** (#1) | **19.0** (#1) | 1.2 |
| 0.0 | 1.0 | 91.2 (#2) | **66.2** (#1) | 53.2 (#3) | 63.7 (#3) | 18.4 (#3) | 2.4 |

Table 6: **Do NNs lead to better generalization?** The first row corresponds to the *ρ-MoCo* baseline and second row corresponds to IIVCL. All models are pretrained on K400 for 200 epochs. $\lambda_{Intra}$=0.0 means no intra-video positives are used. We denote rank in blue parenthesis (where 1st = best) on each task.

| | | | UCF | | | | HMDB | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Method | Network | Pretrain | R@1 | R@5 | R@10 | R@20 | R@1 | R@5 | R@10 | R@20 |
| SpeedNet Benaim et al. (2020) | S3D-G | K400 | 13.0 | 28.1 | 37.5 | 49.5 | | | | |
| GDT Patrick et al. (2020) | R(2+1)D | K400 | 57.4 | 73.4 | 80.8 | 88.1 | 25.4 | 51.4 | 63.9 | 75.0 |
| VCLR Kuang et al. (2021) | R2D-50 | K400 | 70.6 | 80.1 | 86.3 | 90.7 | 35.2 | 58.4 | 68.8 | 79.8 |
| *ρ-MoCo* Feichtenhofer et al. (2021) | R3D-50 | K400 | 73.2 | 87.0 | 91.8 | 95.5 | 36.3 | 61.9 | 72.0 | 82.5 |
| **IIVCL** | R3D-50 | K400 | 74.2 | 87.6 | 92.1 | 95.1 | 37.6 | 62.2 | 72.9 | 82.5 |

Table 7: **Zero-shot video retrieval on UCF101 and HMDB.** Recall @ topK for UCF and HMDB.

analog of NNCLR Dwibedi et al. (2021). We summarize the average rank per task for each configuration. Pure NN sampling is surprisingly competitive with pure intra-video sampling on every task, despite learning zero local semantics during SSL pretraining. However, combining the intra and NN loss leads to the best performance with a small boost, supporting our intuition.

### 4.5 Effect of More Epochs and More NNs

Downstream accuracy increases with the number of temporal samples per video and pretraining duration (Table 5). Performance does not seem to saturate.

### 4.6 Few-Shot Learning and Zero-Shot Video Retrieval

We first compare against *ρ-MoCo* on UCF101 using finetuning when training data is limited to 1%, 5%, and 20%, and the evaluation set remains the same. We observe that IIVCL is more data efficient across all three subsets. We then compare against *ρ-MoCo* on K400 using linear evaluation when training data is limited to 1% and 10%, and the evaluation set remains the same. We observe similar improvements across both subsets for IIVCL. For both UCF and K400, the delta between IIVCL and *ρ-MoCo* is largest for the smallest training set of 1% data, indicating that nearest-neighbors are particularly helpful for generalizing to few-shot setting. See Table 4.

We also evaluate on video retrieval where the extracted features are directly used to find the nearest-neighbors. IIVCL outperforms *ρ-MoCo* for all but one recall threshold on UCF and all recall thresholds on HMDB. This indicates that even without any downstream training, IIVCL is better able to push similar videos of a different downstream dataset closer in the embedding space. See Table 7.

## 5 Conclusion

We presented IIVCL, which addresses limitations of existing contrastive learning works that sample only intra-video positives by leveraging NN samples from a global neighborhood. IIVCL is simple, improves performance on several video tasks, and can be directly plugged into existing work.

# A APPENDIX

## A.1 FIGURES

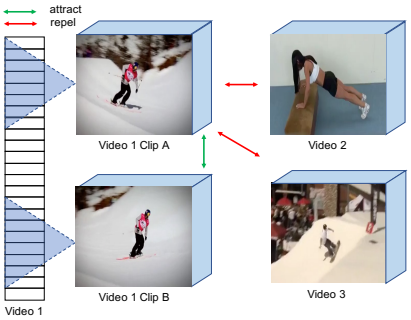
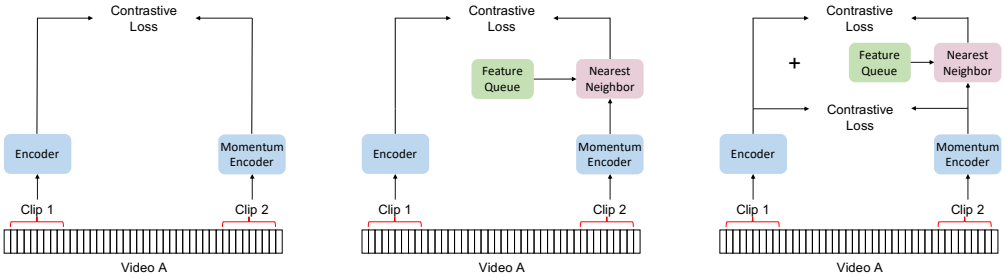



