No More Shortcuts: Realizing the Potential of Temporal Self-Supervision

Ishan Rajendrakumar Dave[1]*, **Simon Jenni**[2], **Mubarak Shah**[1]

[1]Center for Research in Computer Vision, University of Central Florida, USA
[2]Adobe Research, USA
ishandave@ucf.edu, jenni@adobe.com, shah@crcv.ucf.edu

Abstract

Self-supervised approaches for video have shown impressive results in video understanding tasks. However, unlike early works that leverage temporal self-supervision, current state-of-the-art methods primarily rely on tasks from the image domain (e.g., contrastive learning) that do not explicitly promote the learning of temporal features. We identify two factors that limit existing temporal self-supervision: 1) tasks are too simple, resulting in saturated training performance, and 2) we uncover shortcuts based on local appearance statistics that hinder the learning of high-level features. To address these issues, we propose 1) a more challenging reformulation of temporal self-supervision as frame-level (rather than clip-level) recognition tasks and 2) an effective augmentation strategy to mitigate shortcuts. Our model extends a representation of single video frames, pre-trained through contrastive learning, with a transformer that we train through temporal self-supervision. We demonstrate experimentally that our more challenging frame-level task formulations and the removal of shortcuts drastically improve the quality of features learned through temporal self-supervision. Our extensive experiments show state-of-the-art performance across 10 video understanding datasets, illustrating the generalization ability and robustness of our learned video representations. Project Page: https://daveishan.github.io/nms-webpage.

Introduction

Self-supervised learning (SSL) has unlocked the potential of large amounts of unlabelled data for large-scale pre-training of image (Zhou et al. 2022a; Li et al. 2022; He et al. 2022; Chen, Xie, and He 2021; Chen et al. 2020) and, more recently, video representations (Tong et al. 2022; Pan et al. 2021; Feichtenhofer et al. 2021; Dave et al. 2022; Feichtenhofer et al. 2022). The potential benefits of SSL methods on video are even greater than on images due to their larger dimensionality and much greater cost of comprehensive human labeling.

Video understanding tasks, such as action recognition, depend on representations that capture both the static scene appearance (*e.g.*, object textures, pose, and layout) and scene dynamics (*e.g.*, the change in pose and relative dynamics of objects). Human activity recognition, in particular, crucially depends on accurate representations of the human pose and how it deforms over time as a person performs an action.

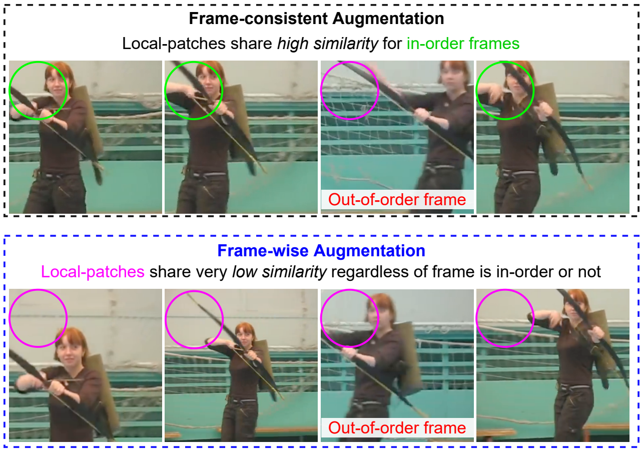
Frame-consistent Augmentation
Local-patches share *high similarity* for in-order frames
Out-of-order frame
Frame-wise Augmentation
Local-patches share very *low similarity* regardless of frame is in-order or not
Out-of-order frame

Figure 1: Breaking Shortcuts in Temporal SSL Tasks. We illustrate an example of our pretext task where the model has to localize out-of-order frames. In the first row, the input sequence has identical augmentations applied to each frame, whereas the second row shows our proposed independent frame-wise augmentations. We observe that the task can be solved by observing only a local image patch (highlighted circle) when identical augmentations are used. As a result, the task does not require learning higher-level features, *e.g.*, object dynamics. In contrast, when applying the proposed frame-wise augmentations (second row), a local observation (highlighted circle) is insufficient, and the model must consider the global scene context.

However, to what extent current video SSL approaches capture both static and temporal features in videos is unclear. Indeed, we observe that many of the current methods rely on contrastive objectives (Qian et al. 2021b; Feichtenhofer et al. 2021; Pan et al. 2021; Yang et al. 2020), which encourage spatial and temporal invariance and thus do not promote the learning of temporal features. While several works demonstrated the benefits of including additional objectives that encourage the learning of temporal features, *e.g.*, through pretext tasks (Jenni, Meishvili, and Favaro 2020; Xu et al. 2019; Bai et al. 2020; Misra, Zitnick, and Hebert 2016), it

*Majority of work done as an intern at Adobe Research, USA.

is unclear whether these methods truly capture changes in object dynamics or are rather hindered by relying on lower-level local motion features.

Such "shortcuts" are common in self-supervised task formulations (Doersch, Gupta, and Efros 2015; Jenni and Favaro 2018) and prevent the learning of higher-level features. Indeed, we observe as one of our key insights, that existing temporal learning tasks, *i.e.*, those recognizing temporal transformations (Misra, Zitnick, and Hebert 2016; Benaim et al. 2020; Jenni, Meishvili, and Favaro 2020), can be solved by relying on only local appearance statistics (see Figure 1 for an illustration and Table 1 for additional empirical evidence). These low-level solutions hinder the learning of temporal features that capture the global object dynamics. Furthermore, existing temporal pretext tasks are often limited by near-perfect training performance, suggesting that the task difficulty is not sufficient.

To address these issues, we make two technical contributions. First, to address the newly identified shortcuts in temporal self-supervision, we propose an effective frame-independent augmentation strategy. In contrast to best practice in (self-supervised) video representation learning with 3D-CNNs (Qian et al. 2021b), where temporally consistent augmentations are the norm, we demonstrate that an independent per-frame jittering is hugely beneficial for temporal pretext tasks. As can be seen in Figure 1, such frame-wise augmentations force the model to consider global features (*e.g.*, pose) rather than local low-level shortcuts. This increases the difficulty of the task and promotes better feature learning.

As our second contribution and to further increase the difficulty of the pretext tasks, we propose to reformulate temporal self-supervision as a frame-level time-varying recognition task instead of the typical formulation as a clip-level classification task (Misra, Zitnick, and Hebert 2016; Benaim et al. 2020; Jenni, Meishvili, and Favaro 2020; Wang, Jiao, and Liu 2020; Jenni and Jin 2021). Concretely, inspired by prior pretext tasks about the temporal ordering and playback speed of whole video clips (Misra, Zitnick, and Hebert 2016; Benaim et al. 2020), we pose the following time-varying tasks: 1) Out-of-order Frame Localization (OFL) and 2) Time-varying Skiprate Prediction (TSP). In OFL, the model has to identify a subset of the frames that are out-of-order, *i.e.*, do not match the natural temporal order of most of the frames. For TSP, the network needs to predict the playback rate at each frame, which we artificially vary over time. Both these tasks require an accurate classification of *each frame* in the sequence, making them more challenging than prior *clip-level* formulations. We realize these learning tasks with a video transformer architecture (Neimark et al. 2021), wherein a frame encoder (pre-trained through image contrastive learning) is extended with a temporal transformer (trained through our temporal SSL tasks). This model thus effectively fuses the benefits of contrastive and temporal self-supervision.

Finally, we perform a very comprehensive evaluation of the learned video representations on a large number of downstream video understanding tasks to assess their generalization ability and robustness. While most prior video SSL studies focussed primarily on action recognition benchmarks (*e.g.*, UCF101, HMDB51, Kinetics400, Something-SomethingV2, NTU60, Charades), we additionally evaluate our representations and compare to the prior state-of-the-art on a variety of other aspects, including holistic video understanding (HVU), temporal correspondence tasks (DAVIS and JHMDB), and gait recognition (CASIA-B). We also evaluate the robustness of the video retrieval task to input video perturbations, following (Schiappa et al. 2023).

