

# Few-shot Action Recognition with Hierarchical Matching and Contrastive Learning

Sipeng Zheng<sup>1</sup>, Shizhe Chen<sup>2</sup>, Qin Jin<sup>1\*</sup>

<sup>1</sup>Renmin University of China <sup>2</sup>Inria

{zhengsipeng@, qjin@}.ruc.edu.cn shizhe.chen@inria.fr

**Abstract.** Few-shot action recognition aims to recognize actions in test videos based on limited annotated data of target action classes. The dominant approaches project videos into a metric space and classify videos via nearest neighboring. They mainly measure video similarities using global or temporal alignment alone, while an optimum matching should be multi-level. However, the complexity of learning coarse-to-fine matching quickly rises as we focus on finer-grained visual cues, and the lack of detailed local supervision is another challenge. In this work, we propose a hierarchical matching model to support comprehensive similarity measure at global, temporal and spatial levels via a zoom-in matching module. We further propose a mixed-supervised hierarchical contrastive learning (HCL), which not only employs supervised contrastive learning to differentiate videos at different levels, but also utilizes cycle consistency as weak supervision to align discriminative temporal clips or spatial patches. Our model achieves state-of-the-art performance on four benchmarks especially under the most challenging 1-shot recognition setting.

**Keywords:** few-shot learning, action recognition, contrastive learning

## 1 Introduction

Large-scale video datasets [5, 13] have greatly accelerated the research on action recognition using deep neural networks [31], which however, is data-hungry and hard to generalize well on new classes with limited training examples. Therefore, few-shot action recognition (FSAR) [48, 3] has attracted more and more attention. One of the mainstream approaches for FSAR is the metric-based method [32, 41]. The key idea is to learn a generalizable metric from action classes with abundant labeled videos, and such metric can be used to measure the similarity between any videos. In this way, we recognize the few-shot classes by computing the similarity between the query video and the few labeled videos.

There is no doubt that an ideal metric should be learned from multi-levels for accurate video matching *e.g.*, at instance, clip, frame or even patch level. However, the matching complexity drastically rises from coarse- to fine-grained, and combining the alignment of these different granularities is quiet challenging. For example, global matching refers only to the similarity of a pair of features,

---

\* Qin Jin is the corresponding author.

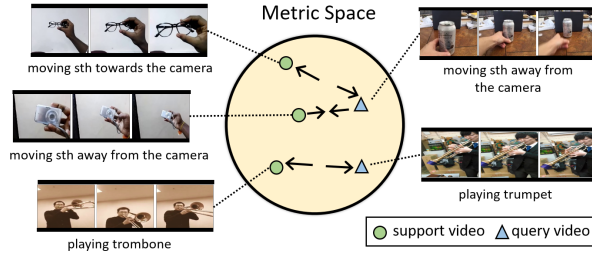


Fig. 1: Metric-based approaches for FSAR learn a metric space to measure video similarities. In addition to global cues, the video representation in the space should hierarchically capture temporal dynamics and discriminative spatial regions to correctly match query videos with support videos.

while patch matching may need to deal with a large number of patch-to-patch alignment. Subject to this limitation, existing metric-based FSAR works simply compare two videos from a single granularity, mainly including global or temporal matching. The global matching approaches [20, 11] encode a video as a fixed-size vector to compute similarities, which fail to differentiate different temporal dynamics such as “moving towards” or “moving away” as shown in Fig. 1. The temporal matching approaches instead leverage temporal alignment between frames [4] or clips [29]. Despite great progress, these works suffer from condensed spatial information. For example, actions “playing trombone” and “playing trumpet” in Fig. 1 have similar temporal movements. One needs to focus on discriminative spatial regions of the instrument in order to classify them correctly. Therefore, a mechanism to reliably and efficiently capture various alignment in videos is necessary.

Another challenge to learn both coarse- and fine-grained alignment simultaneously lies in the learning approach on few-shot sets. Earlier FSAR methods [41, 20, 3] employ cross entropy loss to train on global features, which are prone to overfit and do not generalize well for few-shot classes. More recent works [29] adopt supervised contrastive loss [17] in episodic training, where a limited number of action classes are used per training iteration. Such episodic training mimics standard  $N$ -way  $K$ -shot setting [32], but it cannot take full advantage of contrastive learning, which usually requires diverse and large number of negative examples [14, 45] to learn good representations. In addition, the supervisions are only available at the video level, and it is expensive to manually annotate temporal or spatial alignment between videos to train fine-grained matching. Therefore the training becomes quite challenging without detailed annotations.

