

SVFormer: Semi-supervised Video Transformer for Action Recognition

Zhen Xing^{1,2} Qi Dai³ Han Hu³ Jingjing Chen^{1,2} Zuxuan Wu^{1,2†} Yu-Gang Jiang^{1,2}

¹ Shanghai Key Lab of Intell. Info. Processing, School of CS, Fudan University

² Shanghai Collaborative Innovation Center of Intelligent Visual Computing

³ Microsoft Research Asia

Abstract

Semi-supervised action recognition is a challenging but critical task due to the high cost of video annotations. Existing approaches mainly use convolutional neural networks, yet current revolutionary vision transformer models have been less explored. In this paper, we investigate the use of transformer models under the SSL setting for action recognition. To this end, we introduce SVFormer, which adopts a steady pseudo-labeling framework (i.e., EMA-Teacher) to cope with unlabeled video samples. While a wide range of data augmentations have been shown effective for semi-supervised image classification, they generally produce limited results for video recognition. We therefore introduce a novel augmentation strategy, Tube Token-Mix, tailored for video data where video clips are mixed via a mask with consistent masked tokens over the temporal axis. In addition, we propose a temporal warping augmentation to cover the complex temporal variation in videos, which stretches selected frames to various temporal durations in the clip. Extensive experiments on three datasets Kinetics-400, UCF-101, and HMDB-51 verify the advantage of SVFormer. In particular, SVFormer outperforms the state-of-the-art by 31.5% with fewer training epochs under the 1% labeling rate of Kinetics-400. Our method can hopefully serve as a strong benchmark and encourage future search on semi-supervised action recognition with Transformer networks. Code is released at <https://github.com/ChenHsing/SVFormer>.

1. Introduction

Videos have gradually replaced images and texts on Internet and grown at an exponential rate. On video websites such as YouTube, millions of new videos are uploaded every day. Supervised video understanding works [4, 15, 17, 29, 34, 56, 70] have achieved great successes. They rely on

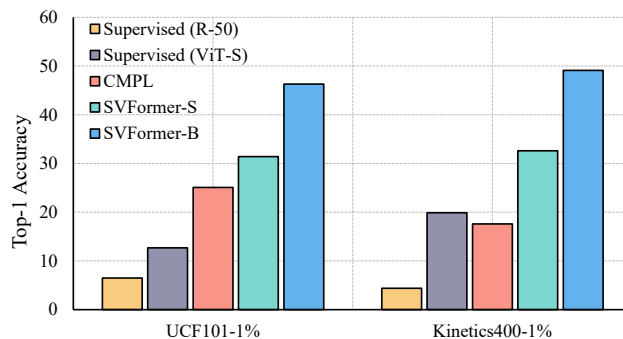


Figure 1. Comparison of our method with the supervised baseline and previous state-of-the-art SSL method [64]. SVFormer significantly outperforms previous methods under the case with very little labeled data.

large-scale manual annotations, yet labeling so many videos is time-consuming and labor-intensive. How to make use of unlabeled videos that are readily available for better video understanding is of great importance [25, 38, 39].

In this spirit, semi-supervised action recognition [25, 40, 57] explores how to enhance the performance of deep learning models using large-scale unlabeled data. This is generally done with labeled data to pretrain the networks [57, 64], and then leveraging the pretrained models to generate pseudo labels for unlabeled data, a process known as pseudo labeling. The obtained pseudo labels are further used to refine the pretrained models. In order to improve the quality of pseudo labeling, previous methods [57, 62] use additional modalities such as optical flow [3] and temporal gradient [50], or introduce auxiliary networks [64] to supervise unlabeled data. Though these methods present promising results, they typically require additional training or inference cost, preventing them from scaling up.

Recently, video transformers [2, 4, 34] have shown strong results compared to CNNs [15, 17, 22]. Though great success has been achieved, the exploration of transformers on semi-supervised video tasks remains unexplored. While it sounds appealing to extend vision transformers directly to SSL, a previous study shows that transformers perform significantly worse compared to CNNs in the low-data regime

[†] Corresponding author.

due to the lack of inductive bias [54]. As a result, directly applying SSL methods, *e.g.*, FixMatch [41], to ViT [13] leads to an inferior performance [54].

Surprisingly, in the video domain, we observe that TimeSformer, a popular video Transformer [4], initialized with weights from ImageNet [11], demonstrates promising results even when annotations are limited [37]. This encourages us to explore the great potential of transformers for action recognition in the SSL setting.

Existing SSL methods generally use image augmentations (*e.g.*, Mixup [67] and CutMix [66]) to speed up convergence under limited label resources. However, such pixel-level mixing strategies are not perfectly suitable for transformer architectures, which operate on tokens produced by patch splitting layers. In addition, strategies like Mixup and CutMix are particularly designed for image tasks, which fail to consider the temporal nature of video data. Therefore, as will be shown empirically, directly using Mixup or CutMix for semi-supervised action recognition leads to unsatisfactory performance.

In this work, we propose SVFormer, a transformer-based semi-supervised action recognition method. Concretely, SVFormer adopts a consistency loss that builds two differently augmented views and demands consistent predictions between them. Most importantly, we propose Tube TokenMix (TTMix), an augmentation method that is naturally suitable for video Transformer. Unlike Mixup and CutMix, Tube TokenMix combines features at the token-level after tokenization via a mask, where the mask has consistent masked tokens over the temporal axis. Such a design could better model the temporal correlations between tokens.

Temporal augmentations in literatures (*e.g.* varying frame rates) only consider simple temporal scaling or shifting, neglecting the complex temporal changes of each part in human action. To help the model learn strong temporal dynamics, we further introduce the Temporal Warping Augmentation (TWAug), which arbitrarily changes the temporal length of each frame in the clip. TWAug can cover the complex temporal variation in videos and is complementary to spatial augmentations [10]. When combining TWAug with TTMix, significant improvements are achieved.

As shown in Fig. 1, SVFormer achieves promising results in several benchmarks. (i) We observe that the supervised Transformer baseline is much better than the Conv-based method [22], and is even comparable with the 3D-ResNet state-of-the-art method [64] on Kinetics400 when trained with 1% of labels. (ii) SVFormer-S significantly outperforms previous state-of-the-arts with similar parameters and inference cost, measured by FLOPs. (iii) Our method is also effective for the larger SVFormer-B model. Our contributions are as follows:

- We are the first to explore the transformer model for semi-supervised video recognition. Unlike SSL for

image recognition with transformers, we find that using parameters pretrained on ImageNet is of great importance to ensure decent results for action recognition in the low-data regime.