Figure 1: Popular contrastive learning methods sample positive clips within the same video boundary, e.g. clips A and B. However, other *similar* videos such as 3 are **never** used as positives, even if semantically similar.




(a) Intra-video sampling. (b) Inter-video nearest-neighbor. (c) IIVCL

Figure 2: IIVCL Overview. Fig. 2a shows intra-video positive sampling as used by state-of-the-art (Qian et al., 2021; Feichtenhofer et al., 2021). Fig. 2b shows our proposal to leverage nearest-neighboring samples from an evolving queue as positives. Fig. 2c is our IIVCL which combines a) and b) to learn similarity both within the same video and between different videos. NN sampling is simple and directly plugs into existing methods.

## A.2 ADDITIONAL METHODS: CONTRASTIVE LOSS

Contrastive learning maximizes the similarity of a given embedded sample $q$ with its embedded positive key $k^+$, while minimizing similarity to negative embeddings $n_i$. In the rest of this work, we refer to ($q$, $k^+$) as "positive pairs". We utilize the InfoNCE loss (Oord et al., 2018) for self-supervised video representation learning, which is given below:

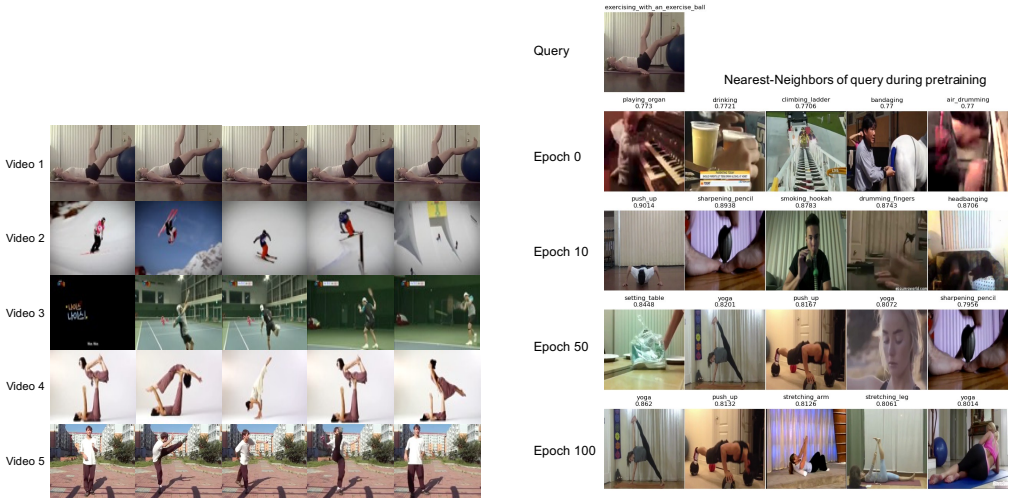
$$\mathcal{L}^{\text{NCE}}(q, k^+, \mathcal{N}^-) = -\log \frac{\exp(sim(q, k^+)/\tau)}{\sum\limits_{k \in \{k^+\} \cup \mathcal{N}^-} \exp(sim(q, k)/\tau)} \tag{4}$$

where $\tau > 0$ is a temperature hyper-parameter and $sim(\cdot)$ denotes the similarity function — which in this work is the dot product (cosine) similarity between two $\ell_2$ normalized vectors: $sim(q, k) = q \cdot k = q^T k/(||q||\ ||k||)$.

## A.3 ADDITIONAL METHODS: MULTI-TASK OBJECTIVE

**Combined Intra and Inter Training Objective**
We use the same backbone but separate MLP projection heads to process the intra-video and NN positive pairs. As each of these pretext tasks learns a different notion of similarity, we combine them via a multi-task loss. We also maintain two separate queues of embeddings: $Q_{\text{Intra}}$ and $Q_{\text{NN}}$. $Q_{\text{NN}}$ is used both to find the NN and provide negative keys (excluding the NN), while $Q_{\text{Intra}}$ only provides negative keys. We expand on these details in section A.4.

