Self-supervised Video Transformer

Kanchana Ranasinghe[1,2] Muzammal Naseer[2,3] Salman Khan[2,3]
Fahad Shahbaz Khan[2,4] Michael S. Ryoo[1]

[1]Stony Brook University [2]Mohamed bin Zayed University of AI
[3]Australian National University [4]Linköping University

kranasinghe@cs.stonybrook.edu

Abstract

In this paper, we propose self-supervised training for video transformers using unlabeled video data. From a given video, we create local and global spatiotemporal views with varying spatial sizes and frame rates. Our self-supervised objective seeks to match the features of these different views representing the same video, to be invariant to spatiotemporal variations in actions. To the best of our knowledge, the proposed approach is the first to alleviate the dependency on negative samples or dedicated memory banks in Self-supervised Video Transformer (SVT). Further, owing to the flexibility of Transformer models, SVT supports slow-fast video processing within a single architecture using dynamically adjusted positional encoding and supports long-term relationship modeling along spatiotemporal dimensions. Our approach performs well on four action recognition benchmarks (Kinetics-400, UCF-101, HMDB-51, and SSv2) and converges faster with small batch sizes. Code is available at: https://git.io/J1juJ

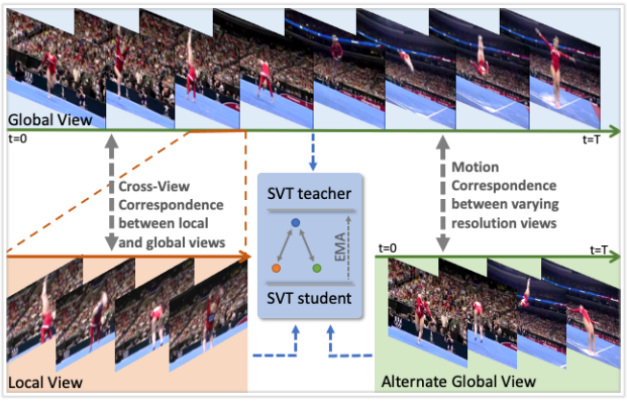

Figure 1. Our Self-supervised Video Transformer (SVT) learns cross-view and motion correspondences by jointly matching video clips sampled with varying field of view and temporal resolutions. Specifically, *Global* views (top and bottom right) with different temporal resolutions as well as *Local* views (bottom left) from different spatiotemporal windows are sampled. The representations of these multiple views are matched in a student-teacher framework to learn cross-view and motion correspondences (middle block). The proposed self-supervised framework can learn high-quality spatiotemporal features while converging faster.

1. Introduction

Self-supervised learning enables extraction of meaningful representations from unlabeled data, alleviating the need for expensive annotations. Recent self-supervised methods perform on-par with supervised learning for certain vision tasks [11, 12, 16, 37]. The necessity of self-supervised learning is even greater in domains such as video analysis where annotations are more expensive [40, 44, 63, 65].

At the same time, the emergence of vision transformers (ViTs) [26] and their successful adoption to different computer vision tasks including video understanding [4, 9, 29, 53, 67, 68] within the supervised setting shows their promise in the video domain. In fact, recent works using simple ViT backbones [9] surpass convolutional neural networks (CNN) for supervised video analysis with reduced compute. Motivated by the ability of self-attention to model long-range dependencies, we propose a simple yet effective method to train video transformers [9] in a self-supervised manner. This process uses spatial and temporal context as a supervisory signal (from unlabelled videos) to learn motion, scale, and viewpoint invariant features.

Many existing self-supervised representation learning methods on videos [64, 65, 88] use contrastive learning objectives which can require larger batch sizes, longer training regimes, careful negative mining and dedicated memory banks. Further, the contrastive objectives require careful temporal sampling [64] and multiple networks looking at similar/different clips to develop attract/repel loss formulations [65]. In contrast, we propose to learn self-supervised features from unlabelled videos via self-distillation [72] by a twin network strategy (student-teacher models) [13, 34].

Our proposed approach, Self-supervised Video Transformer (SVT), trains student and teacher models with a similarity objective [13] that matches the representations along spatial and temporal dimensions by space and time attention [9]. We achieve this by creating spatiotemporal positive views that differ in spatial sizes and are sampled at different time frames from a single video (Fig. 1). During training, teacher video transformer parameters are updated as an exponential moving average of the student video transformer. Both of these networks process different spatiotemporal views of the same video and our objective function is designed to predict one view from the other in the feature space. This allows SVT to learn robust features that are *invariant* to spatiotemporal changes in videos while generating discriminative features across videos [34]. SVT does not depend on negative mining or large batch sizes and remains computationally efficient as it converges within only a few epochs ($\approx$ 20 on Kinetics-400 [14]).

In addition to the above advantages, our design allows the flexibility to model varying time-resolutions and spatial scales within a unified architecture. This is a much desired feature for video processing since real-world actions can occur with varying temporal and spatial details. Remarkably, current self-supervision based video frameworks [64, 87] operate on fixed spatial and temporal scales which can pose difficulties in modeling the expressivity and dynamic nature of actions. We note that convolutional backbones used in these approaches lack the adaptability to varying temporal resolutions (due to fixed number of channels) and thus require dedicated networks for each resolution [30, 45]. To address this challenge, the proposed SVT uses dynamically adjusted positional encodings to handle varying temporal resolutions within the same architecture. Further, the self-attention mechanism in SVT can capture both local and global long-range dependencies across both space and time, offering much larger receptive fields as compared to traditional convolutional kernels [57].

The main contributions in this work are as follows:

- We introduce a novel mechanism for self-supervised training of video transformers by exploiting spatiotemporal correspondences between varying fields of view (global and local) across space and time (Sec. 3.2).
- Self-supervision in SVT is performed via a joint motion and crossview correspondence learning objective. Specifically, global and local spatiotemporal views with varying frame rates and spatial characteristics (Sec. 3.2.1 Sec. 3.2.2) are matched by our motion and crossview correspondences in the latent space.
- A unique property of our architecture is that it allows slow-fast training and inference using a single video transformer. To this end, we propose to use dynamic positional encoding within SVT to handle variable frame rate inputs generated from our sampling strategy (Sec. 3.2.3).