Contributions. Our contributions can be summarized as follows: 1) We identify shortcuts in temporal SSL based on local patch similarity. To mitigate these shortcuts, we propose a frame-independent augmentation strategy, 2) We propose frame-wise and time-varying reformulations of temporal pretext tasks instead of the typical clip-level formulations to increase the difficulty of the learning tasks, 3) We validate our contributions in extensive ablation experiments and comprehensively evaluate the learned video representations' generalization ability, including action-related tasks, holistic video understanding, temporal correspondence, and robustness to input perturbations. Our results demonstrate state-of-the-art performance across numerous benchmarks.

Prior Work

Prior work in self-supervised video representation learning is based on a variety of learning objectives. These include contrastive learning (Qian et al. 2021b; Yang et al. 2020; Dave et al. 2022, 2023), pretext tasks (Misra, Zitnick, and Hebert 2016; Benaim et al. 2020; Wei et al. 2018), masked video modeling (Tong et al. 2022; Feichtenhofer et al. 2022), and hybrid approaches combining multiple objectives (Bai et al. 2020; Jenni and Jin 2021; Yao et al. 2020). We compare our results with all these approaches in the experiments but focus the following discussion on related works also relying on temporal self-supervision, which is most relevant to our approach.

Temporal Pretext Tasks. These methods are based on recognizing different distortions of the natural temporal evolution of videos. Several works have explored *temporal ordering tasks*, which mainly deal with verifying or predicting the temporal order of video frames or short clips. For example, (Misra, Zitnick, and Hebert 2016; Yao et al. 2020; Bai et al. 2020) posed frame order verification as a prediction task of whether a sequence of 3-frames is in order or not. Other works posed temporal ordering as a sorting task, *e.g.*, of 3-frame sequences (Lee et al. 2017) or a set of short video clips (Xu et al. 2019; Hu et al. 2021). Other temporal learning signals in videos can be found in the playback direction (Wei et al. 2018) (forward vs. backward playback) and the playback speed of the video. For example, several works proposed the classification of different artificial playback speeds applied to a video for learning (Benaim et al. 2020; Wang, Jiao, and Liu 2020), or additionally recognizing non-uniformly warped videos (Jenni, Meishvili, and Favaro 2020).

How do our task formulations for OFL and TSP differ from prior work? Prior temporal pretext tasks are primarily formulated at the sequence level, *e.g.*, identifying whether

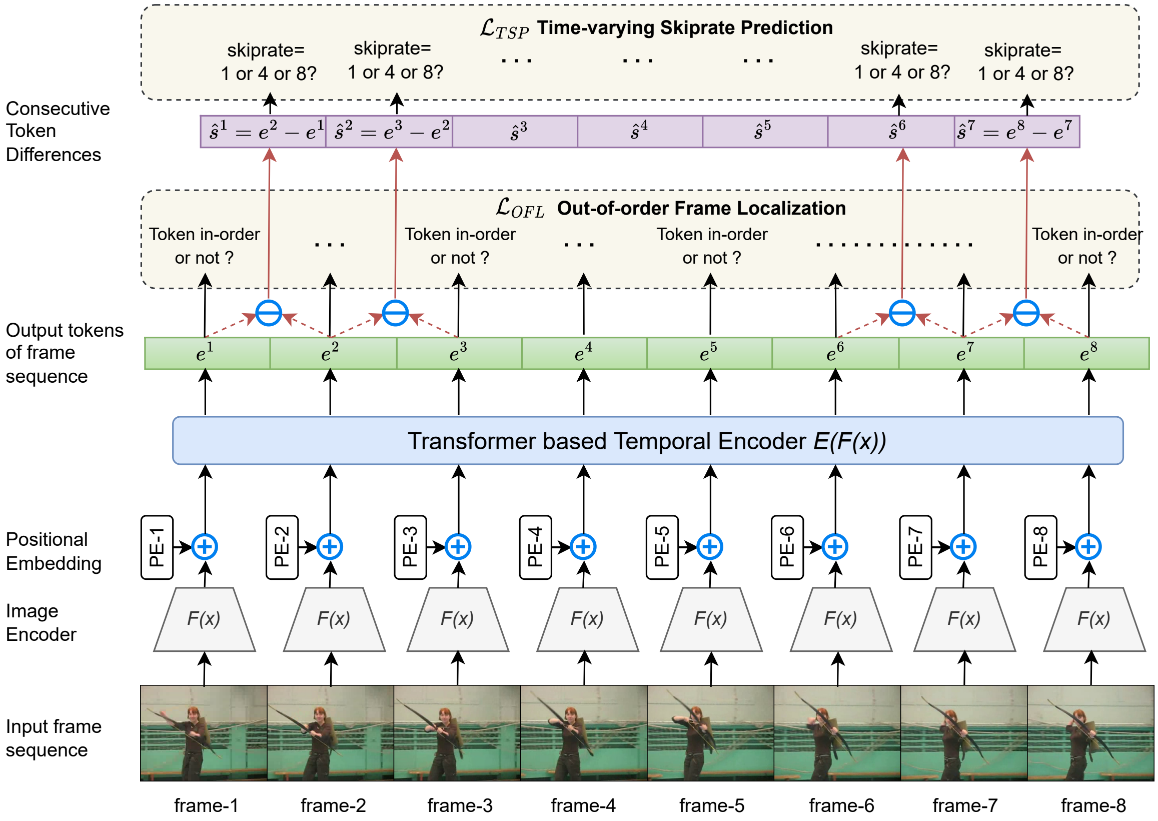

Figure 2: Model Overview. We utilize a Video Transformer Network (Neimark et al. 2021) architecture, which extends an image encoder F with a transformer E that takes frame embeddings as input tokens. To train such a model on unlabelled videos, we optimize a self-supervised objective consisting of two novel temporal pretext tasks: Out-of-order Frame Localization (OFL), and Time-Varying Skiprate Prediction (TSP). Details of each objective can be found in the Method section.

the whole sequence of frames is in correct order (Misra, Zitnick, and Hebert 2016) or exhibits a normal playback speed (Benaim et al. 2020). In contrast, our formulation is posed as a frame-wise prediction task, where each frame in the sequence is assigned a different label indicating if the frame is in order (OFL) or at what skip rate is applied to it (TSP). As our experiments show (Table 3), our more challenging per-frame formulations result in significantly better-performing features.

Shortcuts in Self-Supervised Learning. Many self-supervised learning tasks suffer from trivial solutions exploiting low-level cues in the data to solve the task (shortcuts). This hinders the learning of high-level generalizable features. For example, chromatic aberration cues were identified as shortcuts in image-based pretext tasks (Doersch, Gupta, and Efros 2015) and codec artifacts in video (Wei et al. 2018). Likewise, contrastive learning approaches rely on strong data augmentation to prevent trivial solutions (Chen et al. 2020). *To our best knowledge, we are the first to identify and prevent shortcuts in temporal-pretext tasks due to local patch appearances* (see Figure 1). As we demonstrate in Table 1 and 2, our frame-wise augmentation strategy is effective at preventing this shortcut and considerably improves downstream feature performance.