In this paper, we tackle the above challenges by comparing any two videos based on: global video representations, temporally aligned clip representations to capture temporal orders, and spatially aligned patch representations to encode detailed spatial information. To be specific, we firstly propose a hierarchical matching model to more comprehensively and efficiently measure video similari-

ties. Our proposed model matches videos progressively from coarse-level to fine-grained level, using features of coarse level to focus on local information at finer-grained level *e.g.*, from clip to patch. Such matching mechanism, called zoom-in matching module, alleviates the complexity of hierarchical matching to better scale up when aligning fine-grained visual cues like clips or patches. Secondly, we develop a hierarchical contrastive learning (HCL) algorithm for coarse-to-fine video representation learning. Specifically, we develop a mixed-supervised contrastive learning to avoid the limitations of previous episodic training paradigm and thus learn more discriminatively. In addition to supervised contrastive learning, we use cycle consistency to build temporal and spatial associations between videos of the same action class. It enables contrastive learning of discriminative local information via weak supervision – meaning that only class labels are given. Note that noises of irrelevant cues from contexts are unavoidable when building the local contrastive alignment, we thus incorporate a semantic attention component to suppress them. We carry out extensive experiments on four FSAR benchmarks including Kinetics, SSv2, UCF-101 and HMDB-51. Our approach achieves state-of-the-art results under various few-shot settings, and superior performance in the more challenging cross-domain evaluation as well.

In summary, our contributions are three-fold:

- We propose a hierarchical matching model for FSAR. The hierarchical architecture utilizes a zoom-in matching module to alleviate the complexity and computation cost for multi-level matching, therefore video similarities using coarse-to-fine cues can be measured.
- We propose the mixed-supervised hierarchical contrastive learning (HCL) to learn generalizable and fine-grained video representations, by using cycle consistency for weakly-supervised spatial-temporal association. Additionally, a semantic attention component is applied to suppress contextual noises.
- We carry out experiments on four benchmark datasets to validate our model, which achieve state-of-the-art performance especially under the 1-shot setup.

## 2 Related Work

**Action Recognition** has received significant improvements thanks to deep neural networks [18, 15]. Early deep models [16, 31, 42] adopt 2D CNNs in temporal domain. 3D CNNs [38] are then proposed to encode short-range temporal dynamics in videos. Just to name a few, Carreira *et al.* [5] propose I3D to inflate 2D CNN to 3D CNN; Tran *et al.* [39] and Qiu *et al.* [30] decompose 3D convolution into 2D and 1D convolutions for efficiency; Wang *et al.* [43] insert non-local blocks into 3D CNNs. More recently, transformer architectures [1, 2] are exploited in video domain to capture long-range dependency. Despite strong performance, these models are hard to generalize to new action classes with limited examples. In this work, we focus on few-shot action recognition.

**Few-shot Learning** approaches can be categorized into three types: generative methods [22, 26], optimization-based methods [10, 44], and metric-based methods [32, 41, 35, 36, 24]. The generative method synthesizes new data of few-shot

classes to enlarge the training data. The optimization-based method learns a good initialization of the network, which can be easily fine-tuned to an unseen target task without sufficient labels. Instead, metric-based method aims to learn a metric to measure similarities of images or videos, and then employs nearest neighboring for classification. Most existing few-shot action recognition works [48, 4, 8] follow the metric-based approach. Fu *et al.* [11] employ global video features on RGB and depth modalities for similarity measure. To capture temporal dynamics in the video, Zhu *et al.* [48] use a compound memory network to reserve the representation of key frames. Zhang *et al.* [47] align short-range while discarding long-range dependencies using a permutation invariant attention with jigsaws for self-supervised training. Cao *et al.* [4] propose to minimize the temporal distance of pairwise video sequences based on the DTW algorithm. Perrett *et al.* [29] use attention mechanism to construct query-specific class prototype for clip matching. Different from previous works, we exploit a hierarchical matching to capture from coarse to fine information for comparison of videos.