- We propose a token-level augmentation Tube TokenMix, which is more suitable for video Transformer than pixel-level mixing strategies. Coupled with Temporal Warping Augmentation, which improves temporal variations between frames, TTMix achieves significant boost compared with image augmentation.
- We conduct extensive experiments on three benchmark datasets. The performances of our method in two different sizes (*i.e.*, SVFormer-B and SVFormer-S) outperform state-of-the-art approaches by clear margins. Our method sets a strong baseline for future transformer-based works.

2. Related Works

Deep Semi-supervised Learning Deep learning relies on large-scale annotated data, however collecting these annotations is labor-intensive. Semi-supervised learning is a natural solution to reduce the cost of labeling, which leverages a few labeled samples and a large amount of unlabeled samples to train the model. The research and application of SSL mainly focus on image recognition [41, 44, 54] with a two-step process: data augmentation and consistency regularization. Concretely, different data augmentations [10] views are input to the model, and their output consistencies are enforced through a consistency loss. Another line of work generates new data and labels using mixing [5, 19, 49] to train the network. Among these state-of-the-art methods, FixMatch have been widely used due its effective and its variants have been extended to many other applications, such as object detection [27, 32], semantic segmentation [1, 8], 3D reconstruction [61], etc. Although FixMatch has achieved good performance in many tasks, it may not achieve satisfactory results when directly transferred to video action recognition due to the lack of temporal augmentation. In this paper, we introduce temporal augmentation TWAug with mixing based method TTMix, which is suitable for video transformers at SSL settings.

Semi-supervised Action Recognition VideoSSL [25] presents a comparative study of applying 2D SSL methods to videos, which verifies the limitations of the direct extension of pseudo labeling method. TCL [40] explores the effect of a group contrastive loss and self-supervised tasks. MvPL [62] and LTG [57] introduce optical flow or temporal gradient modal to generate high quality pseudo labels for training, respectively. CMPL [64] introduce an auxiliary network, which requires more frames in training, increasing the difficulty of application. Besides, previous methods are

all based on 2D [40] or 3D convolutional [57, 62, 64] networks, which require more training epochs. Our approach is the first to make the exploration of Video Transformer for SSL action recognition and achieves the best performance with the least training cost.

Video Transformer The great success of vision transformer [13, 30, 33, 45, 68] in image recognition leads to the development of exploring the transformer-base architecture for video recognition tasks. VTN [36] uses additional temporal attention on the top of the pretrained ViT [13]. TimeFormer [4] investigates different spatial-temporal attention mechanisms and adopts factored space time attention as a trade-off of speed and accuracy. ViViT [2] explores four different types of attention, and selects the global spatio-temporal attention as the default to achieve promising performance. In addition, MviT [14], Video Swin [34], UniFormer [28] and Video Mobile-Former [53] incorporate the inductive bias in convolution into transformers. While these methods focus on fully-supervised setting, limited effort has been made for transformers in the semi-supervised setting.

Data Augmentation Data augmentation is an essential step in modern deep networks to improve the training efficiency and performance. Cutout [12] removes random rectangle regions in images. Mixup [67] performs image mixing by linearly interpolating both the raw image and labels. In CutMix [66], patches are cut and pasted among image pairs. AutoAugment [10] automatically searches for augmentation strategies to improve the results. PixMix [24] explores the natural structural complexity of images when performing mixing. TokenMix [31] mixes images at the token-level and allows the region to be multiple isolated parts. Though these methods have achieved good results, they are all specially designed for pure image and most of them are pixel-level augmentations. In contrast, our TTMix coupled with TWAug is devised for video.

3. Method

In this section, we first introduce the preliminaries of SSL in Sec. 3.1. The pipeline of our proposed SVFormer is described in Sec. 3.2. Then we detail the proposed Tube TokenMix (TTMix) in Sec. 3.3, as well as the effective albeit simple temporal warping augmentation. Finally, we show the training paradigm in Sec. 3.4.

3.1. Preliminaries of SSL

Suppose we have N training video samples, including N_L labeled videos $(x_l, y_l) \in \mathcal{D}_L$ and N_U unlabeled videos $x_u \in \mathcal{D}_U$, where x_l is the labeled video sample with a category label y_l , and x_u is the unlabeled video sample. In general, $N_U \gg N_L$. The aim of SSL is to utilize both \mathcal{D}_L and \mathcal{D}_U to train the model.

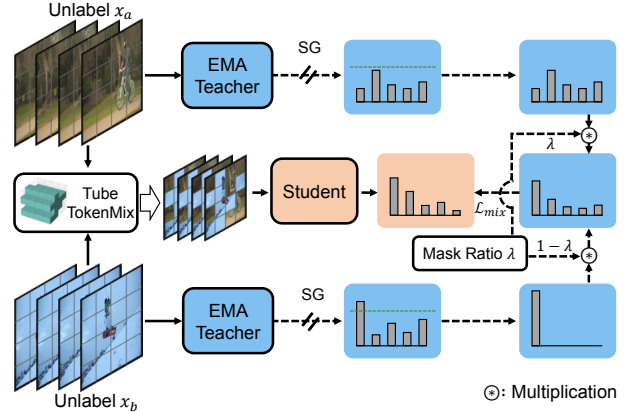


Figure 2. Overview of our Tube TokenMix training framework. The two input unlabeled samples are mixed via a tube mask, where the generated sample is fed into the Student model. The two samples are also fed into the EMA-Teacher to obtain their pseudo labels, which are further linearly interpolated via mask ratio λ to produce pseudo label for the generated sample. “SG” means stop gradient. The data augmentation is omitted here.

3.2. Pipeline

SVFormer follows the popular semi-supervised learning framework FixMatch [41] that use a consistency loss between two differently augmented views. The training paradigm is divided into two parts. For the labeled set $\{(x_l, y_l)\}_{l=1}^{N_L}$, the model optimizes the supervised loss \mathcal{L}_s :

$$\mathcal{L}_s = \frac{1}{N_L} \sum_{l=1}^{N_L} \mathcal{H}(\mathcal{F}(x_l), y_l), \quad (1)$$

where $\mathcal{F}(\cdot)$ refers to the predictions produced by the model and \mathcal{H} is the standard cross entropy loss.