Our extensive experiments and results on various video datasets including Kinetics-400 [14], UCF-101 [69], HMDB-51 [49], and SSv2 [33] show state-of-the-art transfer of our self-supervised features using only RGB data. Further, our method shows a rapid convergence rate.

2. Related Work

Transformers in Vision. Since the initial success of transformers in natural language processing (NLP) tasks [21, 77], they have emerged as a competitive architecture for various other domains [46]. Among vision tasks, the initial works focused on a combination of convolutional and self-attention based architectures [10, 83, 85, 90]. A convolution free variant, vision transformer (ViT) [26], achieved competitive performance on image classification tasks. While earlier works proposing ViT [26] depended on large-scale datasets, more recent efforts achieve similar results with medium-scale datasets using various augmentation strategies [71, 76]. Later architectures also explore improving computational efficiency of ViTs focusing on transformer blocks [52, 67, 89]. ViTs have also been adopted for video classification tasks [4, 9, 29, 67, 68]. Our work builds on the TimeSformer backbone [9], a direct adaptation of standard ViTs using separate attention across dimensions.

Self-supervised Learning in Images. Early image-based self-supervised learning work focused on pretext tasks that require useful representations to solve [24, 25, 48, 58, 61, 78, 91]. However, recently contrastive methods have dominated self-supervised learning [6, 13, 16, 17, 19, 27, 37, 38, 50, 55, 74, 75] . These approaches generally consider two views of a single sample (transformed through augmentations) and pull them (positives) together while pushing away from all other (negative) samples in representation space [6, 59]. Key drawbacks of these methods are the necessity for careful mining of positive / negative samples [75] and reliance on large numbers of negative samples (leading to large batch sizes [16] or memory banks [37]). While clustering methods improve on this using cluster targets [3, 5, 7, 11, 12, 41, 73, 88], recent regression based methods that predict alternate representations [13, 34] eliminate the need for sample mining and negative samples. In particular, Caron *et al*. [13] explore predicting spatially local-global correspondences with ViT backbones within the image domain, which we extend in our work to the video domain with suitable improvements.

Self-supervised Learning in Videos. While self-supervised learning in videos were initially dominated by approaches based on pretext tasks unique to the video domain [2, 32, 43, 54, 56, 62, 70, 79–81, 84], recent work focuses more on contrastive losses similar to the image domain [22, 31, 35, 36, 64, 65]. A combination of previous pretext tasks over multiple modalities with cross-modality distillation is presented in [63]; SVT differs in how our

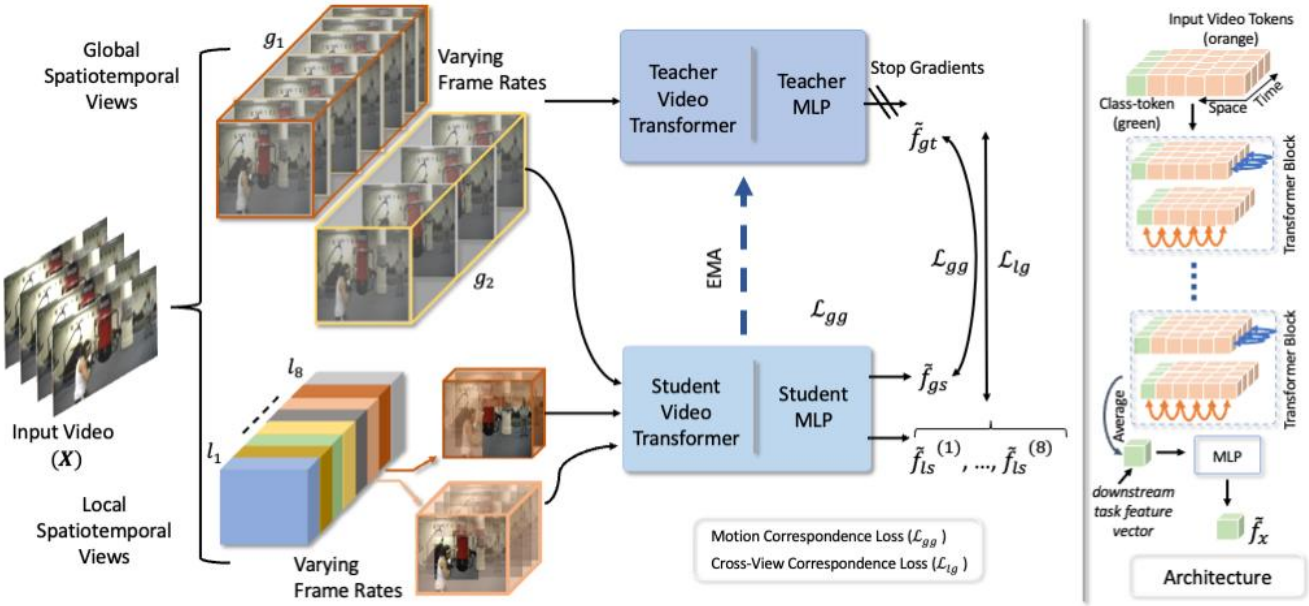

Figure 2. Our spatiotemporal sampling generates global and local views from a given input video. Global views contain different frame rates and spatial characteristics controlled by sampling strategy and combinations of augmentations. Local views have varying frame rates as well as limited fields of view due to random cropping. One global view is randomly selected and passed through the teacher model to generate a target while other global and local views are passed through the student model. Network weights are then updated by matching the online student local (*cross-view correspondences*) and global (*motion correspondences*) views to the target teacher global view. We use a standard ViT backbone with separate space-time attention [9] followed by an MLP for predicting target features from online features.

self-distillation operates within a single modality and network. The idea of varying resolution along temporal dimension is explored in [15, 40]. They use contrastive losses between different videos at same resolution for speed consistency or the same video at different resolutions for appearance consistency. Unlike these works, we jointly vary spatial and temporal resolutions and use a predictive objective as self-supervision. The idea of views with limited locality is also explored in [8, 20, 65]. While [8] uses views of varying locality for disentangling the representation space into temporally local and global features using contrastive objectives, our approach uses view locality to learn correspondences along and across dimensions with our predictive objective. A similar predictive objective with temporal locality constrained views is used in [65] and contrastive losses with spatial local-global crops is used in [20]; however our approach focuses on spatio-temporal constraints extending correspondences across dimensions, uses a single shared network for processing alternate views, and additionally combines varying resolutions to generate our alternate views exploiting unique ViT architectural features.