**Contrastive Learning** has shown great capability to learn generic representations from unlabeled data [14, 28] in recent years. Wu *et al.* [45] aim to push different augmentations of an instance closer in the embedding space using a memory bank to store instance vectors, which is followed by several works [46, 37, 27]. Khosla *et al.* [17] extend to learn contrast under class supervision. Recent works [12, 34] point out that contrastive learning helps to avoid few-shot learning from limitations like over-fitting [6, 21] or supervision collapse [8], which serves as auxiliary losses to learn the representation alignment.

### 3 Method

**Problem Formulation.** In few-shot action recognition (FSAR) setting, videos in a dataset are split into two sets  $\mathcal{D}_{base}$  and  $\mathcal{D}_{novel}$ . Action classes in  $\mathcal{D}_{base}$  and  $\mathcal{D}_{novel}$  are disjoint.  $\mathcal{D}_{base}$  contains abundant labeled videos per action class and is used for training, while  $\mathcal{D}_{novel}$  is used to evaluate few-shot learning performance in a  $N$ -way  $K$ -shot manner. Such evaluation consists of a series of tasks called episodes [32]. For each episode, we randomly sample  $N$  action classes with  $K$  videos per class from  $\mathcal{D}_{novel}$  as “support set”. The rest videos of the  $N$  action classes in  $\mathcal{D}_{novel}$  are used to sample “query set” for testing. A model is evaluated by averaging recognition performances over all episodes.

**Our Idea.** We follow the metric-based methods [32, 41] to learn a metric space based on  $\mathcal{D}_{base}$ , where classification can be performed by computing similarities among videos in the query set and support set. However, previous works fail to optimize coarse-to-fine representations with multi-level alignment. In this work, we propose to leverage multi-level matching at global, temporal and spatial levels, by developing a hierarchical matching model paired with a mixed-supervised hierarchical contrastive learning (Sec 3.3). Our hierarchical matching model consists of a video encoder to extract multi-level visual cues (Sec 3.1) and a zoom-in matching module to measure video similarities hierarchically (Sec 3.2).

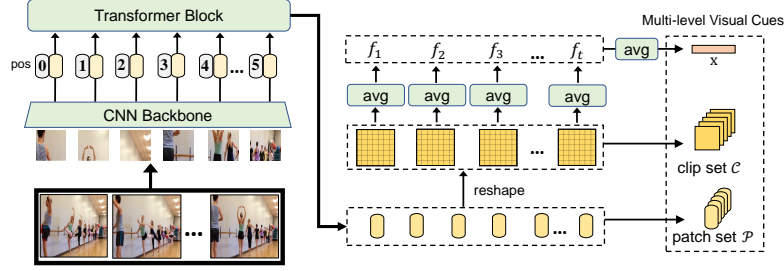


Fig. 2: Network structure of the video encoder. It firstly encodes a video into a sequence of contextualized patches and then generates global embedding  $x$ , temporal embeddings  $\mathcal{C}$  and spatial embeddings  $\mathcal{P}$  for the video.

### 3.1 Video Encoder

The video encoder contains a CNN backbone [15] and a transformer block [40] to extract contextualized video representations as shown in Fig. 2. To be specific, we uniformly sample  $t$  frames as inputs for each video. The CNN backbone extracts a feature map with size  $h \times w$  for each frame. We flatten feature maps of all frames into a sequence of  $t \times h \times w$  patches. Then the transformer block encodes the space-time position [2] of each patch and employs self-attention to model space-time relationships among all the patches. Let  $\mathcal{P} = \{p_1, p_2, \dots, p_{thw}\}$  be the output embeddings of all patches, where  $p_i \in \mathbb{R}^d$  and  $d$  is the dimensionality. We adopt average pooling on the spatial dimension  $h \times w$  per frame to obtain frame features  $\mathcal{F} = \{f_1, f_2, \dots, f_t\}$ ,  $f_i \in \mathbb{R}^d$ . In this paper, we define multi-level visual cues for the following zoom-in matching.