For unlabeled samples x_u , we first use weak augmentations (e.g., random horizontal flipping, random scaling, and random cropping) and strong augmentations (e.g., AutoAugment [10] or Dropout [43]) to generate two views separately, $x_w = \mathcal{A}_{weak}(x_u)$, $x_s = \mathcal{A}_{strong}(x_u)$. Then the pseudo label of the weak view $\hat{y}_w = \arg \max(\mathcal{F}(x_w))$, which is produced by the model, is utilized to supervise the strong view, with the following unsupervised loss:

$$\mathcal{L}_{un} = \frac{1}{N_U} \sum_{u=1}^{N_U} \mathbb{I}(\max(\mathcal{F}(x_w)) > \delta) \mathcal{H}(\mathcal{F}(x_s), \hat{y}_w), \quad (2)$$

where δ is the predefined threshold, and \mathbb{I} is the indicator function that equals 1 when the maximum class probability exceeds δ otherwise 0. The confidence indicator is used to filter the noisy pseudo labels.

EMA-Teacher In FixMatch, the two augmented inputs share the same model, which tends to cause model collapsing easily [21, 23]. Therefore, we adopt the exponential

moving average (EMA)-Teacher in our framework, which is an improved version of FixMatch. The pseudo labels are generated by the EMA-Teacher model, whose parameters are updated by exponential moving average of the student parameters, formulated as:

$$\theta_t \leftarrow m\theta_t + (1 - m)\theta_s, \quad (3)$$

where m is a momentum coefficient, θ_t and θ_s are the parameters of teacher and student model, respectively. EMA has achieved success in many tasks, such as self-supervised learning [21, 23], SSL of image classification [19, 44], and object detection [27, 32]. Here we are the first to adopt this method in semi-supervised video action recognition.

3.3. Tube TokenMix

One of the core problems in semi-supervised frameworks is how to enrich the dataset with high-quality pseudo labels. Mixup [67] is a widely adopted data augmentation strategy, which performs convex combination between pairs of samples and labels as follows:

$$\hat{x} = \lambda \cdot x_1 + (1 - \lambda) \cdot x_2, \quad (4)$$

$$\hat{y} = \lambda \cdot y_1 + (1 - \lambda) \cdot y_2, \quad (5)$$

where the ratio λ is a scalar that conforms to the beta distribution. Mixup [67] and its variants (*e.g.* CutMix [66]) have achieved success in many tasks in the low-data regime, such as long-tail classification [9, 69], domain adaptation [63] [55], few-shot learning [35, 60], *etc.* For SSL, Mixup also performs well by mixing the pseudo labels of unlabeled samples in image classification [49].

Mixing in Videos While directly applying Mixup or CutMix to video scenarios results in clear improvements in Conv-based methods [71], these methods show unsatisfactory performance in our method. The reason is that our method adopts the Transformer backbone, where the pixel-level mixing augmentation (Mixup or CutMix) may be not perfectly suitable for such token-level models [31]. To narrow the gap, we propose 3 token-level mixing augmentation methods for video data, namely, Rand TokenMix, Frame TokenMix, and Tube TokenMix.

Fig. 2 illustrates the pipeline of our method. Given unlabeled video clips $x_a, x_b \in \mathbb{R}^{H \times W \times T}$, our method employ a token-level mask $\mathbf{M} \in \{0, 1\}^{H \times W \times T}$ to perform sample mixing. Note that H and W are the height and width of the frame after patch tokenization, and T is the clip length. To generate a new sample x_{mix} , we mix x_a and x_b after strong data augmentations \mathcal{A}_{strong} as follows:

$$x_{mix} = \mathcal{A}_{strong}(x_a) \odot \mathbf{M} + \mathcal{A}_{strong}(x_b) \odot (\mathbf{1} - \mathbf{M}), \quad (6)$$

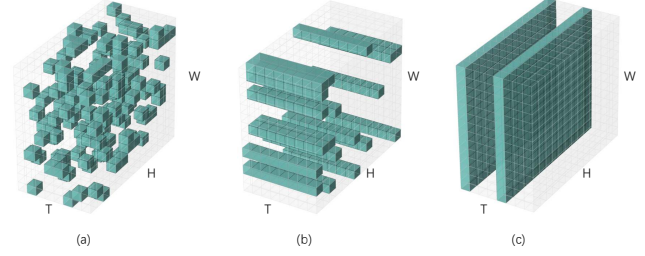


Figure 3. Examples of masks in three token mixing strategies. (a) Rand TokenMix. (b) Tube TokenMix. (c) Frame TokenMix.

Algorithm 1 Consistency loss for Tube TokenMix

Require: Unlabeled clip batch \mathbf{x}_a
Require: Tube TokenMask \mathbf{M} , mask ratio λ
Require: Teacher model \mathcal{F}_t
Require: Student model \mathcal{F}_s
Require: Confidence threshold δ

$\mathbf{x}_b = \text{shuffle}(\mathbf{x}_a)$ {shuffle samples in batch}
 $\hat{\mathbf{x}}_a = \text{spatial_aug}(\mathbf{x}_a)$ {spatial aug.}
 $\hat{\mathbf{x}}_b = \text{temporal_aug}(\mathbf{x}_b)$ {temporal warping aug.}
 $\hat{\mathbf{y}}_a = \text{stop_gradient}(\mathcal{F}_t(\mathbf{x}_a))$ {teacher pred.}
 $\hat{\mathbf{y}}_b = \text{stop_gradient}(\mathcal{F}_t(\mathbf{x}_b))$
 $\mathbf{c}_a = \max_i \mathbf{y}_a[i]$ {confidence of prediction}
 $\mathbf{c}_b = \max_i \mathbf{y}_b[i]$
 $\mathbf{x}_{mix} = \hat{\mathbf{x}}_a * \mathbf{M} + \hat{\mathbf{x}}_b * (1 - \mathbf{M})$ {mix clips}
 $\hat{\mathbf{y}}_{mix} = \hat{\mathbf{y}}_a * \lambda + \hat{\mathbf{y}}_b * (1 - \lambda)$ {mix tea. preds.}
 $\mathbf{c}_{mix} = \mathbf{c}_a * \lambda + \mathbf{c}_b * (1 - \lambda)$ {mix confidence.}
 $\mathbf{q} = \text{mean}(\mathbf{c}_{mix} \geq \delta)$ {mean of conf. mask}
 $\mathbf{y}_{mix} = \mathcal{F}_s(\mathbf{x}_{mix})$ {stu. pred. on mixed clip}
 $\mathcal{L}_{mix} = q \|\mathbf{y}_{mix} - \hat{\mathbf{y}}_{mix}\|_2^2$ {cons. loss}

Return \mathcal{L}_{mix}

where \odot is element-wise multiplication, and $\mathbf{1}$ is a binary mask with all ones.