Specifically, let $f^q(\cdot)$ and $f^k(\cdot)$ be the encoder and its offline momentum-updated version, $g^{\text{Intra}}(\cdot)$ and $g^{\text{NN}}(\cdot)$ be two separate MLP heads, and $x_1$ and $x_2$ be two subclips sampled from the same video.




(a) Intra-video sampling results in **low diversity** of positive keys because subclips belong to the same short video. Each row shows a different video.

(b) For a query image (first row), the evolution of the top-5 nearest-neighbors during SSL pretraining starting from random initialization.

Figure 3: Intra-video vs. inter-video nearest-neighbor positives. In a), positive keys are always restricted to a single video boundary, which limits diversity. In b), positive keys are sampled globally using similarity in the learned feature space, which improves during pretraining. NNs do not necessarily belong to the same semantic class as the query. This global notion of similarity improves generalization.

We first obtain the embeddings ($z_1^{\text{Intra}}$, $z_2^{\text{Intra}}$) and ($z_1^{\text{NN}}$, $z_2^{\text{NN}}$).

$$z_1^{\text{Intra}} = g^{\text{Intra}}(f^q(x_1)); \quad z_2^{\text{Intra}} = g^{\text{Intra}}(f^k(x_2))$$
$$z_1^{\text{NN}} = g^{\text{NN}}(f^q(x_1)); \quad z_2^{\text{NN}} = g^{\text{NN}}(f^k(x_2))$$

After obtaining the embeddings, we combine Eqs. 1 and 3 to get the final training objective. Note that we use a symmetric loss but show only one side for simplicity. $\lambda_{Intra}$ and $\lambda_{NN}$ are tunable parameters that control the contribution of each loss, which in our work is 1.0 for both. An ablation for this is in Tab. 6.

$$\begin{aligned}\mathcal{L}(z_1^{\text{Intra}}, z_2^{\text{Intra}}, z_1^{\text{NN}}, z_2^{\text{NN}}) = \lambda_{Intra} \cdot \mathcal{L}_{Intra}(z_1^{\text{Intra}}, z_2^{\text{Intra}}, Q_{\text{Intra}}) \\ + \lambda_{NN} \cdot \mathcal{L}_{NN}(z_1^{\text{NN}}, z_2^{\text{NN}}, Q_{\text{NN}})\end{aligned} \tag{5}$$

Note that class labels are not used and the model is free to learn its own notion of similarity. Over the course of pretraining, the model becomes better at picking semantically similar nearest-neighbor video clips while introducing additional diversity of positive samples that is not possible through sampling intra-video clips, as shown in Fig. 3.

### A.4 Additional Methods: Pretraining Methodology

#### A.4.1 Momentum Encoder and Queue

We make several design choices that enable end-to-end learning from unlabeled videos using our method. As contrastive learning requires large batch sizes (Chen et al., 2020a) and computing video embeddings is expensive, we use a FIFO queue that is updated with embeddings from a momentum encoder, similar to He et al. (2020); Chen et al. (2020c). The momentum encoder's weights $\theta_k$ are updated as a moving average of the encoder's weights $\theta_q$, with momentum coeff. $m \in [0, 1)$, as given by Eq. 6. Thus the momentum encoder receives no gradients.

$$\theta_k \leftarrow m\theta_k + (1 - m)\theta_q \tag{6}$$

We share the same encoder but utilize separate MLP heads when processing intra-video and nearest-neighbor positives. We also maintain two separate queues for each task. Note that the queue contains approximate representations of a large subset of pretraining data in memory and is dynamically

updated "for free" since we only use embeddings that are already computed during the forward pass. Our method scales to large data and is more efficient than methods that utilize clustering such as Caron et al. (2020); Li et al. (2020); Chen et al. (2021a); Patrick et al. (2020). Our method also allows for more up-to-date representations than methods that use an offline positive set (Morgado et al., 2021; Chen et al., 2021b).