First, we apply average pooling over all frame embeddings to generate a global representation  $x$  for the video, which is prone to lose fine-grained temporal and spatial details. Second, to capture temporal sensitive cues, we sample  $N_c$  clips  $\mathcal{C} = \{c_1, c_2, \dots, c_{N_c}\}$  from continuous frames to capture various temporal scales in the video similar to [29]. A clip  $c_i = \{f_{i_1}, f_{i_2}, \dots, f_{i_{|c_i|}}\}$  is a subset of  $\mathcal{F}$  with  $|c_i|$  frames and its embedding is computed as follows to keep the temporal order:

$$c_i = \text{MLP}([f_{i_1}; f_{i_2}; \dots; f_{i_{|c_i|}}]), c_i \in \mathbb{R}^d \quad (1)$$

where  $[\cdot]$  denotes vector concatenation and MLP is a multi-layer perceptron. Note that we reuse  $c_i$  to denote both clip and its embedding and so does the patch  $p_i$ . Finally, we use patch embeddings  $\mathcal{P}$  to provide spatial visual cues.

### 3.2 Zoom-in Matching Module

Given the above multi-level representations, *i.e.*, global embedding  $x$ , temporal embeddings  $\mathcal{C}$  and spatial embeddings  $\mathcal{P}$ , we progressively zoom-in to measure similarities of a query video  $v$  and a support video  $\hat{v}$  at three coarse-to-fine levels.

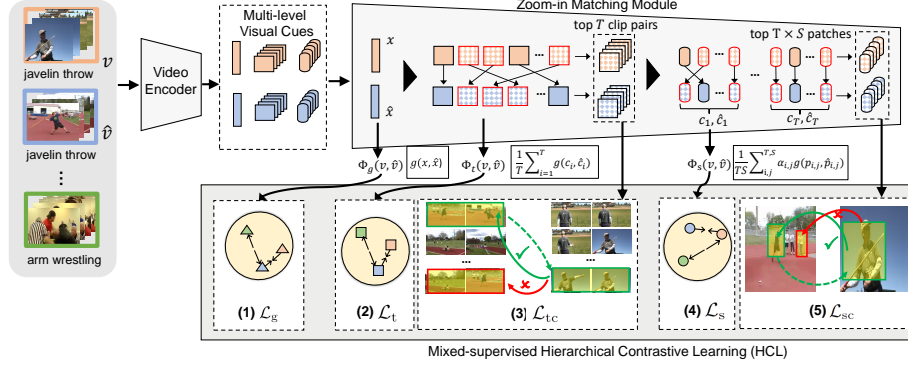


Fig. 3: Illustration of our model. *Top*: hierarchical matching with a zoom-in module to compare coarse-to-fine video similarities, using multi-level visual cues including global embedding  $x$ , clip embedding  $c_i$  and patch embedding  $p_{i,j}$ ; *Bottom*: Mixed-supervised hierarchical contrastive learning (HCL) including five contrastive loss terms  $\mathcal{L}_g$ ,  $\mathcal{L}_t$ ,  $\mathcal{L}_s$  and  $\mathcal{L}_{tc}$ ,  $\mathcal{L}_{sc}$  that are described in Sec. 3.3.

**Global Matching.** We directly compute a cosine similarity  $g(\cdot)$  between  $x$  and  $\hat{x}$  for global matching, which is written by:

$$\Phi_g(v, \hat{v}) = g(x, \hat{x}) = \frac{x}{\|x\|} \odot \frac{\hat{x}}{\|\hat{x}\|}, \quad (2)$$

where  $\|\cdot\|$  means L2 norm and  $\odot$  denotes inner-product operation.

**Temporal Matching.** Temporal information is important to distinguish actions especially for those with similar objects but in different temporal orders, such as “open the door” and “close the door”. We therefore propose to match videos in a finer-grained clip level which captures local temporal dynamics. We use clip features  $\mathcal{C}, \hat{\mathcal{C}}$  to compute temporal matching scores between  $v$  and  $\hat{v}$ . To be specific, for each  $c_i \in \mathcal{C}$ , we pick its most similar clip in  $\hat{\mathcal{C}}$  and form a temporally matched pair  $(c_i, \hat{c}_i)$ . We rank all pairs by their feature similarity and select top  $T$  pairs to compute the temporal matching score:

$$\Phi_t(v, \hat{v}) = \frac{1}{T} \sum_{i=1}^T g(c_i, \hat{c}_i). \quad (3)$$

**Spatial Matching.** The discriminative spatial regions to differentiate actions can be small, such as “eat burger” vs. “eat doughnuts”, making the spatially condensed embeddings  $\mathcal{C}$  less effective to capture such fine-grained information. We further apply spatial matching between patches from the temporally aligned clip pairs  $(c_i, \hat{c}_i), i \in [1, T]$  mentioned above. By doing so, we avoid to enumerate all possible patch-to-patch alignment in the entire video, which contains a numerous number of noisy information with a large burden of computation cost. Similar to temporal matching, for each picked clip pair, we align each patch  $p_{i,j}$  in  $c_i$  with the most similar patch  $\hat{p}_{i,j}$  in clip  $\hat{c}_i$ , and select top  $S$  aligned patches

by the similarity score. In this way, we obtain  $T \times S$  patch pairs from the video to compute the spatial matching score as follows, where  $\alpha_{i,j} = 1$  if semantic attention component is not used.

$$\Phi_s(v, \hat{v}) = \frac{1}{TS} \sum_{i=1}^T \sum_{j=1}^S \alpha_{i,j} g(p_{i,j}, \hat{p}_{i,j}). \quad (4)$$

**Semantic Attention Component.** Not all aligned patches with high similarity are relevant to the action. For example, videos with similar backgrounds are likely to rank background patch pairs on the top. To suppress noises from semantically irrelevant patch pairs, we propose to re-weight the semantic correlation of each patch pair with the action. In particular, assume the action class of the support video  $\hat{v}$  is  $\hat{y}$ , we use BERT [7] to obtain its class embedding as  $e_{\hat{y}}$ . Then the semantic attention weight of patch  $p_{i,j}$  in clip  $c_i$  is reassigned as:

$$\alpha_{i,j} = \frac{\exp(p_{i,j} W e_{\hat{y}} / \sqrt{d})}{\sum_{k=1}^{N_p} \exp(p_{i,k} W e_{\hat{y}} / \sqrt{d})}, \quad (5)$$

where  $W$  denotes a projection matrix and  $N_p$  is the number of patches in clip  $c_i$ ,  $d$  is the dimensionality. The  $\alpha_{i,j}$  added in Eq. 4 emphasizes semantically salient patches and disregard irrelevant background noises in matching.

The final matching score  $\Phi(v, \hat{v})$  between video  $v$  and  $\hat{v}$  is aggregated from the three hierarchical matching scores as follows. We use  $\Phi(v, \hat{v})$  to compare the similarity between any videos during the **evaluation** and **inference**.

$$\Phi(v, \hat{v}) = \Phi_g(v, \hat{v}) + \Phi_t(v, \hat{v}) + \Phi_s(v, \hat{v}). \quad (6)$$

**Computation Cost Analysis** The zoom-in module mainly reduces the cost of spatial matching. Assuming we enumerate all clips with 2 frames in the video for the pairwise matching ( $C_t^2$  clips per video). The zoom-in module applies temporal matching across video clips and then selects the top- $T$  aligned clips for spatial matching ( $T \ll C_t^2$ ). Hence, the computation complexity for spatial matching is  $\mathcal{O}(T^2 h^2 w^2)$ . The model without zoom-in module, however, applies spatial matching for all video clips instead of the top ones. Therefore, the complexity is  $\mathcal{O}(t^4 h^2 w^2)$ , which is more computationally expensive.

### 3.3 Mixed-supervised Hierarchical Contrastive Learning

In order to learn coarse-to-fine representations, we propose mixed-supervised hierarchical contrastive learning (HCL) as shown in Fig. 3 for training visual cues of temporal and spatial levels. Apart from supervised constrastive learning to differentiate videos of different classes, our HCL further utilizes cycle consistency to enable spatio-temporal constrastive learning in a weakly-supervised manner.

**Supervised Contrastive Learning.** Given a mini-batch of  $B$  videos, we compute the global similarity  $\Phi_g(v_i, v_j)$  between any two videos  $v_i$  and  $v_j$  in the

batch. A video pair  $(v_i, v_j)$  where  $i, j \in [1, B], i \neq j$  is positive only when  $y_i = y_j$ , otherwise it is negative. The global contrastive loss is then written as follows:

$$\mathcal{L}_g = -\frac{1}{B^2} \sum_{i \neq j} \mathbb{1}_{y_i = y_j} \log \frac{\exp(\Phi_g(v_i, v_j)/\tau)}{\sum_{k=1}^B \mathbb{1}_{k \neq i} \exp(\Phi_g(v_i, v_k)/\tau)}, \quad (7)$$

where  $\tau$  is temperature hyper-parameter and  $\mathbb{1}$  is an indicator function. To be noted, our supervised contrastive learning is different from previous works based on episodic training [32], which only allows negative examples within the  $N$  video classes in each episode. Our training instead contains more diverse negative examples, which are demonstrated to be beneficial for representation learning [6, 23, 25]. Similarly, we use the temporal matching score  $\Phi_t(v_i, v_j)$  and spatial matching score  $\Phi_s(v_i, v_j)$  to compute  $\mathcal{L}_t$  and  $\mathcal{L}_s$  respectively as Eq (7).