The mask \mathbf{M} differs in the three augmentation methods, as demonstrated in Fig. 3. For Rand TokenMix, the masked tokens are randomly selected from the whole video clip (from $H \times W \times T$ tokens). For Frame TokenMix, we randomly select frames from the T frames and mask all the tokens in these frames. For Tube TokenMix, we adopt the tube-style masking strategy, that is, different frames share the same spatial mask matrix. In this case, the mask \mathbf{M} has consistent masked tokens over the temporal axis. While our mask design shares similarity with the recent masked image/video modeling [16, 47, 51, 52, 58, 59], our motivation is totally different. They focus on removing certain regions and making the model predict the masked areas for feature learning. In contrast, we leverage the mask to mix two clips and synthesize a new data sample.

The mixed sample x_{mix} is then fed to the student model \mathcal{F}_s , obtaining the model prediction $y_{mix} = \mathcal{F}_s(x_{mix})$. In addition, the pseudo labels \hat{y}_a, \hat{y}_b for x_a, x_b

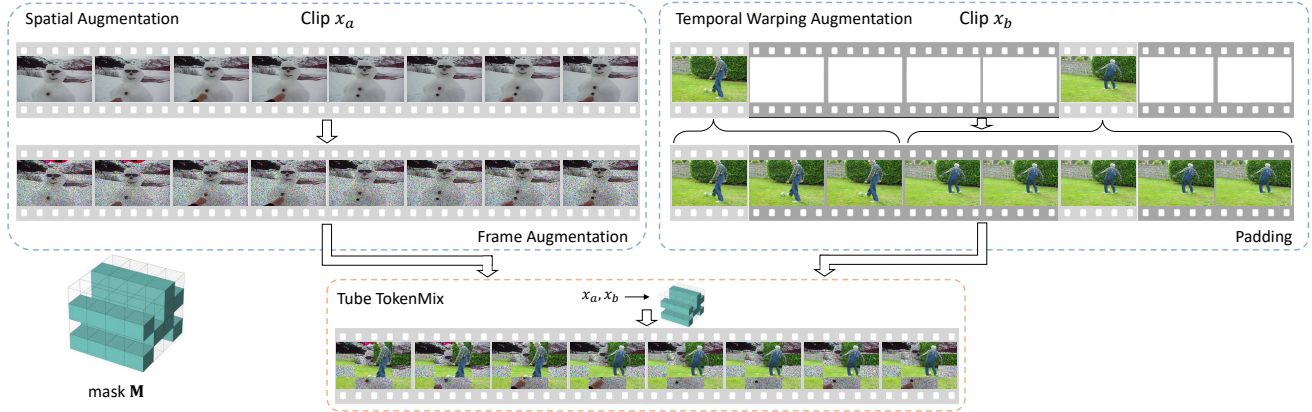


Figure 4. One example of utilizing Temporal Warping Augmentation together with spatial augmentation in TTMix. The two clips are first transformed by the two augmentations separately, after which TTMix is performed to generate a new clip.

are produced by inputting the weak augmented samples $\mathcal{A}_{weak}(x_a), \mathcal{A}_{weak}(x_b)$ to the teacher model \mathcal{F}_t :

$$\hat{y}_a = \arg \max(\mathcal{F}_t(\mathcal{A}_{weak}(x_a))), \quad (7)$$

$$\hat{y}_b = \arg \max(\mathcal{F}_t(\mathcal{A}_{weak}(x_b))). \quad (8)$$

Note that if $\max(\mathcal{F}_t(\mathcal{A}_{weak}(x))) < \delta$, the pseudo label \hat{y} remains the soft label $\mathcal{F}_t(\mathcal{A}_{weak}(x))$. The pseudo label \hat{y}_{mix} for x_{mix} is generated by mixing \hat{y}_a and \hat{y}_b with mask ratio λ :

$$\hat{y}_{mix} = \lambda \cdot \hat{y}_a + (1 - \lambda) \cdot \hat{y}_b. \quad (9)$$

Finally, the student model is optimized by the following consistency loss:

$$\mathcal{L}_{mix} = \frac{1}{N_m} \sum (\hat{y}_{mix} - y_{mix})^2, \quad (10)$$

where N_m is the number of mixed samples. The algorithm of consistency loss for TTMix is shown in Algorithm 1.

Temporal Warping Augmentation Most existing augmentation methods are designed for image tasks, which focus more on the spatial augmentation. They manipulate single or a pair of images to generate new image samples, without considering any temporal changes. Even the commonly adopted temporal augmentations, including varying temporal locations [18] and frame rates [40, 65], only consider simple temporal shift or scaling, that is, changing the holistic location or play speed. However, human actions are very complex and can have different temporal variation at every timestamp. To cover such challenging cases, we propose to distort the temporal duration of each frame, thus introducing higher randomness into the data.

Our Temporal Warping Augmentation (TWAug) can stretch one frame to various temporal length. Given an extracted video clip of T frames (*e.g.*, 8 frames), we randomly

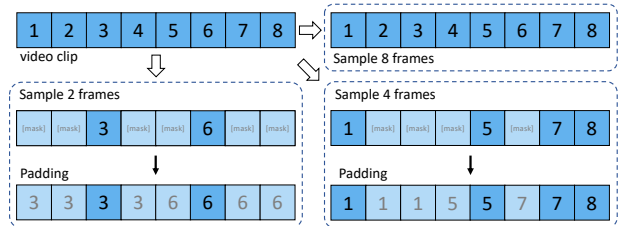


Figure 5. Illustration of Temporal Warping Augmentation. We demonstrate three augment examples of different selected frames.

determine to keep all the frames, or select a small portion of frames (*e.g.*, 2 or 4 frames) while masking the others. The masked frames are then padded with random neighbouring visible (unmasked) frames. Note that after temporal padding, the frame order is still retained. Fig. 5 shows three examples of selecting 2, 4, and 8 frames, respectively. The proposed TWAug can help the model learn the flexible temporal dynamics during training.

The Temporal Warping Augmentation serves as a strong augmentation in TTMix. Typically, we combine TWAug with the conventional spatial augmentation [10, 43] to perform the mixing. As shown in Fig. 4, the two input clips are first transformed by spatial augmentation and TWAug separately, after which the two clips are mixed through TTMix. We verify the effectiveness of our TWAug in Sec. 4.3.

3.4. Training Paradigm

The training of SVFormer consists of three parts: supervised loss formulated by Eq. (1), unsupervised pseudo-label consistency loss Eq. (2), and TTMix consistency loss Eq. (10). The final loss function is as follows:

$$\mathcal{L}_{all} = \mathcal{L}_s + \gamma_1 \mathcal{L}_{un} + \gamma_2 \mathcal{L}_{mix}, \quad (11)$$

where γ_1 and γ_2 are the hyperparameters for balancing the loss items.