3. Self-supervised Video Transformer

In this section, we discuss our Self-supervised Video Transformer (SVT) approach. Unlike contrastive methods, we process two clips from the same video with varying spatial-temporal characteristics (Sec. 3.2) avoiding the need for negative mining or memory banks. Our loss formulation matches the representations from both dissimilar clips to enforce invariance to motion and spatial changes for the same action sequence. A naive objective enforcing invariance would collapse all representations to be the same, however we use a teacher-student network pair where the former acts as a more stable target for the later, enabling convergence of the online student network to learn discriminative representations [34]. This approach simultaneously incorporates rich spatiotemporal context in the representations while keeping them discriminative. In the following, we first introduce the overall architecture of SVT in Sec. 3.1 followed by the self-supervised learning process in Sec. 3.2, our objective functions in Sec. 3.3 and inference in Sec. 3.4.

3.1. SVT: Architecture

We apply separate attention along temporal and spatial dimensions of input video clips using a video transformer [9]. Consider a video $\boldsymbol{X} = \{\boldsymbol{x}_t\}_{t=1}^{N}$, where N represents the number of frames. We define a *clip* (also termed *view* interchangeably) as a subset of these N frames selected through a sampling strategy. We define H, W, T to be the height, width, and number of frames respectively for the sampled clip. Our sampling methodology (Fig. 2) generates two types of clips, *global* ($\boldsymbol{g}$) and *local* ($\boldsymbol{l}$) spatiotemporal views. Both $\boldsymbol{g}$ and $\boldsymbol{l}$ are subsets of the video frame set $\boldsymbol{X}$, with views $\boldsymbol{g} = \{\boldsymbol{x}'_t\}_{t=1}^{K_g}$, $\boldsymbol{l} = \{\boldsymbol{x}''_t\}_{t=1}^{K_l}$, and $|K_l| \leq |K_g|$.

Global views are generated by uniformly sampling a variable number of frames from a randomly selected 90% portion of a video's time axis. We generate two such global spatiotemporal views $(\boldsymbol{g}_1, \boldsymbol{g}_2)$ at low $(T = 8)$ and high $(T = 16)$ frame rates and spatial resolution $H = W = 224$. **Local views** are generated by uniformly sampling frames from a randomly selected video region covering $1/8^{th}$ of the time axis and $\approx 40\%$ area along spatial axes. We generate eight such local spatiotemporal views with $T \in \{2, 4, 8, 16\}$ and spatial resolution fixed at $H = W = 96$. Specifically, we randomly sample two global $(\boldsymbol{g}_1, \boldsymbol{g}_2)$ and eight local $(\boldsymbol{l}_1, \ldots, \boldsymbol{l}_8)$ spatiotemporal views. Note that both spatial and temporal dimensions within our sampled views differ from those of the original video. We introduce the channel dimension, C, which is fixed at 3 for RGB inputs considered in our case. Our SVT, comprising of 12 encoder blocks, processes each sampled clip of shape $(C \times T \times W \times H)$, where $W \leq 224$, $H \leq 224$ and $T \leq 16$ (different for each clip). Our network architecture (Fig. 2) is designed to process such varied resolution clips during both training and inference stages within a single architecture (Sec. 3.2.3).

During training, we divide each frame within a view into patches [26]. Thus, for a given view of maximum size $H = W = 224$ and $T = 16$, each SVT encoder block processes a maximum of 196 spatial and 16 temporal tokens, and the embedding dimension of each token is $\mathbb{R}^{768}$ [26]. Since the maximum number of spatial and temporal tokens vary due to variable dimensions in our views, we deploy dynamic positional encoding (Sec. 3.2.3) to account for any missing tokens for views of size $W < 224$, $H < 224$ and $T < 16$. Note the minimum spatial and temporal sizes in our proposed views are $H = W = 96$ and $T = 2$, respectively. In addition to these input spatial and temporal tokens, we use a single classification token as the feature vector within the architecture [21,26]. This classification token represents the common features learned by the SVT along spatial and temporal dimensions of a given video. Finally, we use a multi-layer perceptron (MLP) as a projection head over the classification token from the final encoder block [13, 34]. We define the output of our projection head as $\boldsymbol{f}$.

As illustrated in Fig. 2, our overall approach uses a teacher-student setup inspired from [13, 34] for self-distillation [72]. Our teacher model is an exact architectural replica of the student model.

3.2. SVT: Self-supervised Training

We train SVT in a self-supervised manner by predicting the different views (video clips) with varying spatiotemporal characteristics from each other in the feature space of student and teacher models. To this end, we adopt a simple routing strategy that randomly selects and passes different views through the teacher and student models. The teacher SVT processes a given global spatiotemporal view to produce a feature vector, $\boldsymbol{f}_{g_t}$, which is used as the target label, while the student SVT processes local and global spatiotemporal views to produce feature vectors, $\boldsymbol{f}_{g_s}$, and $\boldsymbol{f}_{l_s}^{(1)}, ..., \boldsymbol{f}_{l_s}^{(8)}$, which are matched to the target feature $\boldsymbol{f}_{g_t}$ through our proposed loss (Eq. 1). During each training step, we update the student model weights via backpropagation while teacher weights are updated as an exponential moving average (EMA) of the student weights [13].

Our motivation for predicting such varying views of a video is that it leads to modeling the contextual information defining the underlying distribution of videos by learning motion correspondences (global to global spatiotemporal view matching) and cross-view correspondences (local to global spatiotemporal view matching) (Fig. I). This makes the model invariant to motion, scale and viewpoint variations. Thus, our self-supervised video representation learning approach depends on closing the gap between feature representations of different spatiotemporal views from the same video using a self-distillation mechanism. Next, we explain how motion correspondences and cross-view correspondences are learned, followed by our loss formulation.