Method

Let $\{\mathbf{x_1}, \ldots, \mathbf{x_n}\}$ be set of unlabeled videos where each video consists of a sequence of frames, *i.e.*, $\mathbf{x_i} = [x_i^1, \ldots, x_i^{n_i}]$ and n_i defines the number of frames in video $\mathbf{x_i}$. Let further $\mathbf{x_i}[\mathcal{I}] = [x_i^j]_{j \in \mathcal{I}}$ denote a sequence of video frames based on frame indices $\mathcal{I} \subset \{1, \ldots, n_i\}$. In our model, we first extract frame feature vectors $f_i^j = F(x_i^j) \in \mathbb{R}^d$ with an image encoder f for each frame of the sequence, *i.e.*, $F\left(\mathbf{x_i}[\mathcal{I}]\right) = [f_i^j]_{j \in \mathcal{I}}$. These feature vectors are then considered as frame-tokens and fed as input to a transformer network E along with learnable position encodings. Concretely, the full model is given by $E\left(F\left(\mathbf{x_i}[\mathcal{I}]\right) + \mathrm{PE}_{|\mathcal{I}|}\right) = [e_i^1, \ldots, e_i^{|\mathcal{I}|}]$, where $\mathrm{PE}_{|\mathcal{I}|}$ is a sequence of $|\mathcal{I}|$ learnable position encodings, and e_i^j denote the set of output tokens after the transformer. We will now describe the various self-supervised learning objectives in our framework.

Out-of-order Frame Localization. As a first temporal pretext task, we propose to train the frame token transformer E to localize out-of-order frames. In this task, a subset of the frames fed to the transformer are incorrectly placed, *i.e.*, they have a random time shift applied to them. We then train the network E to predict whether each frame in the sequence

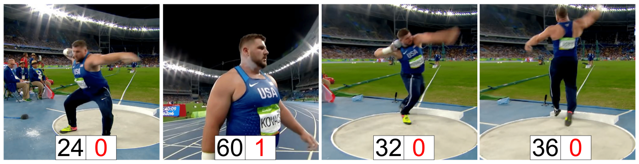

Figure 3: Out-of-order Frame Localization (OFL). The goal is to detect which frames are out of temporal order. The task is posed as a binary classification problem, where 0 indicates in-sequence frames and 1 indicates out-of-order frames. Frame-IDs are shown in black, and the self-supervised targets in red color.

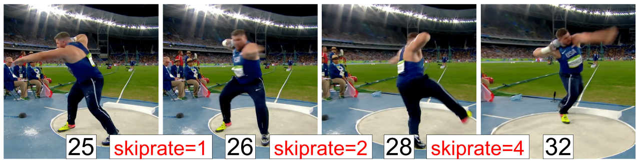

Figure 4: Time-Varying Skiprate Prediction (TSP). The goal is to predict the skiprate between the consecutive frames. We pose the task as an M-way classification problem. In this example, M=3 skiprate classes are used. The skiprate label is shown in red color between consecutive frames.

is correctly placed. Since most frames will have a correct placement, the transformer can leverage this global context to infer the correct movement patterns in the sequence and learn to detect frames that do not conform to them. We call this pretext task Out-of-order Frame Localization (OFL).

To build example sequences for training, we manipulate the frame sampling indices $\mathcal{I}$. A correct sequence during training is considered to be one where 1) all frames are in correct temporal order and 2) there is a constant offset between consecutive frames. Concretely, for a video $\mathbf{x_i}$ we sample correct frame indices as $\mathcal{I} = [t, t+\Delta, \dots, t+(p-1)\Delta]$, where p denotes the number of frames, Δ is the fixed frame offset, and t is the starting frame, which is randomly sampled from $t \sim \mathcal{U}(\{1, \dots, n_i - (p-1)\Delta\})$. To build inputs and targets for OFL training, we then sample and replace a random subset of the frame indices with other indices. Concretely, let $\mathcal{M} \in \{0,1\}^p$ be a randomly sampled binary sequence indicating for each index in $\mathcal{I}$ whether it should be changed ($\mathcal{M}_i = 1$) or kept ($\mathcal{M}_i = 0$). The ratio of changed indices $\rho = \frac{1}{p}\sum \mathcal{M}_i$ is randomly sampled from the interval $[0, 0.5]$ during training. Finally, the transformed indices are constructed as $\hat{\mathcal{I}} = (1-\mathcal{M}) \otimes \mathcal{I} + \mathcal{M} \otimes \mathcal{S}$, where $\otimes$ denotes element-wise multiplication and $\mathcal{S} \subset \{1, \dots, n_i\}$ is a random sequence of other frame indices. In our best setting, we restrict the sampling of $\mathcal{S}$ to $\{\min(\mathcal{I}) - \Delta, \dots, \min(\mathcal{I}-1)\} \cup \{\max(\mathcal{I})+1, \dots, \max(\mathcal{I})+\Delta\}$, *i.e.*, sampling frames before or after $\mathcal{I}$ with a maximum distance of Δ.

Finally, the OFL objective $\mathcal{L}_{OFL}$ is given by a standard binary cross-entropy loss, *i.e.*,

$$\mathcal{L}_{OFL} = -\sum_{e_i^j \in E\left(F\left(\mathbf{x_i}[\hat{\mathcal{I}}]\right)\right)} \mathcal{M}_j \log \sigma(e_i^j) + (1-\mathcal{M}_j) \log\left(1 - \sigma(e_i^j)\right), \tag{1}$$

where σ indicates a linear layer followed by a sigmoid activation function. An example input sequence and out-of-order frame are shown in Figure 3. We extensively study the design choices of this task in the Ablation section.

Time-varying Skiprate Prediction. As a second temporal pretext task, we propose the recognition of time-varying playback speeds (skiprate). For this task, we modify the frame indices for sampling model inputs to $\mathcal{I}_s = [t, t+s^1, \dots, t+\sum_{i=1}^{p} s^i]$, where $s^i \sim \mathcal{U}(\{1,4,8\})$ are independently sampled skiprates between frames i and $i+1$. The task is then to predict the sequence of time-varying skiprates s^i from the input sequence $\mathbf{x_i}[\mathcal{I}_s]$. Concretely, we model the probability of observing the three different skiprate classes as $\hat{s}^i = \psi(e^{i+1} - e^i) \in \mathbb{R}^3$, where ψ is a linear layer followed by a softmax activation acting on temporal differences of token embeddings. The TSP loss $\mathcal{L}_{TSP}$ is then given by a standard 3-way classification loss between the predicted and ground-truth skiprate classes $\hat{s}^i$ and s^i. Example inputs and target outputs are provided in Figure 4. TSP design choices are studied in the Ablation section.

Contrastive Loss We utilize two types of frame contrastive losses in our model training. The first one is a cross-clip term, wherein positive pairs for learning are built from frames of the *same clip* (or an augmented version of it), and negative pairs are built with frames belonging to *different videos*. Concretely, given a training batch $\mathcal{B}$ of the frame features f_i^j, the cross-clip loss is given by

$$\mathcal{L}_{C1} = \sum_{F(\mathbf{x_i}[\mathcal{I}]) \in \mathcal{B}} \sum_{j,k \in \mathcal{I}} \log\left(\frac{d\left(f_i^j, \hat{f}_i^k\right)}{\sum_{f_l^k \in \mathcal{B}} \mathbb{1}\{i \neq l\} d\left(f_i^j, \hat{f}_l^k\right)}\right), \tag{2}$$

where f_i^j and $\hat{f}_i^j$ denote features of two differently augmented views of x_i^j and

$$d(u_1, u_2) := \exp\left(\frac{1}{\lambda}\frac{\phi(u_1)^\intercal \phi(u_2)}{\|\phi(u_1)\|_2 \|\phi(u_2)\|_2}\right), \tag{3}$$

is a measure of similarity between the feature vectors u_1 and u_2, and ϕ is a projection implemented via a multilayer perceptron (MLP).