Table 1. **Comparisons with state-of-the-art methods on UCF-101 and Kinetics-400.** Note that 3D-ResNet-18 and 3D-ResNet-50 denote the backbone networks and their depths. We report Top-1 accuracy as the evaluation metric. “Input” shows the modalities used during training, where “V” is the raw RGB video, “F” is optical flow and “G” is the temporal gradient.

Method	Backbone	Input	w ImgNet	Epoch	UCF-101		Kinetics-400	
					1%	10%	1%	10%
Supervised	3D-ResNet-50	V	✓	200	6.5	32.4	4.4	36.2
	ViT-S	V	✓	30	12.7	62.5	19.9	56.6
FixMatch (NeurIPS 2020) [41]	SlowFast-R50	V	✓	200	16.1	55.1	10.1	49.4
VideoSSL(WACV 2021) [25]	3D-ResNet-18	V	✓	-	-	42.0	-	33.8
TCL (CVPR 2021) [40]	TSM-ResNet-18	V		400	-	-	8.5	-
ActorCutMix (CVIU 2021) [71]	R(2+1)D-34	V	✓	600	-	53.0	9.02	-
MvPL (ICCV 2021) [62]	3D-ResNet-50	V+F+G		600	22.8	80.5	17.0	58.2
CMPL (CVPR 2022) [64]	R50 + R50-1/4	V	✓	200	25.1	79.1	17.6	58.4
LTG (CVPR 2022) [57]	3D-ResNet-18	V+G		180/360	-	62.4	9.8	43.8
TACL(TCSVT 2022) [46]	3D-ResNet-50	V	✓	200	-	55.6	-	-
L2A (ECCV 2022) [20]	3D-ResNet-18	V	✓	400	-	60.1	-	-
SVFormer-S (Ours)	ViT-S	V	✓	30	31.4	79.1	32.6	61.6
SVFormer-B (Ours)	ViT-B	V	✓	30	46.3	86.7	49.1	69.4

4. Experiment

In this section, we first introduce the experimental settings in Sec. 4.1. Following previous work [57, 64], we conduct experiments under different labeling rates in Sec 4.2. In addition, we also perform ablation experiments and empirical analysis in Section 4.3. If not emphasized, we only use RGB modal for inference with the official validation set.

4.1. Experiment Settings

Datasets Kinetics-400 [7] is a large-scale human action video dataset, with up to 245k training samples and 20k validation samples, covering 400 different categories. We follow the state-of-the-art methods MvPL [62] and CMPL [64] to sample 6 or 60 labeled training videos per category, *i.e.* at 1% or 10% labeling rates. UCF-101 [42] is a dataset with 13,320 video samples, which consists of 101 categories. We also sample 1 or 10 samples in each category as the labeled set following CMPL [64]. As for HMDB-51 [26], it is a small-scale dataset with only 51 categories composed of 6,766 videos. Following the division of LTG [57] and VideoSSL [25], we conduct experiments at three different labeling rates: 40%, 50%, and 60%.

Evaluation Metric We show the accuracy of Top-1 in main results, and also present the accuracy of Top-5 in some ablation experiments.

Baseline We utilize the ViT [13] extended video TimeSformer [4] as the backbone of our baseline. The hyperparameters are mostly kept the same as the baseline, and we adopt the divided space-time attention as in TimeSformer [4]. Since TimeSformer only have ViT-Base models, we implement SVFormer-Small model from DeiT-S [48] with

Table 2. **Comparisons with state-of-the-art methods on HMDB-51.** We report Top-1 accuracy. “Input” shows the modalities used during training, where “V” is the raw RGB video, “F” is optical flow and “G” is the temporal gradient.

	Backbone	Input	40%	50%	60%
VideoSSL [25]	3D-R18	V	32.7	36.2	37.0
ActorCutMix [71]	R(2+1)D-34	V	32.9	38.2	38.9
MvPL [62]	3D-R18	V+F+G	30.5	33.9	35.8
LTG [57]	3D-R18	V+G	46.5	48.4	49.7
TACL [46]	3D-R18	V	38.7	40.2	41.7
L2A [20]	3D-R18	V	42.1	46.3	47.1
SVFormer-S (Ours)	ViT-S	V	56.2	58.2	59.7
SVFormer-B (Ours)	ViT-B	V	61.6	64.4	68.2

the dimension of 384 and 6 heads, in order to have comparable number of parameters with other Conv-based methods [22, 62, 64]. For fair comparisons, we train 30 epochs for TimeSformer as the supervised baseline.

Training and Inference For training, we follow the setting of TimeSformer [4]. The training uses 8 or 16 GPUs, with a SGD optimizer using a momentum of 0.9 and a weight decay of 0.001. For each setting, the basic learning rate is set to 0.005, which is divided by 10 at epochs 25, and 28. As for the confidence score threshold, we search for the optimal δ from {0.3, 0.5, 0.7, 0.9}. γ_1 and γ_2 are set to 2. The masking ratio λ is sampled from beta distribution $\text{Beta}(\alpha, \alpha)$, where $\alpha = 10$. In the testing phase, following the inference strategies in MvPL [62] and CMPL [64], we uniformly sample five clips from the entire video, and make three different crops to get 224×224 resolution to cover most of the spatial areas of the clips. The final prediction is the average of the softmax probabilities of these 5×3 pre-

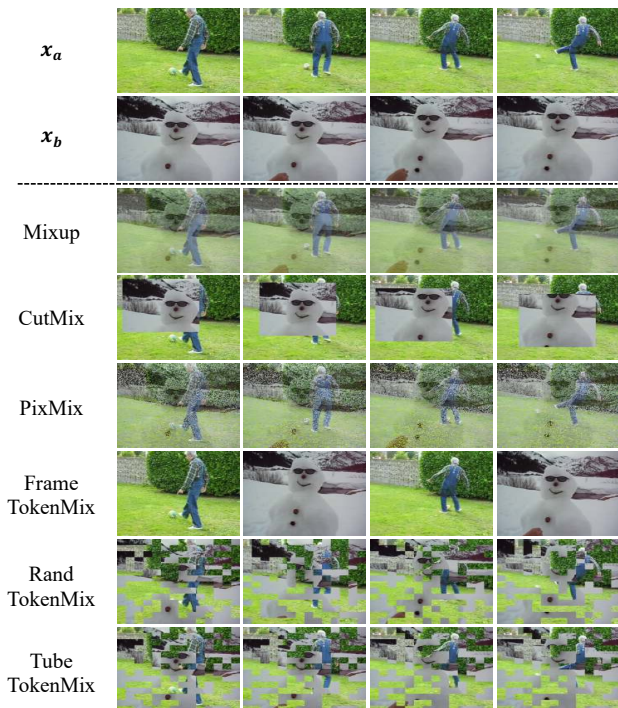


Figure 6. Example of the traditional pixel-level mixing methods and our proposed token-level mixing. Note that token-level methods mix two samples after tokenization but the visualization is shown on image-level for clear presentation.

dictions. We also conduct a comparison of inference setting in the ablation study in Sec. 4.3.