#### A.4.2 Implementation Details

**Loss.** The temperature $\tau = 0.1$. $\lambda_{Intra} = 1.0$, $\lambda_{NN} = 1.0$.

**Encoder.** For all experiments, we use a ResNet3D-50 8x8 slow pathway from Feichtenhofer et al. (2019) with He initialization (He et al., 2015), unless otherwise indicated. Outputs are taken after the global average pooling layer to form a 2048-d embedding. Following Chen et al. (2020c); Feichtenhofer et al. (2021), we use a 3-layer projection MLP during pretraining only. The MLP has hidden dimension 2048 and final embedding dimension of 128 with no batch norm (BN). The MLP is then removed for downstream experiments. As mentioned above, we use two separate MLP heads for producing intra-video and NN embeddings.

**Pretraining Hyperparameters.** We train for 200 epochs using SGD optimizer (momentum 0.9, weight decay $10^{-4}$) with a total batch size of 512. BN statistics are computed per GPU. We linearly warmup the learning rate to $0.4$ over the first 35 epochs, then use half-period cosine decay (Loshchilov & Hutter, 2016).

We use a queue storing 65536 negatives and shuffling-BN to avoid information leakage and overfitting (He et al., 2020). The momentum encoder weights are updated per Eq. 6 with an annealed momentum coeff. as in Feichtenhofer et al. (2021), initialized to 0.994.

**Data and Augmentations** We sample two 8-frame clips with a temporal stride of 8 from each video for self-supervised pretraining. We apply random shortest-side resizing to [256, 320] pixels, color jittering (ratio 0.4, p=0.8), grayscale conversion (p=0.2), Gaussian blur (p=0.5), horizontal flip (p=0.2), and random cropping to $224 \times 224$.

### A.5 Evaluation Details

#### A.5.1 Action Recognition

For UCF101 Soomro et al. (2012) and HMDB51 Kuehne et al. (2011), we report finetuning top-1 accuracy on split 1. UCF101 contains 9.5K/3.7K train/test videos with 101 action classes, and HMDB51 contains 3.5K/1.5K videos (mostly from movies) with 51 action classes. For K400 Kay et al. (2017), we report linear evaluation top-1 accuracy. Kinetics contains 240K/19K train/test videos with 400 action classes. We sample $8 \times 8$ clips for all datasets. At test-time, we use standard 10 (temporal) $\times$ 3 (spatial) crop evaluation Feichtenhofer et al. (2019). We report the avg. of three runs.

#### A.5.2 Action Detection

We evaluate on the AVA dataset (Gu et al., 2018) which contains 221K/57K training and validation videos, and report mean Average Precision (mAP) at IOU threshold 0.5. We follow Feichtenhofer et al. (2021) and use our self-supervised trained R3D-50 as the backbone for a Faster R-CNN detector. We then extend the 2D RoI features into 3D along the temporal axis, and apply RoIAlign and temporal global average pooling. The RoI features are then max-pooled and fed to a per-class sigmoid classifier. We also use a similar training schedule as Feichtenhofer et al. (2021), except we train for only 20 epochs with batch size 64, and use an initial learning rate of 0.1 with 10x step-wise learning rate decay at epochs 5, 10, and 15.

### A.6 More Ablations

#### A.6.1 MLP Heads During Pretraining

We trained a version of IIVCL that shares the MLP head for both intra-video and NN pairs. In this case, the pretraining loss fails to converge and downstream task results are poor. We hypothesize

this is due to the different feature spaces learned for intra-video clips vs. inter-video NNs. Thus, we share the backbone but use separate MLP heads during pretraining.

### A.7 Additional Visualizations

We provide additional qualitative examples to help visualize what the model is learning.