As a second frame-wise loss, we consider within-clip terms, where positive pairs are built solely through different image augmentations of the *same frame*, and negative pairs are constructed with two different frames from the *same clip*. Formally, this loss term is given by

$$\mathcal{L}_{C2} = \sum_{F(\mathbf{x_i}[\mathcal{I}]) \in \mathcal{B}} \log\left(\frac{d\left(f_i^j, \hat{f}_i^j\right)}{\sum_{f_i^k \in F(\mathbf{x_i}[\mathcal{I}])} d\left(f_i^j, \hat{f}_i^k\right)}\right). \tag{4}$$

Combined Training Objective. Finally, we combine our frame-level temporal pretext tasks with the contrastive loss $\mathcal{L}_C$ ($\mathcal{L}_{C1} + \mathcal{L}_{C2}$). To summarize, we optimize

$$\mathcal{L}_{SSL} = \lambda_O \mathcal{L}_{OFL} + \lambda_T \mathcal{L}_{TSP} + \lambda_C \mathcal{L}_C, \tag{5}$$

Scale	Framewise Augment?	Pretext Task Accuracies	
		OFL (mAP)	TSP (Top-1)
Full (224 x 224)	No	87.6	57.9
Patch (32 x 32)		78.2 (-10.7%)	52.3 (-9.6%)
Full (224 x 224)	Yes	84.1	56.8
Patch (32 x 32)		26.1 (**-68.9%**)	33.1 (**-41.5%**)

Table 1: Evidence for Shortcuts in Temporal Pretext Tasks. We show the pretext task performance of models trained on the full scene (224×224) or on only local patches (32×32), with and without our frame-wise augmentation strategy. We observe that a local patch model can achieve high accuracy when consistent augmentations are used, indicating that global features are not necessary. In contrast, only the model with the full scene context is able to achieve non-trivial performance when frame-wise augmentations are used. The relative drop in accuracy is shown in red.

where λ_O, λ_T, and λ_C are loss weights. The model architecture and training objectives are illustrated in Figure 2.

Avoiding Shortcuts in Temporal Pretext Tasks. The recognition of wrongly placed frames in our OFL task should encourage the learning of motion features, *e.g.*, temporal patterns of object deformations. Such patterns are crucial for video understanding tasks, *e.g.*, for human action recognition. However, we observe that the OFL task can often be solved by comparing the appearance of spatially local image patches in neighboring frames (see Figure 1 and experimental evidence in Table 1). Such low-level task solutions that only consider local spatio-temporal pixel statistics could hamper the learning of higher-level video features that capture the more important changes in object dynamics. As a solution to this problem, we propose to process the training frame sequences $\mathbf{x_i}[\hat{\mathcal{I}}]$ with spatial jittering applied *independently* to each frame in the sequence. Concretely, when augmenting a frame sequence, we first apply a standard augmentation strategy for contrastive learning τ_c (*e.g.*, color jittering, horizontal flipping, etc.), which is applied consistently to all frames and contains only weak spatial cropping. We then independently apply additional spatial augmentations τ_i (*e.g.*, random resizing and cropping) to each frame. The proposed augmentation for an example $\mathbf{x_i}$ can be expressed as

$$\hat{\mathbf{x}}_{\mathbf{i}} = [\tau_1 \circ \tau_c(x_i^1), \ldots, \tau_{n_i} \circ \tau_c(x_i^{n_i})]. \tag{6}$$

Such frame-wise augmentations introduce large differences in the local frame patch appearances and force the model to consider more global spatio-temporal features (*e.g.*, object dynamics) to solve the task.

Experiments

In this section, we first provide details about datasets, implementation, and experimental setup. We perform ablation studies and evaluate our SSL representation on various downstream tasks.

Datasets. We use the following set of established video benchmarks in our experiments:

UCF101 (Soomro, Zamir, and Shah 2012) is human action dataset containing 101 classes of indoor and outdoor actions.

HMDB51 (Kuehne et al. 2011) is a relatively small-scale dataset of 51 action classes with high intra-class diversity.

Kinetics400 (Carreira and Zisserman 2017) is a large-scale action dataset containing 400 human activity classes collected from YouTube.

Something-Something V2 (SSv2) (Goyal et al. 2017) consists of over 174 action classes, providing a challenging and diverse set of actions and environmental contexts.

NTU60 (Shahroudy et al. 2016) is a large-scale benchmark for human action recognition, containing over 56,000 action samples with 60 diverse action classes performed by multiple subjects in real-world indoor and outdoor environments.

Charades (Sigurdsson et al. 2018) is a multi-label action dataset containing 157 daily-life indoor actions in untrimmed videos.

Holistic Video Understanding (HVU) (Diba et al. 2020) is a large-scale benchmark addressing multi-label and multi-task video understanding of multiple semantic aspects, including scenes, objects, actions, attributes, and concepts.

DAVIS-2017 (Pont-Tuset et al. 2017) provides object-level pixel-wise annotations. The evaluation set contains 57 different objects.

CASIA-B (Yu, Tan, and Tan 2006) is a gait-recognition dataset for indoor walking videos of 124 subjects with 11 views from each.

JHMDB Pose (Jhuang et al. 2013) provides 31,838 annotated frames with 13 joints (shoulder, elbow, knee, etc.).

Implementation Details. Our framework is built on the Video Transformer Network (VTN) (Neimark et al. 2021) architecture. In our default experimental setting, we utilize a Vision Transformer (ViT) (Dosovitskiy et al. 2020) network as our image encoder $F(\cdot)$. We perform our self-supervised pertaining on unlabelled videos of Kinetics400. As inputs to our network, we feed 8 frames of resolution 224×224. During training, we use the common set of geometric augmentations (random crop, resize, flipping) and color jittering (random grayscale, color jittering, random erasing).

Experimental Setup. We focus our experiments on *fixed-feature* evaluation, *i.e.*, we keep the learned video representations fixed and evaluate features via retrieval and linear probing experiments. This focus is motivated by 1) the relevance to downstream video search applications, 2) the more direct probing of properties in the learned features, and 3) the better scalability of this approach to large-scale video processing. While we acknowledge that superior performance can often be achieved through a full finetuning of the network, it is often infeasible to fine-tune networks and reprocess massive amounts of high-dimensional video data in practice. In contrast, training and inference with a shallow model (*e.g.*, a linear classifier) on pre-extracted video features are considerably more scalable.

Ablations

We perform extensive ablations to verify our frame-wise pretext task formulation and illustrate the importance of shortcut removal for temporal self-supervision. All ablations

	OFL	TSP	Framewise	UCF101		HMDB51	
	$\mathcal{L}_{OFL}$	$\mathcal{L}_{TSP}$	Augment	R@1	Lin	R@1	Lin
init	✗	✗	✗	79.20	85.11	41.44	50.20
(a)	✗	✗	✓	80.15	85.31	43.35	52.42
(b)	✗	✓	✓	82.73	88.12	46.14	54.22
(c)	✓	✗	✓	83.90	89.29	48.98	58.28
(d)	✓	✓	✗	82.50	87.80	46.51	55.60
(e)	✓	✓	✓	**84.68**	**89.90**	**50.20**	**58.70**

Table 2: SSL Objectives Ablation Experiments. We performed experiments to investigate the effect of the various loss terms in our model (a-d) and the frame-wise augmentation strategy to remove shortcuts (e).

are performed with UCF101 pretraining, and we report results with linear probing and nearest-neighbor retrieval.

Influence of No-Shortcuts Temporal SSL. In Table 2, we analyze the influence of the temporal pretext tasks $\mathcal{L}_{OFL}$ and $\mathcal{L}_{TSP}$ and illustrate the importance of avoiding shortcuts in temporal SSL through our frame-wise augmentation strategy. We also report the performance when using the image SSL pre-trained backbone only (init) and when using contrastive learning on videos only (a). We observe only minor improvements from further contrastive training on videos in (a). In contrast, both our temporal pretext tasks ($\mathcal{L}_{OFL}$ and $\mathcal{L}_{TSP}$) (b)-(c) contribute significantly to downstream performance, increasing accuracy by **4-7%** across downstream tasks over the contrastive baselines. This highlights the importance of capturing temporal features for video representation learning. Finally, in (d), we see that removing the framewise augmentation strategy reduces the performance by a significant **2-4%**. This illustrates our key insight that temporal SSL methods have been plagued by shortcuts and did not achieve their full potential.