**Weakly-supervised Contrastive Learning via Cycle Consistency.** The temporal and spatial matching relies on fine-grained alignment of features at the clip and patch level respectively. To enhance such alignment, we propose to leverage cycle consistency in temporal and spatial contrastive learning. Given video  $v$  and  $\hat{v}$  of the same action class, we build temporal cycle consistency [9] of their top  $T$  aligned clip pairs as supervision for training. For each clip  $c_i \in \mathcal{C}$ , we first compute its soft nearest neighbor  $\hat{c}_{j^*} \in \hat{\mathcal{C}}$ , which is:

$$\hat{c}_{j^*} = \sum_{j=1}^{N_c} \hat{\beta}_j \hat{c}_j, \text{ where } \hat{\beta}_j = \frac{\exp(g(c_i, \hat{c}_j))}{\sum_{k=1}^{N_c} \exp(g(c_i, \hat{c}_k))}. \quad (8)$$

$N_c$  is the clip number of video. Then we track back  $\hat{c}_{j^*}$  to find its nearest neighbor  $c_{i^*}$  in  $v$ . If the alignment is well trained, the pair  $(c_i, c_{i^*})$  should satisfy the cycle consistency so that  $c_i = c_{i^*}$ . Therefore, the temporal cycle consistency loss is:

$$\mathcal{L}_{tc}(v, \hat{v}) = -\frac{1}{T} \sum_{i=1}^T \log \frac{\exp(g(\hat{c}_{j^*}, c_i))}{\sum_{k=1}^T \exp(g(\hat{c}_{j^*}, c_k))}. \quad (9)$$

The temporal cycle consistency allows to learn from clip-to-clip association to improve the temporal alignment. We average such losses of all pairwise videos of the same class in a mini-batch as  $\mathcal{L}_{tc}$ .

It is however more challenging to extend the temporal cycle consistency in the spatial domain. Similar to challenges in spatial matching, firstly, searching all patches in videos is computationally expensive. Secondly, it is also unnecessary to enforce every patch to satisfy cycle consistency, e.g., semantically irrelevant patches. Therefore, we only build such patch-level consistency for the top  $T$  similar clip pairs from two videos of the same class. For each clip pair  $(c, \hat{c})$ , the spatial cycle consistency is built on top of their patch sets:

$$\mathcal{L}_{sc}(c, \hat{c}) = -\frac{1}{N_p} \sum_{i=1}^{N_p} \alpha_i \log \frac{\exp(g(\hat{p}_{j^*}, p_i))}{\sum_{k=1}^{N_p} \exp(g(\hat{p}_{j^*}, p_k))}, \quad (10)$$

where  $\hat{p}_{j^*}$  is the soft nearest neighbor computed similarly as Eq (8),  $N_p$  is the patch number of a clip,  $\alpha_i$  is the semantic attention weight using Eq. 5. When



$\alpha_i$  is small, the gradient will be down weighted because it implies that the patch  $p_i$  has weak semantic association with the action. We average the loss for all selected clip pairs in a batch as  $\mathcal{L}_{sc}$ .