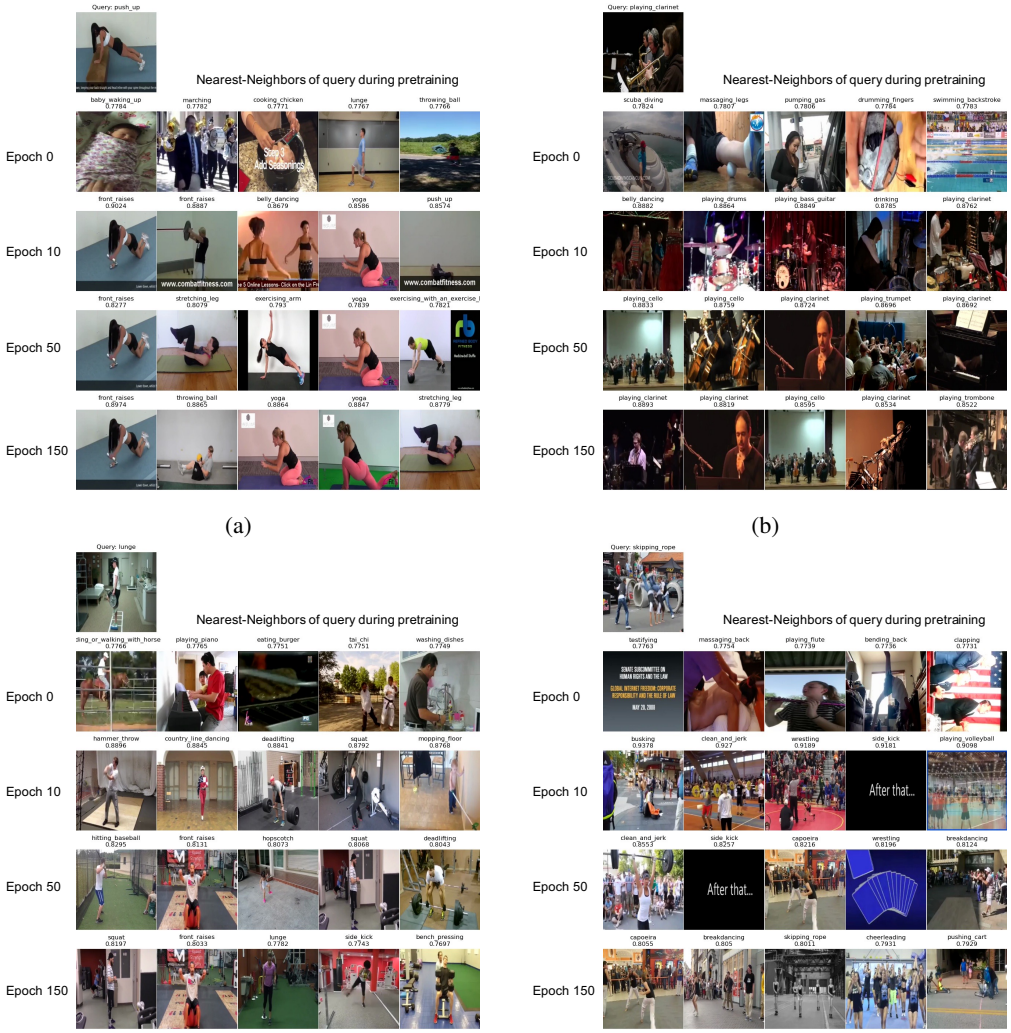



Figure 4: Evolution of inter-video nearest-neighbor positives during pretraining, starting from random initialization. Videos are sampled across the dataset using similarity measured by the learned feature space, which improves during pretraining. Top row is the query video while other rows indicate different epochs of pretraining.

### A.8 Nearest-Neighbor Evolution During Pretraining

In Figure 4, we show how the nearest-neighbors vary over the course of pre-training, starting from random initialization. We observe that as training progresses, the nearest-neighbors become more semantically similar, which complements Figure 3b) of the main paper.

### A.9 DISCUSSION

#### A.9.1 CO-OCCURRENCE OF SEMANTIC CLASSES DURING PRETRAINING

Although labels are not used during self-supervised pretraining, we analyze the probability that a video in the negative queue belongs to the same class as the query video, to understand why the nearest-neighbor objective is beneficial. Assume the queue samples are uniformly sampled and that each class is balanced. Let the pretraining dataset have $K$ balanced classes, and the queue have $Q$ uniformly sampled samples where $Q$ is smaller than the size of the dataset. Then the probability of the above event is 1 - $[(K-1)/K]^Q$, which is well over 0.9 for $K$=400 (number of classes in Kinetics-400), $Q$=1024. Note that this calculation also applies to approaches that sample negatives from the mini-batch; let $Q$ be the mini-batch size. CVRL Qian et al. (2021) uses a mini-batch size of 1024 during pretraining. Thus, it is extremely likely that videos belonging to same class as the query are pushed away in the embedding space as negatives in works like He et al. (2020); Chen et al. (2020a); Qian et al. (2021); Feichtenhofer et al. (2021). Our work does not address this issue by removing poor choices of negatives from the negative set, but rather leverages those similar videos as additional positive keys for a second loss term via the NN sampling strategy, thus providing additional sources of similarity to learn from that would otherwise be ignored.