Method	UCF101		HMDB51	
	R@1	Lin	R@1	Lin
Clip-level	80.85	86.42	44.85	54.54
OFL	83.90 (↑4%)	89.29 (↑3%)	48.98 (↑9%)	58.28 (↑7%)
Clip-level	81.35	86.71	44.85	52.47
TSP	82.73 (↑2%)	88.12 (↑2%)	46.14 (↑3%)	54.22 (↑3%)

Table 3: Comparison with clip-level temporal tasks

Comparing Clip-Level vs. Frame-Level Pretext Tasks. In this study, we compare our frame-wise reformulation of temporal pretext tasks with traditional global (*i.e.*, clip-level) task formulations. The first two rows of Table 3 compare clip-level frame verification vs. our frame-level OFL task. We can see clear performance gains across all downstream protocols going from a clip-level task to a frame-level task. The last two rows of Table 3 suggest similar conclusions clip-level vs. frame-level skip rate prediction task. We find that transitioning from clip-level to frame-level tasks significantly raises the difficulty of pretext task, as indicated by respective pretext accuracies of 99% vs. 87% order-verification, and 96% vs. 55% the skip-prediction (numbers not shown in Table 3).

Out-of-order Frame Localization. OFL has two main design parameters to explore: (1) the percentage of out-of-order frames in the frame sequence, and (2) from where in the video to sample out-of-order frames (*i.e.*, how far from the correct position). We also report pretext task performance in terms of mAP to indicate the difficulty of OFL. In Table 4, we compare fixed outlier rates (b)-(c) to sampling outlier rates from a given interval at random (d)-(e). We find that randomizing the outlier rate provides clear benefits. Note the negative correlation between SSL and downstream performance, which validates our aim to increase the difficulty of temporal SSL. We use unrestricted sampling (*i.e.*, Table 5 (a)) in this experiment.

	Probability of	OFL	UCF101		HMDB51	
	outlier token	Task	R@1	Lin	R@1	Lin
(a)	0.0	-	82.73	88.12	46.14	54.22
(b)	0.25	87.39	83.40	88.89	47.30	55.91
(c)	0.50	79.88	83.10	88.40	46.80	55.10
(d)	$\mathcal{U}([0.0, 0.25])$	81.15	83.70	89.14	47.90	56.19
(e)	$\mathcal{U}([0.0, 0.50])$	76.76	**84.01**	**89.49**	**48.76**	**57.87**

Table 4: Ablations of outlier token probability in OFL task.

In Table 5, we explore the sampling position of outlier frames. We find that both a too-simple (b) and a too-difficult and potentially ambiguous OFL task (c) result in poor downstream performance. The best results are achieved by sampling the out-of-order frames within 64 frames of the in-order position.

	Out of Order frame	OFL	UCF101		HMDB51	
	Sampling Restriction	Task	R@1	Lin	R@1	Lin
(a)	Unrestricted	76.76	84.01	89.49	48.76	57.87
(b)	Min. Distance = 8	81.26	83.27	87.67	47.52	56.50
(c)	Max. Distance = 8	61.21	83.79	88.41	47.93	56.68
(d)	Max. Distance = 64	74.37	**84.68**	**89.90**	**50.20**	**58.70**

Table 5: Abl of replacement frame sampling in OFL task.

Time-varying Skiprate Prediction. We explore the design of TSP in Table 6, by using different subsets of $\{1, 2, 4, 8\}$ as skip rates for TSP. We observe that going from 2-way classification (b)-(c) to 3-way classification (d)-(e) consistently improves performance. This again suggests that challenging pretext task formulations help across downstream tasks.

	Playback	TSP	UCF101		HMDB51	
	Set	Task Acc.	R@1	Lin	R@1	Lin
(a)	Φ	-	83.09	88.05	47.58	56.70
(b)	{1,4}	72.10	83.62	88.64	48.30	57.43
(c)	{2,4}	71.55	83.53	88.76	48.30	57.55
(d)	**{1,4,8}**	55.14	**84.01**	**89.49**	48.76	57.87
(e)	{2,4,8}	54.66	84.00	89.23	**48.81**	**57.92**

Table 6: Ablations of various skip rates in TSP task.

Different Frame-wise Augmentations. We proposed using frame-wise augmentation to break the local patch similarity-based shortcuts in temporal pretext tasks. In this study, we

Spatial Cropping	Color Jittering	Horizontal Flipping	UCF101 R@1	UCF101 Lin	HMDB51 R@1	HMDB51 Lin
✓	✗	✗	**84.68**	**89.90**	**50.20**	**58.70**
✓	✓	✗	84.24	89.30	49.50	57.81
✓	✗	✓	84.04	88.97	48.08	56.61

Table 7: Ablation for frame-wise augmentations

compare the performance of various frame-wise augmentations. From Table 7, we can observe that adding color jittering to the random cropping decreases the performance by a small margin, however, adding the frame-wise horizontal flipping reduces the performance noticeably by 1-2%. In our default setting, we only use frame-wise cropping.

Static and Temporal Feature Analysis. Since our model learns and disentangles static and temporal features, we explore the influence of temporal features on our fused video representation in Table 8 (Kinetics400 pretraining). While static appearance features achieve a strong baseline performance, temporal features lead to consistent and significant improvements.

Features	UCF101 R@1	UCF101 Lin	HMDB51 R@1	HMDB51 Lin
Static only	84.53	89.11	50.91	55.90
Static + Temporal	**86.22**	**91.50**	**52.61**	**62.50**

Table 8: Influence of temporal features.

Different Backbones and Initializations. Following the prior work (Ranasinghe et al. 2022; Yao et al. 2020; Li et al. 2021), we use self-supervised weights learned on ImageNet-1k (Deng et al. 2009) as initialization for the image encoder. We compare various image encoders and their initialization methods in Table 9.(a) In our default setting, we report results with ViT-L initialized with MUGS (Zhou et al. 2022b) pre-training on ImageNet-1k.(b) We get similar/slightly better results using iBOT (Zhou et al. 2022a) pre-training.(c) Our results improve by 2-4% by using the iBOT pre-training on ImageNet-21k, which shows the advantage of our method to leverage the improvement in the image self-supervised methods.(d,e) show our results with ViT-B backbone initialized by MUGS (Zhou et al. 2022b) and DINO (Caron et al. 2021) pre-training on Imagenet-1k. Lastly, in(f), we show similar results to our default setting with computationally efficient SWIN-B backbone initialized with EsViT (Li et al. 2022) pre-training.

	Arch.	Pretraining Method	UCF101 R@1	UCF101 Lin	HMDB51 R@1	HMDB51 Lin
(a)	ViT-L	MUGS	84.68	89.90	50.20	58.70
(b)	ViT-L	iBOT	84.52	89.81	51.21	59.45
(c)	ViT-L	iBOT (21k)	86.40	91.60	54.37	63.30
(d)	ViT-B	MUGS	84.08	89.33	48.43	58.20
(e)	ViT-B	DINO	84.90	90.71	45.81	57.97
(f)	SWIN-B	EsViT	84.25	89.64	46.73	58.11

Table 9: Different initialization and backbone

Downstream Tasks

We compare our SSL representations learned on Kinetics400 to prior video SSL approaches on numerous video understanding benchmarks. Note that, while prior methods differ widely in terms of network architecture (among other factors), our results with ViT-B are directly comparable with the prior state-of-the-art approaches SVT, and VideoMAE.

Video Retrieval on UCF101 and HMDB51. We perform action retrieval experiments to demonstrate the suitability of our features for semantic video similarity search. Following prior works (Han, Xie, and Zisserman 2020a; Dave et al. 2022; Diba et al. 2021), the test set of each dataset is used as a *query-set*, and the training set is considered as a *search-set*. We report Top-1 and Top-5 retrieval accuracy in Table 10. Our method outperforms all prior works and achieves **3.3%** and **8.2%** absolute improvement of Top-1 accuracy on UCF101 and HMDB51.

Action Recognition on UCF, HMDB, and Kinetics. We report top-1 accuracies of linear probes and finetuning in Table 10. The results demonstrate that our method is highly competitive and outperforms most previous works on these standard benchmarks. This highlights its potential to achieve excellent results on videos found on the *web*.