3.2.1 Motion Correspondences

A defining characteristic of a video is the frame rate. Varying the frame rate can change motion context of a video (*e.g.*, walking slow vs walking fast) while controlling nuanced actions (*e.g.*, subtle body-movements of walking action). In general, clips are sampled from videos at a fixed frame rate [64, 87]. However, given two clips of varying frame rate (different number of total frames for each clip), predicting one from the other in feature space explicitly involves modeling the motion correspondences (MC) of objects across frames. Further, predicting subtle movements captured at high frame rates will force a model to learn motion related contextual information from a low frame rate input. We model this desired property into our training by matching global to global spatiotemporal views. Refer to Appendix A for further details.

3.2.2 Cross-View Correspondences

In addition to learning motion correspondences, our training strategy aims to model relationships across spatiotemporal variations as well by learning cross-view correspondences (CVC). The cross-view correspondences are learned by matching the local spatiotemporal views processed by our student SVT $(\boldsymbol{f}_{l_s}^{(i)} : i \in [1, 8])$ with a global spatiotemporal view representation processed by our teacher SVT model $(\boldsymbol{f}_{g_t})$. Our local views cover a limited portion of videos along both spatial and temporal dimensions.

Our intuition is that predicting a global spatiotemporal view of a video from a local spatiotemporal view in the latent space forces the model to learn high-level contex-

tual information by modeling, **a)** spatial context in the form of possible neighbourhoods of a given spatial crop, and **b)** temporal context in the form of possible previous or future frames from a given temporal crop. Note that in the cross-view correspondences, we predict a global view frame using all frames of a local view by our similarity objective (Eq. 3).

3.2.3 Dynamic Positional Embedding

Vision transformers [26] require inputs to be converted to sequences of tokens, which allows efficient parallel processing. Positional encoding is used to model ordering of these sequences [57]. Interestingly, positional encoding also allows ViT to process variable input resolution by interpolating the positional embedding for the missing tokens. As mentioned earlier, our motion and cross-view correspondences involve varying spatial and temporal resolutions which results in variable spatial and temporal input tokens during training (Sec. 3.1). We use this property of positional encoding to our advantage by accommodating varying spatial and temporal tokens in our proposed training mechanism. In implementing this, during training we use a separate positional encoding vector for spatial and temporal dimensions and fix these vectors to the highest resolution across each dimension. Similar to [26], our positional encoding is a learned vector. We vary the positional encoding vectors through interpolation during training to account for the missing spatial or temporal tokens at lower frame rate or spatial size. This allows our single SVT model to process inputs of varying resolution while also giving the positional embedding a dynamic nature which is more suited for different sized inputs in the downstream tasks. During slow-fast inference (Sec. 3.4) on downstream tasks, the positional encoding is interpolated to the maximum frame count and spatial resolution used across all views.

We note that our learned positional encoding is implicitly tied to frame number to cue the relative ordering of the sampled frames. Given the varying frame rates of views, it does not encode the exact time stamp (frame rate information). We hypothesize that despite not differentiating frame rates, cuing frame order is sufficient for SVT training.

3.2.4 Augmentations

In addition to our sampling strategy (temporal dimension augmentations), we also apply standard image augmentations to the spatial dimension, *i.e.*, augmentations are applied to the individual frames sampled for each view. We follow temporally consistent spatial augmentations [64] where the same randomly selected augmentation is applied equally to all frames belonging to a single view. The standard augmentations used include random color jitter, gray scaling, Gaussian blur, and solarization. We also apply random horizontal flips to datasets not containing flip equivariant classes (*e.g.*, walking left to right).

3.3. SVT Loss

We enforce motion and cross-view correspondences by matching our proposed spatiotemporal views within the feature space. Specifically, we match global to global views to learn motion and local to global views to learn cross-view correspondences by minimizing the following objective:

$$\mathcal{L} = \mathcal{L}_{lg} + \mathcal{L}_{gg}. \tag{1}$$

The global and local spatiotemporal views are passed though the student and teacher models to get the corresponding feature outputs $\boldsymbol{f}_g$ and $\boldsymbol{f}_l$. These feature vectors are normalized to obtain $\tilde{\boldsymbol{f}}$ as follows:

$$\tilde{\boldsymbol{f}}[i] = \frac{\exp(\boldsymbol{f}[i])/\tau}{\sum_{i=1}^{n} \exp(\boldsymbol{f}[i])/\tau},$$

where τ is a temperature parameter used to control sharpness of the exponential function [13] and $\tilde{\boldsymbol{f}}[i]$ is each element of $\tilde{\boldsymbol{f}} \in \mathbb{R}^n$.

Motion Correspondence Loss: We forward pass a global view through the teacher SVT serving as the target feature which is compared with an alternate global view processed by the student SVT to obtain a loss term (Eq. 2). This loss measures the difference in motion correspondences between these two global views.

$$\mathcal{L}_{gg} = -\tilde{\boldsymbol{f}}_{g_t} \cdot \log(\tilde{\boldsymbol{f}}_{g_s}), \tag{2}$$

where, $\tilde{\boldsymbol{f}}_{g_s}$ and $\tilde{\boldsymbol{f}}_{g_t}$ are the feature outputs of different global spatiotemporal views from the student and teacher network respectively and $[\cdot]$ is dot product operator.

Cross-view Correspondence Loss: All local spatiotemporal views are passed through the student SVT model and mapped to a global spatiotemporal view from the teacher SVT model to reduce the difference in feature representation, learning cross-view correspondences (Eq. 3).

$$\mathcal{L}_{lg} = \sum_{i=1}^{k} -\tilde{\boldsymbol{f}}_{g_t} \cdot \log(\tilde{\boldsymbol{f}}_{l_s}^{(i)}), \tag{3}$$

where the sum is performed over k different local spatiotemporal views ($k = 8$ used consistently across all experiments) and $\tilde{\boldsymbol{f}}_{l_s}^{(i)}$ are the feature outputs for i^{th} local view.