We combine all these contrastive losses and the traditional supervised cross-entropy loss  $\mathcal{L}_{ce} = -\log p(y|x)$  as the following overall training objective, where  $\lambda_g$ ,  $\lambda_t$ ,  $\lambda_s$  are hyper-parameters to balance the losses for multi-scale visual cues:

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda_g \mathcal{L}_g + \lambda_t (\mathcal{L}_t + \mathcal{L}_{tc}) + \lambda_s (\mathcal{L}_s + \mathcal{L}_{sc}), \quad (11)$$

## 4 Experiments

### 4.1 Experimental Setup

**Datasets.** We conduct experiments on four datasets, including Kinetics [5], Something v2 (SSv2) [13], HMDB-51 [19], and UCF-101 [33]. Kinetics and SSv2 are the most widely used benchmarks for few-shot action recognition. For Kinetics benchmark, we follow the split in [48] for fair comparison. It uses a subset of Kinetics by selecting 100 action classes with 100 videos per class from the whole dataset. The 100 classes are split into 64, 12 and 24 classes as the training, validation and testing set respectively. For SSv2 benchmark, we adopt two splits proposed in [49] and [4] denoted as SSv2<sup>†</sup> and SSv2\* respectively. SSv2\* contains nearly 70,000 training samples for 64 training classes. Each class has over 1,000 training samples on average which is 10 times larger than class samples in SSv2<sup>†</sup>. For HMDB-51 and UCF-101, we use the split from [47].

**Implementation Details.** We use ResNet-50 [15] pre-trained on ImageNet [18] as CNN backbone for fair comparison with previous works [48, 3, 8]. The semantic embeddings for action classes are obtained from a pretrained BERT [7]. For each video, we uniformly sample 8 frames and resize the frame scale into  $224 \times 224$ . The number of clips and patches selected in temporal and spatial matching is  $T = 10$  and  $S = 10$ . During training, the weight of  $\lambda_g$ ,  $\lambda_t$  and  $\lambda_s$  for hierarchical contrastive loss is set as 0.5, 0.3 and 0.3. We train our model for 15 epochs with 3,000 steps for each epoch. Our model is optimized via SGD with the learning rate of 0.001, which is decayed every 6 epochs by 0.5. We randomly sample 24 classes with 2 videos per class in a mini-batch. We provide more details and our codes in the supplementary material.

**Evaluation Protocol.** We evaluate the performance of our model under 5-way  $K$ -shot setup with  $K \in \{1, 2, 3, 4, 5\}$ . We randomly sample 10,000 episodes from  $\mathcal{D}_{novel}$  in testing. The performance is the average of all episodes.

### 4.2 Ablation Study

**Q1: Is hierarchical contrastive learning more effective than traditional training methods?** We compare different variants of HCL training losses and the traditional cross-entropy loss in Tab. 1. Please note that the temporal or spatial matching will be removed during inference if the corresponding contrastive

Table 1: Ablation of training objectives.  $\mathcal{L}_{ce}$  denotes cross entropy loss,  $\mathcal{L}_g$  is global contrastive loss,  $\mathcal{L}_t + \mathcal{L}_{tc}$  and  $\mathcal{L}_s + \mathcal{L}_{sc}$  represent temporal/spatial contrastive loss enhanced with cycle consistency loss.

	HCL				Kinetics		SSv2*	
	$\mathcal{L}_{ce}$	$\mathcal{L}_g$	$\mathcal{L}_t + \mathcal{L}_{tc}$	$\mathcal{L}_s + \mathcal{L}_{sc}$	1-shot	5-shot	1-shot	5-shot
1	×	×	×	×	59.9	80.1	29.5	44.1
2	✓	×	×	×	62.5	81.6	37.8	55.0
3	×	✓	×	×	54.9	76.4	34.0	50.9
4	✓	✓	×	×	66.1	82.7	40.3	56.7
5	✓	✓	✓	×	70.4	83.9	45.4	62.6
6	✓	✓	×	✓	72.3	84.7	41.5	58.4
7	✓	✓	✓	✓	73.7	85.8	47.3	64.9

loss is not used in training. Row 1 simply adopts a pretrained ResNet-50 to extract global representations and does not involve any training on the video dataset. It already achieves 59.9% and 80.1% accuracy under 1-shot and 5-shot setups on Kinetics, which serves as a strong baseline. Row 2 adds a spatial-temporal transformer on top of the CNN backbone and fine-tunes the whole model via  $\mathcal{L}_{ce}$ . The temporal information and fine-tuning brings stable improvements over row 1 especially on SSv2\* which focuses more on temporal orders.