Action Recognition on SSv2 and NTU60. Since these datasets are captured in controlled and shared settings, they exhibit less scene bias than datasets such as UCF, HMDB, and Kinetics and require a stronger temporal understanding to accurately classify actions. Our method outperforms the best previous methods in linear probing by an absolute margin of **3.2%** and **1.5%** on SSv2 and NTU60, respectively, demonstrating its suitability for action datasets captured in *controlled, real-world settings*.

Multi-Label Action Recognition on Charades. We follow the protocol of (Thoker et al. 2022), where the video-level multi-label prediction task is considered. We report linear multi-label classification performance in terms of mean average precision (mAP) in Table 10. Our method achieves an absolute improvement of **1.3%** over the previous state-of-the-art method, demonstrating its effectiveness in real-world *multi-label and untrimmed videos*.

Holistic Video Downstream on HVU. We perform linear classification on various semantic categories, including scenes, objects, events, attributes, and concepts, along with actions. As all semantics are in multi-label format, we report the performance in terms of mean average precision (mAP), as shown in Table 11. Our method consistently outperforms the prior state-of-the-art methods and achieves the best overall score for *holistic video understanding*.

Video Object Segmentation (VOS) on DAVIS. We follow the semi-supervised protocol of DAVIS-2017 (Pont-Tuset et al. 2017), where the object masks of the first frame of a video are given, and the task is to predict the masks in the rest of the frames. Table 12 shows a comparison with the prior works in the same protocol. All video SSL methods use a ViT-B architecture and are pre-trained on Kinetics400. Our method outperforms other video SSL methods. Some qualitative results are shown in Figure 5.

Human Pose Propagation We use validation videos of JHMDB and follow the evaluation protocol of (Li et al. 2019).

	Action Linear Classification						Action Finetuning		Video Retrieval			
Method	**U101**	**H51**	**K400**	**SSv2**	**NTU60**	**Charades**	**UCF101**	**HMDB51**	**UCF101**		**HMDB51**	
	Top-1	Top-1	Top-1	Top-1	Top-1	mAP (%)	Top-1	Top-1	R@1	R@5	R@1	R@5
He et al. 2020	65.4	-	34.5	7.4	16.0	8.1	83.5	-	-	-	-	-
Xu et al. 2019	-	-	-	-	-	-	64.9	29.5	14.1	30.3	-	-
Wang, Jiao, and Liu 2020	-	-	-	-	-	-	68.0	36.6	25.6	42.7	12.9	31.6
Asano et al. 2020	51.2	-	24.1	4.5	15.7	8.2	84.9	-	52.0	68.6	24.8	47.6
Han, Xie, and Zisserman 2020b	74.5	46.1	-	-	-	-	87.9	54.6	53.3	69.4	23.2	43.2
Wang et al. 2021b	-	-	-	-	-	-	70.3	40.5	16.8	33.4	8.2	25.9
Yao et al. 2020	79.6	42.2	61.9	-	-	-	88.3	55.6	-	-	-	-
Chen and He 2021	-	-	-	-	-	-	-	-	39.0	53.1	17.1	37.3
Qian et al. 2021b	89.8	58.3	66.1	-	-	-	92.9	67.9	-	-	-	-
Feichtenhofer et al. 2021	90.1	*61.1*	**68.3**	*24.5*	51.2	18.1	94.2	*72.1*	76.8	87.1	39.6	64.1
Pan et al. 2021	66.3	-	31.0	19.5	*51.6*	10.5	78.7	49.2	-	-	-	-
Hu et al. 2021	90.2	58.7	66.6	-	-	-	93.5	-	-	-	-	-
Qian et al. 2021a	63.2	33.4	-	-	-	-	79.1	47.6	39.6	57.6	18.8	39.2
Jenni and Jin 2021	74.1	47.5	-	-	-	-	83.7	60.8	64.3	80.9	29.5	55.8
Diba et al. 2021	75.4	47.3	63.4	-	-	-	89.1	55.7	55.4	70.9	24.6	45.1
Li et al. 2021	79.9	-	-	-	-	-	90.5	63.5	67.0	80.8	26.7	52.5
Wang et al. 2021a	37.9	-	7.6	12.2	22.6	9.6	88.4	61.7	29.0	47.3	11.8	30.1
Patrick et al. 2021 (+A)	75.7	-	38.6	11.9	38.2	8.5	89.3	60.0	62.8	79.0	26.1	51.7
Liang et al. 2022	-	-	-	-	-	-	83.8	57.1	35.3	49.9	14.0	32.8
Dave et al. 2022	69.9	-	19.9	10.9	33.5	11.1	84.1	53.6	56.9	72.2	24.1	45.8
Duan et al. 2022 (+D)		-	-	-	-	-	89.6	63.5	54.0	71.8	25.5	52.3
Khorasgani, Chen, and Shkurti 2022	72.3	41.8	-	-	-	-	83.2	52.2	66.7	77.3	25.3	49.8
Ranasinghe et al. 2022	*90.8*	57.8	*68.1*	18.3	50.8	*18.8*	93.7	67.2	*82.9*	*88.0*	*44.4*	*67.4*
Ni et al. 2022 (+F)	88.7	56.5	-	-	-	-	91.5	62.8	65.6	80.3	28.9	56.2
Xiao, Tighe, and Modolo 2022 (+F)	91.5	63.0	-	-	-	-	94.0	67.4	73.4		-	-
Tong et al. 2022	84.6	60.5	61.2	23.1	51.2	15.6	**96.1**	**73.3**	64.0	81.0	32.5	58.9
Jenni, Black, and Collomosse 2023 (+A)	88.0	58.2	-	-	-	-	91.8	71.2	70.7	-	40.5	-
Thoker, Doughty, and Snoek 2023	-	-	-	-	-	10.3	91.0	64.1	-	-	-	-
Ours (ViT-B)	91.0	60.8	68.2	27.0	52.3	20.1	94.2	64.1	85.1	93.1	49.4	74.0
Ours (ViT-L)	**91.5**	**62.5**	**68.3**	**27.7**	**53.1**	**20.4**	*94.3*	64.3	**86.2**	**93.4**	**52.6**	**75.1**

Table 10: Comparison with state-of-the-art methods on Action-related tasks. We report results for linear probing, full fine-tuning, and video retrieval. Methods are sorted chronologically. Methods using additional modality over RGB videos are shown with parenthesis, where A = audio, D = frame differences, and F = optical flow. R@1 and R@5 indicate video retrieval accuracy in Top-1 and Top-5 nearest neighbors, respectively. Best results are shown in bold, and second-best in italics.

Method	**Action**	**Obj.**	**Scene**	**Event**	**Attr.**	**Concept**	**Overall**
SVT	38.48	30.35	30.97	37.87	28.20	35.64	*33.58*
ρBYOL	33.20	25.82	28.40	35.50	24.16	33.21	30.05
VideoMAE	27.49	23.36	24.56	29.78	21.04	28.75	25.83
Ours(ViT-B)	38.65	33.46	34.24	40.23	30.99	38.38	**35.99**

Table 11: Downstream on HVU dataset (Diba et al. 2020)

In this protocol, the key points of the human pose are given for the first frame, and the task is to predict the location of those key points in the subsequent frames. We employ our Kinetics400 pre-trained video SSL model without any further tuning. The performance is measured as the percentage of correct key points (PCK@X), where X is a distance threshold from the ground-truth joint position. The results in Table 13 show the superior performance of our method compared to prior video SSL methods. Qualitative results are shown in Figure 6.

Human Gait Recognition. We evaluate our model on the CASIA-B gait recognition dataset with the standard split. The test set includes 50 subjects each with 10 sequences.