4.2. Main Results

The main results of Kinetics-400 [7] and UCF-101 [42] are shown in Table 1. Compared with previous methods, our model SVFormer-S achieves the best performance with the fewest training epochs among the methods that only use RGB data. In particular, at the labeling rate of 1% setting, SVFormer-S improves previous approach [64] by 6.3% in UCF-101 and 15.0% in Kinetics-400. In addition, when adopting larger models, SVFormer-B significantly outperforms the state-of-the-art methods.

Specifically, in Kinetics-400, SVFormer-B can achieve 69.4% with only 10% labeled data, which is comparable to 77.9% of fully-supervised setting in TimeSformer [4]. Moreover, as shown in Table 2, for the small-scale dataset HMDB-51 [26], our SVFormer-S and SVFormer-B have also improved by about 10% and 15% compared with the previous method [57].

4.3. Ablation Studies

To understand the effect of each part of the design in our method, we conduct extensive ablation studies on the Kinetics-400 and UCF-101 at the 1% labeling ratio setting

Table 3. **SSL framework selection.** We compare the EMA-Teacher framework with FixMatch. The results are reported on Kinetics-400 and UCF-101 with 1% labeling ratio.

Method	UCF-1%		Kinetic-1%	
	Top-1	Top-5	Top-1	Top-5
Baseline	12.7	29.8	19.9	42.3
FixMatch [41]	25.1	47.3	28.2	54.6
EMA-Teacher	31.4	56.9	32.6	59.0

Table 4. **TokenMix Mask sampling.** We compare different token masking strategies. The results are reported on Kinetics-400 and UCF-101 with 1% labeling ratio.

Method/Dataset	UCF-1%		Kinetic-1%	
	Top-1	Top-5	Top-1	Top-5
Baseline	26.1	48.9	23.6	47.7
CutMix [66]	28.7	51.3	28.6	53.7
Mixup [67]	29.8	53.0	29.3	55.1
PixMix [24]	29.7	52.4	29.6	55.8
Frame TokenMix	29.8	54.2	26.3	50.0
Rand TokenMix	30.3	55.3	28.8	54.2
Tube TokenMix	31.4	56.9	32.6	59.0

with SVFormer-S.

Analysis of SSL framework The comparison of FixMatch and EMA-Teacher is shown in Table 3. It is clear that the two methods have significantly improved the baseline approach. In addition, EMA-Teacher has exhibited considerable gains over FixMatch in both datasets with very few labeled samples, probably because it has improved the stability of training. FixMatch [41] may lead to model collapse with limited labels as shown in [6].

Analysis of different mixing strategies We now compare the Tube TokenMix strategy with three pixel-level mixing methods, CutMix [66], Mixup [67], PixMix [24], as well as the other two token-level mixing methods, *i.e.*, Frame TokenMix and Rand TokenMix. The examples of different mixing methods are shown in Fig. 6. The quantitative results are shown in Table 4. Compared with these alternative methods, all mixing methods can improve the performance, which proves the effectiveness of mixing-based consistency losses. In addition, we observe that the token-level methods (Rand TokenMix and Tube TokenMix) perform better than the pixel-level mixing methods. This is not surprising since transformers operate on tokens, and thus token-level mixing has inherent advantages.

The performance of Frame TokenMix is even worse than that of pixel-level mixing methods, which is also expected. We hypothesize that replacing entire frames in video clip will scramble up the temporal reasoning, thus leading to poor temporal attention modeling. In addition, Tube Token-

Table 5. **Effects of Spatial and Temporal Warping Augmentations (TWAug).** The results are reported on Kinetics-400 and UCF-101 with 1% labeling ratio.

	Spatial	Temporal	UCF-1%		Kinetic-1%	
			Top-1	Top-5	Top-1	Top-5
Baseline			29.5	52.1	28.5	54.2
Spatial-only	✓		29.9	53.2	30.3	55.9
Temporal-only		✓	30.2	55.8	30.8	56.4
Spatial+Temporal	✓	✓	31.4	56.9	32.6	59.0

Table 6. **Effects of different inference schemes.** We compare the sparse sampling strategy and dense sampling method with different frames. The $a \times b$ in ‘Frames’ column means sampling a frames at frame rate b . The results are reported on Kinetics-400 and UCF-101 with 1% labeling ratio.

Method	Frames	Test View	UCF-1%	Kinetic-1%
MvPL [62]	8×8	10×3	22.8	17.0
CMPL [64]	8×8	10×3	25.1	17.6
SVFormer-S	8×32	1×3	29.3	31.0
SVFormer-S	8×8	5×3	31.4	32.6
SVFormer-S	16×4	10×3	31.6	33.1

Mix achieves the best results. We suppose the consistent masked tokens over temporal axis can prevent information leaky between adjacent frames in the same spatial locations, especially in such a short-term clip. Therefore, Tube Token-Mix could better model the spatio-temporal correlations.

Analysis of Augmentations The effects of spatial augmentation and temporal warping augmentation are evaluated in Table 5. The baseline indicates removing both strong spatial augmentation (e.g. AutoAugment [10] and Dropout [43]) and TWAug in all branches. In this case, the experimental performance drops dramatically. When spatial augmentation or temporal warping augmentation is incorporated into the baseline separately, the performance is improved. The best practice is to perform data augmentations in both spatial and temporal.