Convergence: Given our two separate student (θ) and teacher (ξ) networks, let us view our overall loss, L as a function of their learnable parameters, $L_{\theta,\xi}$. There exists a concern of collapse to a trivial solution (teacher and student outputs always equal a constant) during training. However, we note that SVT parameters do not converge to such a minimum over $L_{\theta,\xi}$ because: **a)** The SVT teacher parameter updates are not in the direction of $\nabla_\xi L_{\theta,\xi}$ since $\xi_{t+1} \leftarrow \tau\xi_t + (1-\tau)\theta_t$ for $\tau \in [0, 1]$ (EMA update). **b)**

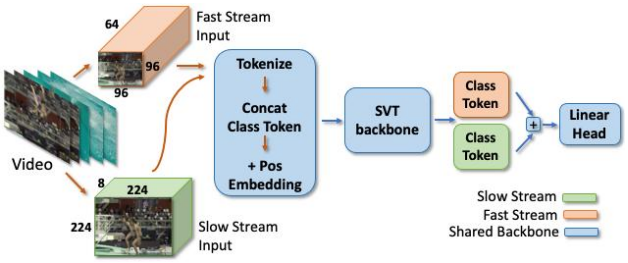

Figure 3. Slow-Fast Inference: we uniformly sample two clips of the same video at resolutions $(8, 224, 244)$ and $(64, 96, 96)$, pass through a shared network, and generate two different feature vectors (class tokens). These vectors are combined in a deterministic manner (with no learnable parameters), e.g. summation, to obtain a joint vector that is fed to the downstream task classifier.

SVT's gradient descent on $L_{\theta,\xi}$ does not act jointly over (θ, ξ). This is similar to BYOL [34] where such a loss acts on the outputs of student and teacher networks. Additionally, as suggested in [13], we also use centering and sharpening of teacher outputs to further facilitate convergence.

3.4. SVT: Slow-Fast Inference

Slow-Fast inference refers to using two different video clips with *high spatial but low temporal* and *low spatial but high temporal* resolutions. This allows capturing finer-information across each dimension with minimal increase in computation. Recent methods [30, 45] deploy such inference but use multiple network architectures for processing videos at different resolutions. However, our dynamic positional encodings allow Slow-Fast inference within our single SVT model (Sec. 3.2.3) as illustrated in Fig. 3. We use this on downstream tasks for improved performance.

4. Experiments

4.1. Experimental Setup and Protocols

Datasets: We use the Kinetics-400 data [14] (train set) for the self-supervised training phase of SVT. We use its validation set for evaluation. Additionally, we evaluate on three downstream datasets, UCF-101 [69], HMBD-51 [49], and Something-Something v2 (SSv2) [33].

Self-supervised Training: We train our models for 20 epochs on the train set of Kinetics-400 dataset [14] without any labels using a batch size of 32 across 4 NVIDIA-A100 GPUs. This batch size refers to the number of unique videos present within a given batch. We randomly initialize weights relevant to temporal attention while spatial attention weights are initialized using a ViT model trained in a self-supervised manner over the ImageNet-1k dataset [66]. We follow this initialization setup to obtain faster space-time ViT convergence similar to the supervised setting [9]. We use an Adam optimizer [47] with a learning rate of $5e{-}4$ scaled using a cosine schedule with linear warmup for 5 epochs [18, 71]. We also use weight decay scaled from 0.04 to 0.1 during training. Our code builds over the training frameworks from [9, 13, 28, 86].

Downstream Tasks: We perform two types of evaluations on our downstream task of action recognition for each dataset, **a) Linear:** We train a linear classifier over our pre-trained SVT backbone (which is frozen during training) for 20 epochs with a batch size of 32 on a single NVIDIA-V100 GPU. We use SGD with an initial learning rate of $8e-3$, a cosine decay schedule and momentum of 0.9 similar to recent work [13, 64]; **b) Fine-tuning:** We replace the projection head over the SVT with a randomly initialized linear layer, initialize the SVT backbone with our pre-trained weights, and train the network end-to-end for 15 epochs with a batch size of 32 across 4 NVIDIA-V100 GPUs. We use SGD with a learning rate of $5e-3$ decayed by a factor of 10 at epochs 11 and 14, momentum of 0.9, and weight decay of $1e-4$ following [9].

During both training of linear classifier and fine-tuning of SVT, we sample two clips of varying spatiotemporal resolution from each video. During evaluation, we follow our proposed slow-fast inference strategy (Sec. 3.4). We use two clips per video sampled at different spatiotemporal resolutions $(T, W, H) \in \{(8, 224, 224), (64, 96, 96)\}$ with 3 spatial crops each for testing (6 clips in total). This is computationally more efficient in comparison to recent works [64, 65] that uniformly sample 10 clips at similar or high resolutions from full-length videos with 3 crops each for testing (total of 30 clips per video).

4.2. Results

We compare SVT with state-of-the-art approaches (trained on RGB input modality for fair comparison) for the downstream task of action recognition.

UCF-101 & HMDB-51: Our method out-performs state-of-the-art for UCF-101 and is on-par for HMDB-51 (Tab. 1). While CORP [39] exhibits higher performance on HMDB-51, we highlight how SVT: **a)** is pretrained for a much shorter duration (20 epochs) with smaller batch-sizes (32); **b)** uses a single architectural design across all tasks. CORP [39] models are pre-trained for 800 epochs with a batch-size of 1024 using 64 NVIDIA-V100 GPUs and uses different variants (CORP_f and CORP_m) to obtain optimal performance on different datasets.

Kinetics-400: We present our results on Kinetics-400 [14] in Tab. 2 where our approach obtains state-of-the-art for both linear evaluation and fine-tuning settings. Performance on Kinetics-400 is heavily dependent on appearance attributes, *i.e.* a large proportion of its videos can be recognized by a single frame [92]. Strong performance of SVT on this dataset exhibits how well our proposed approach learns appearance related contextual information.

Table 1. **UCF-101 [69] & HMBD-51 [49]:** Top-1 (%) accuracy for both linear evaluation and fine-tuning. All models are pre-trained on Kinetics-400 [14] except ELo [63] which uses YouTube8M dataset [1]. Gray shaded methods use additional optical flow inputs. S-Res and T-Res represent spatial and temporal input resolution, respectively. Our approach shows state-of-the-art or on par performance.