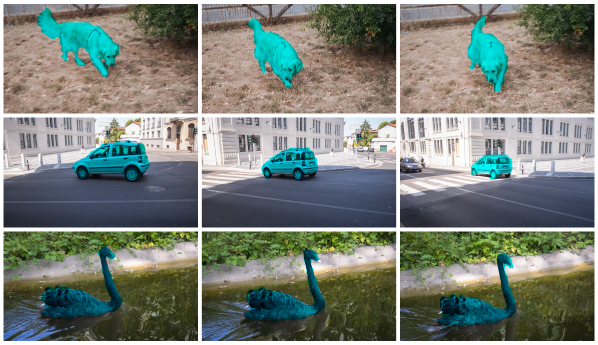

Figure 5: Qualitative Results on Video Object Segmentation (DAVIS-17): Uniformly sampled frames from sequences (top to bottom) dog and car-shadow and blackswan.

The test set is divided into two splits: *gallery* and *probe* set, and the goal is to retrieve the probe set. Ours and prior video SSL methods are pre-trained on Kinetics400, and no further learning is performed for gait recognition. We report rank-1

Pretraining	**J&F-Mean**	**J-Mean**	**F-Mean**
Feichtenhofer et al. 2022	53.5	52.6	54.4
VideoMAE (Tong et al. 2022)	53.8	53.2	54.4
Yang et al. 2022	*56.8*	*55.8*	*57.8*
SVT (Ranasinghe et al. 2022)	48.5	46.8	50.1
Ours (ViT-B)	**62.1**	**60.5**	**63.6**

Table 12: Video Object Segmentation on DAVIS-2017.

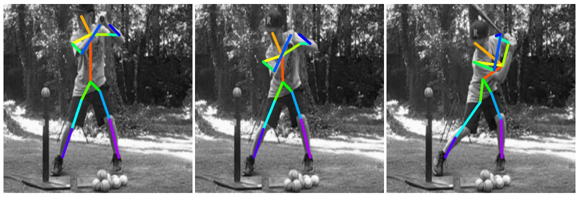

Figure 6: Qualitative results on JHMDB Pose Propagation.

accuracy for each split in Table 14. Our method outperforms prior works by a considerable margin.

Method	**NM**	**BG**	**CL**	**Mean**
ρBYOL	*90.65*	*80.51*	*28.59*	*66.58*
VideoMAE	65.30	57.21	21.40	47.97
Ours	**98.60**	**92.57**	**28.66**	**73.28**

Table 14: Gait Recognition on CASIA-B dataset.

Robustness to input perturbations Following (Schiappa et al. 2023), we adopt the robustness protocol for the video retrieval task on HMDB51, where, query-set and search-set videos are corrupted using random frame-independent perturbations like translation, Gaussian noise, or random JPEG compression. We report Top-1 retrieval accuracy for clean and perturbed videos for recent methods in Table 15, where our method achieves the smallest drop in performance across various perturbations. This superior robustness can be attributed to our model's capacity to learn and maintain the temporal correspondence between frames, even when they are independently perturbed. These qualities make our method highly suitable for *robust video retrieval* scenarios, where noise and perturbations are common challenges.

Conclusion

We have introduced a self-supervised approach for video representation learning. Our model extends a representation of static video frames with a transformer, which we train through self-supervision to capture temporal features. Importantly, we identified and addressed shortcuts in learning through temporal self-supervision and reformulated time-related learning tasks as more challenging frame-wise prediction tasks. We demonstrated the effectiveness of our approach on a wide variety of video understanding tasks for both generalization and robustness of the learned representations. We believe that our advancements in temporal self-supervision could inspire future work in other temporal data modalities (*i.e.*, time-series data) or multi-modal video understanding, *e.g.*, in combination with audio or language.

Pretraining	$PCK_{0.1}$	$PCK_{0.2}$	$PCK_{0.3}$	$PCK_{0.4}$	$PCK_{0.5}$
SVT	35.3	*62.66*	*77.6*	87.26	*91.94*
VideoMAE	*36.5*	62.1	76.7	*88.1*	91.5
Ours (ViT-B)	**43.1**	**69.7**	**81.6**	**88.3**	**92.7**

Table 13: Pose Propagation on JHMDB dataset.

Method	**Clean**	**Translation**	**Gaussian**	**JPEG**
SVT	44.40	43.21 (↓2.7)	41.80 (↓5.9)	41.52 (↓6.5)
ρBYOL	39.60	35.84 (↓9.6)	33.31 (↓15.9)	36.23 (↓8.5)
VideoMAE	32.50	26.72 (↓17.8)	26.22 (↓19.3)	26.61 (↓18.1)
Ours (ViT-B)	49.40	**48.43** (↓**2**)	**47.10** (↓**4.7**)	47.21 (↓4.4)
Ours (ViT-L)	52.60	51.50 (↓2.1)	50.06 (↓4.8)	**50.46** (↓**4.1**)

Table 15: Action Retrieval with perturbation. (↓n) shows relative drop in % compare to clean video retrieval.

References

Asano, Y.; Patrick, M.; Rupprecht, C.; and Vedaldi, A. 2020. Labelling unlabelled videos from scratch with multi-modal self-supervision. *Advances in Neural Information Processing Systems*, 33: 4660–4671.

Bai, Y.; Fan, H.; Misra, I.; Venkatesh, G.; Lu, Y.; Zhou, Y.; Yu, Q.; Chandra, V.; and Yuille, A. 2020. Can Temporal Information Help with Contrastive Self-Supervised Learning? *arXiv preprint arXiv:2011.13046*.

Benaim, S.; Ephrat, A.; Lang, O.; Mosseri, I.; Freeman, W. T.; Rubinstein, M.; Irani, M.; and Dekel, T. 2020. SpeedNet: Learning the Speediness in Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9922–9931.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9650–9660.

Carreira, J.; and Zisserman, A. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6299–6308.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *ICML*.

Chen, X.; and He, K. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15750–15758.

Chen, X.; Xie, S.; and He, K. 2021. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9640–9649.

Dave, I.; Gupta, R.; Rizve, M. N.; and Shah, M. 2022. TCLR: Temporal contrastive learning for video representation. *Computer Vision and Image Understanding*, 103406.

Dave, I. R.; Rizve, M. N.; Chen, C.; and Shah, M. 2023. TimeBalance: Temporally-Invariant and Temporally-Distinctive Video Representations for Semi-Supervised Action Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Diba, A.; Fayyaz, M.; Sharma, V.; Paluri, M.; Gall, J.; Stiefelhagen, R.; and Van Gool, L. 2020. Large scale holistic video understanding. In *European Conference on Computer Vision*, 593–610. Springer.

Diba, A.; Sharma, V.; Safdari, R.; Lotfi, D.; Sarfraz, S.; Stiefelhagen, R.; and Van Gool, L. 2021. Vi2clr: Video and image for visual contrastive learning of representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1502–1512.

Doersch, C.; Gupta, A.; and Efros, A. A. 2015. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, 1422–1430.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Duan, H.; Zhao, N.; Chen, K.; and Lin, D. 2022. TransRank: Self-supervised Video Representation Learning via Ranking-based Transformation Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3000–3010.

Feichtenhofer, C.; Fan, H.; Li, Y.; and He, K. 2022. Masked Autoencoders As Spatiotemporal Learners. In *Advances in Neural Information Processing Systems*.

Feichtenhofer, C.; Fan, H.; Xiong, B.; Girshick, R.; and He, K. 2021. A Large-Scale Study on Unsupervised Spatiotemporal Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3299–3309.

Goyal, R.; Kahou, S. E.; Michalski, V.; Materzyńska, J.; Westphal, S.; Kim, H.; Haenel, V.; Fruend, I.; Yianilos, P.; Mueller-Freitag, M.; Hoppe, F.; Thurau, C.; Bax, I.; and Memisevic, R. 2017. The ”something something” video database for learning and evaluating visual common sense. arXiv:1706.04261.

Han, T.; Xie, W.; and Zisserman, A. 2020a. 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*, 312–329. Springer.