Analysis of Inference We evaluate the effect of frame rate sampling and different inference schemes, as shown in Table 6. Previous methods, i.e. CMPL [64] and MvPL [62], utilize the clip-based sparse sampling, which samples 8 frames at frame rate 8 as a clip. For each video, 10 clips are sampled, where each clip is cropped 3 times according to different spatial positions. Finally, the predictions of 10×3 samples are averaged. TimeSformer [4] adopts the video-based sparse sampling, that is, 8 frames are sampled at frame rate 32 as the representation of the whole video. Then 3 different cropped views are used, namely 8×32 . Applying the video-based sparse sampling scheme as in TimeSformer [4] can reduce the inference cost, but the performance is worse than that of clip-based sampling. Experiments at sampling schemes of 8×8 and 16×4 demonstrate better performance. For the trade-off between effi-

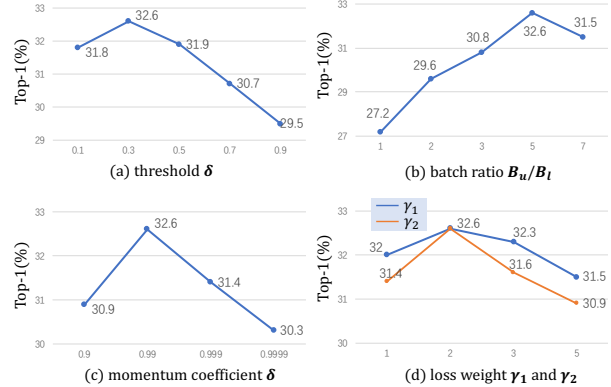


Figure 7. **Effects of hyperparameters.** Results of varying threshold δ , ratio between unlabeled and labeled data in a mini-batch (B_u/B_l), momentum coefficient m , and loss weights (γ_1, γ_2) are included to comprehensively study the effects of the hyperparameters. Reported on Kinetics-400 with 1% labeling ratio.

ciency and accuracy, we use 8×8 as the default setting following CMPL [64] and MvPL [62].

Analysis of hyperparameters Here we explore the effect of different hyperparameters. We conduct experiments under 1% setting of Kinetics-400. We first explore the effect of different threshold values of δ . As shown in Fig. 7(a), we can observe that when labeled samples are extremely scarce, best results are achieved by setting a small δ value ($\delta = 0.3$). We then evaluate how the ratio between labeled samples and unlabeled samples in a mini-batch affect the result. We fix the labeled sample number B_l to 1, and sample B_u unlabeled samples to form a mini-batch, where B_u is in $\{1, 2, 3, 5, 7\}$. The results are shown in Fig. 7(b). When $B_u = 5$, the model produces the highest result. Finally, we explore the choice of momentum coefficient m of EMA and the loss weights γ_1 and γ_2 , as shown in Fig. 7(c) and Fig. 7(d). We thus set $m = 0.99$ and $\gamma_1 = \gamma_2 = 2$ as default setting in all the experiments.

5. Conclusion

In this paper we present SVFormer, a transformer-based semi-supervised video action recognition method. We propose Tube TokenMix, a data augmentation method that is specially designed for video transformer models. Coupled with the temporal warping augmentation, which covers the complex temporal variations by arbitrarily changing the frame length, TTMix achieves significant improvement compared with conventional augmentations. SVFormer outperforms the state-of-the-art with a large margin on UCF-101, HMDB-51 and Kinetics-400 without increasing overheads. Our work establishes a new benchmark for semi-supervised action recognition and encourages future work to adopt Transformer architecture.

Acknowledgement This project was supported by NSFC under Grant No. 62032006 and No. 62102092.

References

- [1] Inigo Alonso, Alberto Sabater, David Ferstl, Luis Monteseano, and Ana C Murillo. Semi-supervised semantic segmentation with pixel-level contrastive learning from a class-wise memory bank. In *ICCV*, 2021. 2
- [2] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *ICCV*, 2021. 1, 3
- [3] Steven S. Beauchemin and John L. Barron. The computation of optical flow. *ACM computing surveys (CSUR)*, 1995. 1
- [4] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, 2021. 1, 2, 3, 6, 7, 8
- [5] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, 2019. 2
- [6] Zhaowei Cai, Avinash Ravichandran, Paolo Favaro, Manchen Wang, Davide Modolo, Rahul Bhotika, Zhuowen Tu, and Stefano Soatto. Semi-supervised vision transformers at scale. In *NeurIPS*, 2022. 7
- [7] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 6, 7
- [8] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *CVPR*, 2021. 2
- [9] Hsin-Ping Chou, Shih-Chieh Chang, Jia-Yu Pan, Wei Wei, and Da-Cheng Juan. Remix: rebalanced mixup. In *ECCV*, 2020. 4
- [10] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *CVPR*, 2019. 2, 3, 5, 8
- [11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 2
- [12] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 3
- [13] 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. In *ICLR*, 2021. 2, 3, 6
- [14] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *ICCV*, 2021. 3
- [15] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *CVPR*, 2020. 1
- [16] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. In *NeurIPS*, 2022. 4
- [17] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019. 1
- [18] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *CVPR*, 2021. 5
- [19] Geoff French, Avital Oliver, and Tim Salimans. Milking cowmask for semi-supervised image classification. *arXiv preprint arXiv:2003.12022*, 2020. 2, 4
- [20] Shreyank N Gowda, Marcus Rohrbach, Frank Keller, and Laura Sevilla-Lara. Learn2augment: Learning to composite videos for data augmentation in action recognition. In *ECCV*, 2022. 6
- [21] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent—a new approach to self-supervised learning. *NeurIPS*, 2020. 4
- [22] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Learning spatio-temporal features with 3d residual networks for action recognition. In *ICCVW*, 2017. 1, 2, 6
- [23] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 4
- [24] Dan Hendrycks, Andy Zou, Mantas Mazeika, Leonard Tang, Bo Li, Dawn Song, and Jacob Steinhardt. Pixmix: Dream-like pictures comprehensively improve safety measures. In *CVPR*, 2022. 3, 7
- [25] Longlong Jing, Toufiq Parag, Zhe Wu, Yingli Tian, and Hongcheng Wang. Videoss!: Semi-supervised learning for video classification. In *WACV*, 2021. 1, 2, 6
- [26] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, 2011. 6, 7
- [27] Hengduo Li, Zuxuan Wu, Abhinav Shrivastava, and Larry S Davis. Rethinking pseudo labels for semi-supervised object detection. In *AAAI*, 2022. 2, 4
- [28] Kunchang Li, Yali Wang, Peng Gao, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. Uniformer: Unified transformer for efficient spatiotemporal representation learning. In *ICLR*, 2022. 3
- [29] Wenxu Li, Gang Pan, Chen Wang, Zhen Xing, and Zhenjun Han. From coarse to fine: Hierarchical structure-aware video summarization. *ACM TOMM*, 2022. 1
- [30] Zhixin Ling, Zhen Xing, Xiangdong Zhou, Manliang Cao, and Guichun Zhou. Panoswin: a pano-style swin transformer for panorama understanding. In *CVPR*, 2023. 3
- [31] Jihao Liu, Boxiao Liu, Hang Zhou, Hongsheng Li, and Yu Liu. Tokenmix: Rethinking image mixing for data augmentation in vision transformers. In *ECCV*, 2022. 3, 4
- [32] Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Peizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. In *ICLR*, 2021. 2, 4
- [33] 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. In *ICCV*, 2021. 3
- [34] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *CVPR*, 2022. 1, 3