Method	Backbone	TFLOPs	S-Res	T-Res	Epochs	UCF-101 [69]		HMDB-51 [49]	
						Linear	Fine-tune	Linear	Fine-tune
MemDPC [35] (ECCV '20)	R2D3D-34	-	224	64	-	54.1	86.1	30.5	54.5
CoCLR [36] (Neurips '20)	S3D	0.07	128	32	100	77.8	87.9	52.4	54.6
ELo [63] (CVPR '20)	R(2+1)D	17.5	224	-	100	-	84.2	-	53.7
RSPNet [15] (AAAI '21)	S3D-G	0.07	112	16	200	-	89.9	-	59.6
VideoMoCo [60](CVPR '21)	R(2+1)D	17.5	112	32	200	78.7	-	49.2	-
BE [82] (CVPR '21)	I3D	2.22	224	16	50	-	87.1	-	56.2
CMD [42](CVPR '21)	R(2+1)D-26	-	112	16	120	-	85.7	-	54.0
CVRL [64](CVPR '21)	R3D-50	3.19	224	32	800	89.2	92.2	57.3	66.7
TCLR [20] (Arxiv '21)	R(2+1)D-18	-	112	16	100	-	84.3	-	54.2
MoDist [87] (Arxiv '21)	R3D-50	3.19	224	8	100	91.5	94.0	63.0	67.4
BraVe [65](ICCV '21)	R3D-50	3.19	224	16	-	92.5	95.1	68.3	74.6
Vi^{2}CLR [23] (ICCV '21)	S3D	0.07	128	32	300	75.4	89.1	47.3	55.7
ASCNet [40] (ICCV '21)	S3D-G	0.07	224	64	200	-	90.8	-	60.5
TEC [44] (ICCV '21)	S3D-G	0.07	128	32	200	-	88.2	-	63.5
LSFD [8] (ICCV '21)	C3D	-	224	16	-	-	79.8	-	52.1
MCN [51] (ICCV '21)	R3D	3.19	128	32	50	73.1	89.7	42.9	59.3
CORP [39] (ICCV '21)	R3D-50	3.19	224	16	800	90.2	93.5	**58.7**	**68.0**
SVT (Ours)	ViT-B [9]	0.59	224	16	20	**90.8**	**93.7**	57.8	67.2

Table 2. **Kinetics-400 [14]:** Top-1 (%) accuracy is reported for both linear evaluation and fine-tuning on the Kinetics-400 validation set. All models are pre-trained on the training set of Kinetics-400 dataset. Our approach shows state-of-the-art performance.

Method	Backbone	Fine-tune	Linear
CVRL [64] (CVPR'21)	R3D-101	70.4	67.6
Vi^{2}CLR [23] (ICCV '21)	S3D	71.2	63.4
CORP [39] (ICCV '21)	R3D-50	-	66.6
SVT (Ours)	ViT-B [9]	**78.1**	**68.1**

Table 3. **SSv2 [33]:** Top-1 (%) for both linear evaluation and fine-tuning on the SSv2 validation set. All models are pre-trained on Kinetics-400. Our approach produces best results.

Method	Backbone	Fine-tune	Linear
MoDist [87] (Arxiv'21)	R3D-50	54.9	16.6
CORP [39] (ICCV '21)	R3D-50	48.8	-
SVT (Ours)	ViT-B [9]	**59.2**	**18.3**

SSv2: Similarly, we obtain state-of-the-art results on SSv2 dataset [33] for both linear evaluation and fine-tune settings as presented in Tab. 3. Multiple classes in SSv2 share similar backgrounds and object appearance, with complex movements differentiating them [39]. Performance on this dataset indicates how SVT feature representations capture strong motion related contextual cues as well.

4.3. Ablative Analysis

We systematically dissect the contribution of each component of our method. We study the effect of five individual elements: **a)** different combinations of local and global view correspondences; **b)** varying field of view along temporal vs spatial dimensions; **c)** temporal sampling strategy; **d)** spatial augmentations; **e)** slow-fast inference. In all our ablative experiments, SVT self-supervised training uses a subset of the Kinetics-400 train set containing 60K videos. Evaluation is carried out over *alternate* train-set splits of UCF-101 and HMDB-51. We train SVT for 20 epochs and evaluate using the same setup as described in Sec. 4.1.

View Correspondences. Learning correspondences between local and global views is the key motivation behind our proposed cross-view correspondences. Since multiple local-global view combinations can be considered for matching and prediction between views, we explore the effect of predicting each type of view from the other in Tab. 4. We observe that jointly predicting local to global and global to global view correspondences results in the optimal performance, while predicting global to local or local to local views leads to reduced performance. We believe this trend exists due to the emphasis on learning rich context in the case of joint prediction, which is absent for individual cases. Further, the performance drop for local to local correspondences (non-overlapping views) conforms with previous findings on the effectiveness of temporally closer positive views for contrastive self-supervised losses [31, 64].

Table 4. **View Correspondences.** Predicting local to global and global to global views remains optimal over any other combination.

$l \rightarrow g$	$g \rightarrow g$	$l \rightarrow l$	$g \rightarrow l$	UCF-101	HMDB-51
✓	✗	✗	✗	84.11	50.72
✗	✓	✗	✗	81.95	49.04
✓	✓	✗	✗	**84.64**	**52.17**
✓	✓	✓	✗	83.11	51.23
✓	✓	✗	✓	84.71	51.88
✓	✓	✓	✓	83.69	51.71

Table 5. **Spatial vs Temporal variations.** Cross-view correspondences with varying field of view along both spatial and temporal dimensions lead to optimal results. Temporal variations between views has more effect than applying only spatial variation.

Spatial	Temporal	UCF-101	HMDB-51
✓	✗	73.81	42.91
✗	✓	82.90	42.59
✓	✓	**84.64**	**52.17**

Table 6. **Temporal Sampling Strategy** . We compare our proposed temporal sampling strategy, motion correspondences (MC) (Sec. 3.2.1), against the alternate approach of temporal interval sampler (TIS) [64] used with CNNs under contrastive settings.

	UCF-101	HMDB-51
Ours + TIS [64]	82.24	50.10
Ours + MC	**84.64**	**52.17**

Table 7. **Augmentations**: Using temporally consistent augmentations (TCA) [64] applied randomly over the spatial dimensions for different views result in consistent improvements on UCF-101 and HMDB-51 datasets.