Additionally, by dynamically computing the positive key using the learned representation space and sampling videos globally, we allow the model to continually evolve its notion of semantic similarity; the quality of the chosen NNs improves as the model learns as demonstrated by Fig. 3. With intra-video positive pair sampling, the learned representation is not used to choose the positive pairs — two clips are simply randomly sampled from within a single video.

#### A.9.2 LIMITATIONS AND FUTURE WORK

While the focus of our work was on improving the diversity of positive keys and balancing global with local notions of similarity, our method can be improved by reducing false negatives similar to works such as Chuang et al. (2020). We could also try leveraging audio-video correspondence. Lastly, we are interested in further analyzing why certain pretext tasks succeed or fail for certain downstream tasks, which may inform the design of future self-supervised learning frameworks.

## REFERENCES

Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9922–9931, 2020.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Thirty-fourth Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

Brian Chen, Andrew Rouditchenko, Kevin Duarte, Hilde Kuehne, Samuel Thomas, Angie Boggust, Rameswar Panda, Brian Kingsbury, Rogerio Feris, David Harwath, et al. Multimodal clustering networks for self-supervised learning from unlabeled videos. *arXiv preprint arXiv:2104.12671*, 2021a.

Shixing Chen, Xiaohan Nie, David Fan, Dongqing Zhang, Vimal Bhat, and Raffay Hamid. Shot contrastive self-supervised learning for scene boundary detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9796–9805, 2021b.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020b.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning, 2020c.

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. *Advances in neural information processing systems*, 33:8765–8775, 2020.

Ishan Dave, Rohit Gupta, Mamshad Nayeem Rizve, and Mubarak Shah. Tclr: Temporal contrastive learning for video representation. *arXiv preprint arXiv:2101.07974*, 2021.

Ali Diba, Vivek Sharma, Luc Van Gool, and Rainer Stiefelhagen. Dynamonet: Dynamic action and motion network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6192–6201, 2019.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9588–9597, October 2021.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6202–6211, 2019.

Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3299–3309, 2021.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pp. 5842–5850, 2017.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Pires, Zhaohan Guo, Mohammad Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *Neural Information Processing Systems*, 2020.

Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6047–6056, 2018.

Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 0–0, 2019.

Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pp. 312–329. Springer, 2020a.

Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. *Advances in Neural Information Processing Systems*, 33:5679–5690, 2020b.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Simon Jenni and Hailin Jin. Time-equivariant contrastive video representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9970–9980, 2021.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Haofei Kuang, Yi Zhu, Zhi Zhang, Xinyu Li, Joseph Tighe, Soren Schwertfeger, Cyrill Stachniss, and Mu Li. Video contrastive learning with global context. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3195–3204, 2021.

Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International conference on computer vision*, pp. 2556–2563. IEEE, 2011.

Junnan Li, Pan Zhou, Caiming Xiong, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12475–12486, 2021.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Tian Pan, Yibing Song, Tianyu Yang, Wenhao Jiang, and Wei Liu. Videomoco: Contrastive video representation learning with temporally adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11205–11214, 2021.

Mandela Patrick, Yuki M Asano, Polina Kuznetsova, Ruth Fong, Joao F Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal self-supervision from generalized data transformations. *arXiv preprint arXiv:2003.04298*, 2020.

Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6964–6974, 2021.

Adrià Recasens, Pauline Luc, Jean-Baptiste Alayrac, Luyu Wang, Ross Hemsley, Florian Strub, Corentin Tallec, Mateusz Malinowski, Viorica Patraucean, Florent Altché, Michal Valko, Jean-Bastien Grill, Aäron van den Oord, and Andrew Zisserman. Broaden your views for self-supervised video learning, 2021.

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

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pp. 776–794. Springer, 2020.

Fanyi Xiao, Joseph Tighe, and Davide Modolo. Modist: Motion distillation for self-supervised video representation learning, 2021.

Ceyuan Yang, Yinghao Xu, Bo Dai, and Bolei Zhou. Video representation learning with visual tempo consistency. *arXiv preprint arXiv:2006.15489*, 2020.

Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6002–6012, 2019.