- [35] Puneet Mangla, Nupur Kumari, Abhishek Sinha, Mayank Singh, Balaji Krishnamurthy, and Vineeth N Balasubramanian. Charting the right manifold: Manifold mixup for few-shot learning. In *WACV*, 2020. 4
- [36] Daniel Neimark, Omri Bar, Maya Zohar, and Dotan Asselmann. Video transformer network. In *ICCVW*, 2021. 3
- [37] Farrukh Rahman, Ömer Mubarek, and Zsolt Kira. On the surprising effectiveness of transformers in low-labeled video recognition. In *NeurIPS*, 2022. 2
- [38] Baifeng Shi, Qi Dai, Judy Hoffman, Kate Saenko, Trevor Darrell, and Huijuan Xu. Temporal action detection with multi-level supervision. In *ICCV*, 2021. 1
- [39] Baifeng Shi, Qi Dai, Yadong Mu, and Jingdong Wang. Weakly-supervised action localization by generative attention modeling. In *CVPR*, 2020. 1
- [40] Ankit Singh, Omprakash Chakraborty, Ashutosh Varshney, Rameswar Panda, Rogerio Feris, Kate Saenko, and Abir Das. Semi-supervised action recognition with temporal contrastive learning. In *CVPR*, 2021. 1, 2, 3, 5, 6
- [41] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, 2020. 2, 3, 6, 7
- [42] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 6, 7
- [43] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 2014. 3, 5, 8
- [44] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, volume 30, 2017. 2, 4
- [45] Rui Tian, Zuxuan Wu, Qi Dai, Han Hu, Yu Qiao, and Yu-Gang Jiang. Resformer: Scaling vits with multi-resolution training. In *CVPR*, 2023. 3
- [46] Anyang Tong, Chao Tang, and Wenjian Wang. Semi-supervised action recognition from temporal augmentation using curriculum learning. *IEEE TCSVT*, 2022. 6
- [47] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *NeurIPS*, 2022. 4
- [48] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 6
- [49] Vikas Verma, Kenji Kawaguchi, Alex Lamb, Juho Kannala, Yoshua Bengio, and David Lopez-Paz. Interpolation consistency training for semi-supervised learning. In *IJCAI*, 2019. 2, 4
- [50] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 1
- [51] Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Yu-Gang Jiang, Luowei Zhou, and Lu Yuan. Bevt: Bert pretraining of video transformers. In *CVPR*, 2022. 4
- [52] Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Lu Yuan, and Yu-Gang Jiang. Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In *CVPR*, 2023. 4
- [53] Rui Wang, Zuxuan Wu, Dongdong Chen, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Luowei Zhou, Lu Yuan, and Yu-Gang Jiang. Video mobile-former: Video recognition with efficient global spatial-temporal modeling. *arXiv preprint arXiv:2208.12257*, 2022. 3
- [54] Zejia Weng, Xitong Yang, Ang Li, Zuxuan Wu, and Yu-Gang Jiang. Semi-supervised vision transformers. In *ECCV*, 2022. 2
- [55] Yuan Wu, Diana Inkpen, and Ahmed El-Roby. Dual mixup regularized learning for adversarial domain adaptation. In *ECCV*, 2020. 4
- [56] Zuxuan Wu, Hengduo Li, Yingbin Zheng, Caiming Xiong, Yu-Gang Jiang, and Davis Larry S. A coarse-to-fine framework for resource efficient video recognition. *IJCV*, 2021. 1
- [57] Junfei Xiao, Longlong Jing, Lin Zhang, Ju He, Qi She, Zongwei Zhou, Alan Yuille, and Yingwei Li. Learning from temporal gradient for semi-supervised action recognition. In *CVPR*, 2022. 1, 2, 3, 6, 7
- [58] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *CVPR*, 2022. 4
- [59] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Yixuan Wei, Qi Dai, and Han Hu. On data scaling in masked image modeling. In *CVPR*, 2023. 4
- [60] Zhen Xing, Yijiang Chen, Zhixin Ling, Xiangdong Zhou, and Yu Xiang. Few-shot single-view 3d reconstruction with memory prior contrastive network. In *ECCV*, 2022. 4
- [61] Zhen Xing, Hengduo Li, Zuxuan Wu, and Yu-Gang Jiang. Semi-supervised single-view 3d reconstruction via prototype shape priors. In *ECCV*, 2022. 2
- [62] Bo Xiong, Haoqi Fan, Kristen Grauman, and Christoph Feichtenhofer. Multiview pseudo-labeling for semi-supervised learning from video. In *ICCV*, 2021. 1, 2, 3, 6, 8
- [63] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. In *AAAI*, 2020. 4
- [64] Yinghao Xu, Fangyun Wei, Xiao Sun, Ceyuan Yang, Yujun Shen, Bo Dai, Bolei Zhou, and Stephen Lin. Cross-model pseudo-labeling for semi-supervised action recognition. In *CVPR*, 2022. 1, 2, 3, 6, 7, 8
- [65] Ceyuan Yang, Yinghao Xu, Bo Dai, and Bolei Zhou. Video representation learning with visual tempo consistency. *arXiv preprint arXiv:2006.15489*, 2020. 5
- [66] Sangdoon Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019. 2, 3, 4, 7

- [67] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. [2](#), [3](#), [4](#), [7](#)
- [68] Xiaosong Zhang, Yunjie Tian, Lingxi Xie, Wei Huang, Qi Dai, Qixiang Ye, and Qi Tian. Hivit: A simpler and more efficient design of hierarchical vision transformer. In *ICLR*, 2023. [3](#)
- [69] Xing Zhang, Zuxuan Wu, Zejia Weng, Huazhu Fu, Jingjing Chen, Yu-Gang Jiang, and Larry S Davis. Videolt: Large-scale long-tailed video recognition. In *ICCV*, 2021. [4](#)
- [70] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020. [1](#)
- [71] Yuliang Zou, Jinwoo Choi, Qitong Wang, and Jia-Bin Huang. Learning representational invariances for data-efficient action recognition. *arXiv preprint arXiv:2103.16565*, 2021. [4](#), [6](#)