	UCF-101	HMDB-51
w/o TCA [64]	84.20	52.10
w TCA [64]	**84.64**	**52.17**

Table 8. **Slow-Fast Inference**: Feeding multiple views of varying spatiotemporal resolutions to a single shared network (multi-view) results in clear performance gains over feeding single-views across both UCF-101 and HMDB-51 datasets.

Slow-Fast	UCF-101	HMDB-51
✗	84.64	52.17
✓	**84.80**	**53.22**

Spatial vs Temporal Field of View. The optimal combination of spatiotemporal views in Tab. 4 involves varying the field of view (crops) along both spatial and temporal dimensions (Sec. 3.2.2). We study the effects of these variations (spatial or temporal) in Tab. 5. No variation along the spatial dimension denotes that all frames are of fixed spatial resolution 224×224 with no spatial cropping, and no temporal variation denotes that all frames in our views are sampled from a fixed time-axis region of a video. We observe that temporal variations have a significant effect on UCF-101, while variations in the field of view along both spatial and temporal dimension perform the best (Tab. 5).

Temporal Sampling Strategy. We study how our proposed temporal sampling strategy for motion correspondences (MC) could be replaced with alternate sampling approaches. To verify the effectiveness of MC, we replace it within SVT with an alternate approach. The temporal interval sampling (TIS) strategy in [64] obtains state-of-the-art performance under their contrastive video self-supervised setting with CNN backbones. Our experiments incorporating TIS in SVT (Tab. 6) highlight the advantage of our proposed MC sampling strategy over TIS.

Augmentations: We next explore standard spatial augmentations used on videos. Temporally consistent augmentations (TCA) proposed in [64] lead to improvements in their CNN based video self-supervision approach. We evaluate its effect on our approach in Tab. 7 which shows slight improvements. Given these performance gains, we adopt TCA in our SVT training process as well.

Slow-Fast Inference: Finally, we study the effect of our proposed Slow-Fast inference (Sec. 3.4) in Tab. 8. We observe higher gains on HMDB-51 [49], where the classes are easier to separate with motion information [36].

5. Conclusion

We present a video transformer based model trained using self-supervised objectives named SVT. Given an input video sequence, our approach first creates a set of spatiotemporally varying views sampled at different scales and frame rates from the same video. We then define two sets of correspondence learning tasks which seek to model the motion properties and cross-view relationships between the sampled clips. Specifically, our self-supervised objective reconstructs one view from the other in the latent space of student and teacher networks. Our approach is fast to train (converges within fewer iterations), does not require negative samples or large batch sizes, models long-range temporal dependencies, and performs dynamic slow-fast inference leading to better downstream performance. SVT is evaluated on four benchmark action recognition datasets where it performs well in comparison to existing state of the art.

Limitations: In this work, we explore SVT within the context of RGB input modality. Given large-scale multi-modal video datasets, the additional supervision available in the form of alternate modalities is not used by our current approach. In future work, we will explore how SVT can be modified to utilize multi-modal data sources.

Acknowledgements: This work was supported in part by the National Science Foundation (IIS-2104404 and CNS-2104416).

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Apostol (Paul) Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. In *ArXiv preprint*, 2016. 7

[2] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *ICCV*, 2015. 2

[3] Humam Alwassel, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2020. 2

[4] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. *ICCV*, 2021. 1, 2

[5] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020. 2

[6] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NeurIPS*, 2019. 2

[7] Miguel A. Bautista, Artsiom Sanakoyeu, Ekaterina Sutter, and Björn Ommer. Cliquecnn: Deep unsupervised exemplar learning. In *NeurIPS*, 2016. 2

[8] Nadine Behrmann, Mohsen Fayyaz, Juergen Gall, and Mehdi Noroozi. Long short view feature decomposition via contrastive video representation learning. In *ICCV*, 2021. 3, 7

[9] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, July 2021. 1, 2, 3, 6, 7, 12

[10] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2

[11] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018. 1, 2

[12] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020. 1, 2

[13] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 1, 2, 4, 5, 6, 12, 13

[14] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the Kinetics dataset. In *CVPR*, 2017. 2, 6, 7, 12

[15] Peihao Chen, Deng Huang, Dongliang He, Xiang Long, Runhao Zeng, Shilei Wen, Mingkui Tan, and Chuang Gan. Rspnet: Relative speed perception for unsupervised video representation learning. In *AAAI*, volume 1, 2021. 3, 7

[16] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 1, 2

[17] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *ArXiv preprint*, 2020. 2

[18] Xinlei Chen*, Saining Xie*, and Kaiming He. An empirical study of training self-supervised vision transformers. *ArXiv preprint*, 2021. 6

[19] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised visual transformers. *ArXiv preprint*, 2021. 2

[20] Ishan Rajendra Dave, Rohit Gupta, Mamshad Nayeem Rizve, and Mubarak Shah. TCLR: Temporal contrastive learning for video representation. *ArXiv preprint*, 2021. 3, 7

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv preprint*, 2018. 2, 4

[22] R Devon et al. Representation learning with video deep infomax. *ArXiv preprint*, 2020. 2

[23] Ali Diba, Vivek Sharma, Reza Safdari, Dariush Lotfi, Saquib Sarfraz, Rainer Stiefelhagen, and Luc Van Gool. Vi2clr: Video and image for visual contrastive learning of representation. In *ICCV*, 2021. 7

[24] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015. 2

[25] Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *ICCV*, 2017. 2

[26] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 1, 2, 4, 5

[27] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *NeurIPS*, 2014. 2

[28] Haoqi Fan, Yanghao Li, Bo Xiong, Wan-Yen Lo, and Christoph Feichtenhofer. Pyslowfast. https://github.com/facebookresearch/slowfast, 2020. 6

[29] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. *ICCV*, 2021. 1, 2

[30] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, pages 6202–6211, 2019. 2, 6

[31] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. *ArXiv preprint*, 2021. 2, 7

[32] Ross Goroshin, Joan Bruna, Jonathan Tompson, David Eigen, and Yann LeCun. Unsupervised learning of spatiotemporally coherent metrics. In *ICCV*, 2015. 2