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

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16000–16009.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9729–9738.

Hu, K.; Shao, J.; Liu, Y.; Raj, B.; Savvides, M.; and Shen, Z. 2021. Contrast and order representations for video self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7939–7949.

Jenni, S.; Black, A.; and Collomosse, J. 2023. Audio-Visual Contrastive Learning with Temporal Self-Supervision. *arXiv preprint arXiv:2302.07702*.

Jenni, S.; and Favaro, P. 2018. Self-supervised feature learning by learning to spot artifacts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2733–2742.

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

Jenni, S.; Meishvili, G.; and Favaro, P. 2020. Video Representation Learning by Recognizing Temporal Transformations. In *The European Conference on Computer Vision (ECCV)*.

Jhuang, H.; Gall, J.; Zuffi, S.; Schmid, C.; and Black, M. J. 2013. Towards understanding action recognition. In *International Conf. on Computer Vision (ICCV)*, 3192–3199.

Khorasgani, S. H.; Chen, Y.; and Shkurti, F. 2022. SLIC: Self-Supervised Learning With Iterative Clustering for Human Action Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16091–16101.

Kuehne, H.; Jhuang, H.; Garrote, E.; Poggio, T.; and Serre, T. 2011. HMDB: a large video database for human motion recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

Lee, H.-Y.; Huang, J.-B.; Singh, M.; and Yang, M.-H. 2017. Unsupervised representation learning by sorting sequences. In *Proceedings of the IEEE International Conference on Computer Vision*, 667–676.

Li, C.; Yang, J.; Zhang, P.; Gao, M.; Xiao, B.; Dai, X.; Yuan, L.; and Gao, J. 2022. Efficient Self-supervised Vision Transformers for Representation Learning. *International Conference on Learning Representations (ICLR)*.

Li, R.; Zhang, Y.; Qiu, Z.; Yao, T.; Liu, D.; and Mei, T. 2021. Motion-focused contrastive learning of video representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2105–2114.

Li, X.; Liu, S.; De Mello, S.; Wang, X.; Kautz, J.; and Yang, M.-H. 2019. Joint-task self-supervised learning for temporal correspondence. *Advances in Neural Information Processing Systems*, 32.

Liang, H.; Quader, N.; Chi, Z.; Chen, L.; Dai, P.; Lu, J.; and Wang, Y. 2022. Self-supervised spatiotemporal representation learning by exploiting video continuity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 1564–1573.

Misra, I.; Zitnick, C. L.; and Hebert, M. 2016. Shuffle and learn: unsupervised learning using temporal order verification. In *European Conference on Computer Vision*, 527–544. Springer.

Neimark, D.; Bar, O.; Zohar, M.; and Asselmann, D. 2021. Video Transformer Network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, 3163–3172.

Ni, J.; Zhou, N.; Qin, J.; Wu, Q.; Liu, J.; Li, B.; and Huang, D. 2022. Motion Sensitive Contrastive Learning for Self-supervised Video Representation. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Pan, T.; Song, Y.; Yang, T.; Jiang, W.; and Liu, W. 2021. Videomoco: Contrastive video representation learning with temporally adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11205–11214.

Patrick, M.; Asano, Y. M.; Kuznetsova, P.; Fong, R.; Henriques, J. F.; Zweig, G.; and Vedaldi, A. 2021. Multi-modal Self-Supervision from Generalized Data Transformations.

Pont-Tuset, J.; Perazzi, F.; Caelles, S.; Arbeláez, P.; Sorkine-Hornung, A.; and Van Gool, L. 2017. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*.

Qian, R.; Li, Y.; Liu, H.; See, J.; Ding, S.; Liu, X.; Li, D.; and Lin, W. 2021a. Enhancing Self-supervised Video Representation Learning via Multi-level Feature Optimization. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

Qian, R.; Meng, T.; Gong, B.; Yang, M.-H.; Wang, H.; Belongie, S.; and Cui, Y. 2021b. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6964–6974.

Ranasinghe, K.; Naseer, M.; Khan, S.; Khan, F. S.; and Ryoo, M. S. 2022. Self-supervised video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2874–2884.

Schiappa, M. C.; Biyani, N.; Kamtam, P.; Vyas, S.; Palangi, H.; Vineet, V.; and Rawat, Y. S. 2023. A Large-Scale Robustness Analysis of Video Action Recognition Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14698–14708.

Shahroudy, A.; Liu, J.; Ng, T.-T.; and Wang, G. 2016. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1010–1019.

Sigurdsson, G. A.; Gupta, A.; Schmid, C.; Farhadi, A.; and Alahari, K. 2018. Charades-Ego: A Large-Scale Dataset of Paired Third and First Person Videos. *CoRR*, abs/1804.09626.

Soomro, K.; Zamir, A. R.; and Shah, M. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Thoker, F. M.; Doughty, H.; Bagad, P.; and Snoek, C. G. 2022. How Severe is Benchmark-Sensitivity in Video Self-Supervised Learning? In *European Conference on Computer Vision*, 632–652. Springer.

Thoker, F. M.; Doughty, H.; and Snoek, C. G. 2023. Tubelet-Contrastive Self-Supervision for Video-Efficient Generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 13812–13823.

Tong, Z.; Song, Y.; Wang, J.; and Wang, L. 2022. VideoMAE: Masked Autoencoders are Data-Efficient Learners for Self-Supervised Video Pre-Training. In *Advances in Neural Information Processing Systems*.

Wang, G.; Zhou, Y.; Luo, C.; Xie, W.; Zeng, W.; and Xiong, Z. 2021a. Unsupervised visual representation learning by tracking patches in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2563–2572.

Wang, J.; Gao, Y.; Li, K.; Jiang, X.; Guo, X.; Ji, R.; and Sun, X. 2021b. Enhancing Unsupervised Video Representation Learning by Decoupling the Scene and the Motion. In *The AAAI Conference on Artificial Intelligence (AAAI)*.

Wang, J.; Jiao, J.; and Liu, Y.-H. 2020. Self-supervised Video Representation Learning by Pace Prediction. In *The European Conference on Computer Vision (ECCV)*.

Wei, D.; Lim, J. J.; Zisserman, A.; and Freeman, W. T. 2018. Learning and using the arrow of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8052–8060.

Xiao, F.; Tighe, J.; and Modolo, D. 2022. MaCLR: Motion-aware contrastive Learning of representations for videos. In *The European Conference on Computer Vision (ECCV)*.

Xu, D.; Xiao, J.; Zhao, Z.; Shao, J.; Xie, D.; and Zhuang, Y. 2019. Self-supervised spatiotemporal learning via video clip order prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 10334–10343.

Yang, C.; Xu, Y.; Dai, B.; and Zhou, B. 2020. Video representation learning with visual tempo consistency. *arXiv preprint arXiv:2006.15489*.

Yang, H.; Huang, D.; Wen, B.; Wu, J.; Yao, H.; Jiang, Y.; Zhu, X.; and Yuan, Z. 2022. Self-supervised Video Representation Learning with Motion-Aware Masked Autoencoders. *arXiv preprint arXiv:2210.04154*.

Yao, T.; Zhang, Y.; Qiu, Z.; Pan, Y.; and Mei, T. 2020. SeCo: Exploring Sequence Supervision for Unsupervised Representation Learning. *arXiv preprint arXiv:2008.00975*.

Yu, S.; Tan, D.; and Tan, T. 2006. A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 4, 441–444. IEEE.

Zhou, J.; Wei, C.; Wang, H.; Shen, W.; Xie, C.; Yuille, A.; and Kong, T. 2022a. iBOT: Image BERT Pre-Training with Online Tokenizer. *International Conference on Learning Representations (ICLR)*.

Zhou, P.; Zhou, Y.; Si, C.; Yu, W.; Ng, T. K.; and Yan, S. 2022b. Mugs: A Multi-Granular Self-Supervised Learning Framework. In *arXiv preprint arXiv:2203.14415*.