[33] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzyńska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The "something something" video database for learning and evaluating visual common sense. In *ArXiv preprint*, 2017. 2, 6, 7, 12

[34] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020. 1, 2, 3, 4, 6

[35] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCV*, 2019. 2, 7

[36] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. *NeurIPS*, 2020. 2, 7, 8

[37] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 1, 2

[38] Olivier J Hénaff, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. In *ICML*, 2020. 2

[39] Kai Hu, Jie Shao, Yuan Liu, Bhiksha Raj, Marios Savvides, and Zhiqiang Shen. Contrast and order representations for video self-supervised learning. In *ICCV*, 2021. 6, 7

[40] Deng Huang, Wenhao Wu, Weiwen Hu, Xu Liu, Dongliang He, Zhihua Wu, Xiangmiao Wu, Mingkui Tan, and Errui Ding. Ascnet: Self-supervised video representation learning with appearance-speed consistency. In *ICCV*, 2021. 1, 3, 7

[41] Jiabo Huang, Qi Dong, Shaogang Gong, and Xiatian Zhu. Unsupervised deep learning by neighbourhood discovery. In *ICML*, 2019. 2

[42] Lianghua Huang, Yu Liu, Bin Wang, Pan Pan, Yinghui Xu, and Rong Jin. Self-supervised video representation learning by context and motion decoupling. *CVPR*, 2021. 7

[43] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H. Adelson. Learning visual groups from co-occurrences in space and time. 2016. 2

[44] Simon Jenni and Hailin Jin. Time-equivariant contrastive video representation learning. In *ICCV*, 2021. 1, 7

[45] Kumara Kahatapitiya and Michael S Ryoo. Coarse-fine networks for temporal activity detection in videos. In *CVPR*, 2021. 2, 6

[46] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ArXiv preprint*, 2021. 2

[47] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[48] Nikos Komodakis and Spyros Gidaris. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018. 2

[49] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. HMDB: A large video database for human motion recognition. In *ICCV*, 2011. 2, 6, 7, 8, 12

[50] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 2020. 2

[51] Yuanze Lin, Xun Guo, and Yan Lu. Self-supervised video representation learning with meta-contrastive network. In *ICCV*, 2021. 7

[52] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *ArXiv preprint*, 2021. 2

[53] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *ArXiv preprint*, 2021. 1

[54] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. In *ICLR*, 2016. 2

[55] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020. 2

[56] Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. Shuffle and learn: Unsupervised learning using temporal order verification. In *ECCV*, 2016. 2

[57] Muzammal Naseer, Kanchana Ranasinghe, Salman Khan, Munawar Hayat, Fahad Khan, and Ming-Hsuan Yang. Intriguing properties of vision transformers. In *NeurIPS*, 2021. 2, 5

[58] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016. 2

[59] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *NeurIPS*, 2018. 2

[60] Tian Pan, Yibing Song, Tianyu Yang, Wenhao Jiang, and Wei Liu. Videomoco: Contrastive video representation learning with temporally adversarial examples. In *CVPR*, 2021. 7

[61] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 2

[62] Viorica Pătrăucean, Ankur Handa, and Roberto Cipolla. Spatio-temporal video autoencoder with differentiable memory. In *ICLR (Workshop)*, 2016. 2

[63] AJ Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning. In *CVPR*, 2020. 1, 2, 7

[64] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. *CVPR*, 2021. 1, 2, 4, 5, 6, 7, 8

[65] Adrià Recasens, Pauline Luc, Jean-Baptiste Alayrac, Luyu Wang, Florian Strub, Corentin Tallec, Mateusz Malinowski, Viorica Patraucean, Florent Altché, Michal Valko, et al. Broaden your views for self-supervised video learning. *ICCV*, 2021. 1, 2, 3, 6, 7

[66] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 6

[67] Michael S. Ryoo, A. J. Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. Tokenlearner: What can 8

learned tokens do for images and videos? *ArXiv preprint*, 2021. 1, 2

[68] Gilad Sharir, Asaf Noy, and Lihi Zelnik-Manor. An image is worth 16x16 words, what is a video worth? *ArXiv preprint*, 2021. 1, 2

[69] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *ArXiv preprint*, 2012. 2, 6, 7, 12

[70] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *ICML*, 2015. 2

[71] Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *ArXiv preprint*, 2021. 2, 6

[72] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *ArXiv preprint*, 2017. 1, 4

[73] Kai Tian, Shuigeng Zhou, and Jihong Guan. Deepcluster: A general clustering framework based on deep learning. In *ECML/PKDD*, 2017. 2

[74] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *ECCV*, 2020. 2

[75] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. In *NeurIPS*, 2020. 2

[76] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers and distillation through attention. In *ICML*, 2021. 2

[77] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 2

[78] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *ICML*, 2008. 2

[79] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *NeurIPS*, 2016. 2

[80] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *ECCV*, 2018. 2

[81] Jacob Walker, Carl Doersch, Abhinav Gupta, and Martial Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *ECCV*, 2016. 2

[82] Jinpeng Wang, Yuting Gao, Ke Li, Yiqi Lin, Andy J Ma, Hao Cheng, Pai Peng, Rongrong Ji, and Xing Sun. Removing the background by adding the background: Towards background robust self-supervised video representation learning. In *CVPR*, 2021. 7

[83] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018. 2

[84] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *ICCV*. 2

[85] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. *ArXiv preprint*, 2020. 2

[86] Ross Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019. 6

[87] Fanyi Xiao, Joseph Tighe, and Davide Modolo. Modist: Motion distillation for self-supervised video representation learning. *ArXiv preprint*, 2021. 2, 4, 7

[88] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *ICML*, 2016. 1, 2

[89] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *CVPR*, 2022. 2

[90] Li Zhang, Dan Xu, Anurag Arnab, and Philip HS Torr. Dynamic graph message passing networks. In *CVPR*, 2020. 2

[91] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016. 2

[92] Yi Zhu, Xinyu Li, Chunhui Liu, Mohammadreza Zolfaghari, Yuanjun Xiong, Chongruo Wu, Zhi Zhang, Joseph Tighe, R Manmatha, and Mu Li. A comprehensive study of deep video action recognition. *ArXiv preprint*, 2020. 6, 12