In row 3, we use the global contrastive loss  $\mathcal{L}_g$  alone for training, which however obtains poor performance on Kinetics even compared with the model without fine-tuning in row 1. Combining  $\mathcal{L}_g$  with  $\mathcal{L}_{ce}$  performs better compared to using them separately, showing the two types of training objectives are complementary.  $\mathcal{L}_{ce}$  alone may suffer from over-fitting especially on Kinetics while  $\mathcal{L}_g$  can improve the generalization of the learned features. Both the temporal and spatial contrastive learning are beneficial as shown in row 5 and 6 respectively. Using both temporal contrastive loss and its corresponding cycle consistency loss,  $\mathcal{L}_t + \mathcal{L}_{tc}$  brings significant improvements especially on SSv2\* with +5.1% for 1-shot and +5.9% for 5-shot setups. On the opposite,  $\mathcal{L}_s + \mathcal{L}_{sc}$  is more effective on Kinetic dataset with +6.2% for 1-shot and +2.0% for 5-shot. The results align with our observation that SSv2\* focuses more on the temporal orders and Kinetics is more discriminative in the spatial dimension. Finally, we achieve the best results by combining  $\mathcal{L}_g$ ,  $\mathcal{L}_t + \mathcal{L}_{tc}$  and  $\mathcal{L}_s + \mathcal{L}_{sc}$  in row 7.

Table 2: Ablation of temporal and spatial cycle consistency losses.

		Kinetics		SSv2*		
$\mathcal{L}_{tc}$	$\mathcal{L}_{sc}$	1-shot	5-shot	1-shot	5-shot	
1	×	×	72.1	84.3	44.7	62.5
2	✓	×	72.4	84.7	46.9	64.5
3	×	✓	73.1	85.2	46.2	64.0
4	✓	✓	73.7	85.8	47.3	64.9

Table 3: Ablation of semantic attention module.

		Kinetics		SSv2*	
semantic_att		1-shot	5-shot	1-shot	5-shot
1	×	71.3	84.4	46.3	64.2
2	✓	73.7	85.8	47.3	64.9

**Q2: Is spatio-temporal cycle consistency beneficial to hierarchical contrastive learning?** In Tab. 2, we compare models with and without temporal and spatial cycle consistency loss  $\mathcal{L}_{tc}$ ,  $\mathcal{L}_{sc}$ . Without  $\mathcal{L}_{tc}$ , our model’s performance on SSv2\* decreases with -1.5% for 1-shot and -0.9% for 5-shot (row 3 vs. row 4). Significant performance degradation can also be observed on Kinetics by removing  $\mathcal{L}_{sc}$  (row 2 vs. row 4). These results indicate that both temporal and spatial cycle consistency losses are beneficial to learning fine-grained association.

**Q3: Does semantic attention component help spatial matching and spatial cycle consistency training?** In Tab. 3, we validate the contribution of semantic attention component for spatial matching in Eq. (4). By removing the semantic attention, the performance of our model on Kinetics drops by -2.4% for 1-shot and -1.4% for 5-shot. Note that the semantic attention weight in Eq. 10 will also be removed. The results demonstrate that re-scaling semantic weight is helpful in learning spatial associations by focusing on semantically relevant patches and eliminating background noises. On SSv2\*, only slight improvement can be observed due to its temporal inclination.

Table 4: Ablation of zoom-in matching at different levels using  $\mathcal{L}_{ce}$  or hierarchical contrastive loss (HCL), where G, T and S denote the global, temporal and spatial matching respectively

(a) Cross Entropy Loss $\mathcal{L}_{ce}$							(b) Hierarchical Contrastive Loss								
Match			Kinetics		SSv2*		Match			Kinetics		SSv2*			
G	T	S	1-shot	5-shot	1-shot	5-shot	G	T	S	1-shot	5-shot	1-shot	5-shot		
1	✓	×	×	62.5	81.6	37.8	55.0	1	✓	×	×	69.1	83.2	42.9	59.2
2	×	✓	×	62.1	81.3	40.5	57.0	2	×	✓	×	71.3	84.2	45.9	62.9
3	×	×	✓	59.0	79.8	33.2	49.5	3	×	×	✓	71.8	84.5	38.1	53.2
4	✓	✓	×	62.9	81.8	40.2	56.6	4	✓	✓	×	71.5	84.3	46.4	63.5
5	✓	×	✓	63.8	82.2	38.5	55.4	5	✓	×	✓	72.7	84.9	43.7	59.8
6	✓	✓	✓	64.4	82.5	40.8	57.3	6	✓	✓	✓	73.7	85.8	47.3	64.9

**Q4: What is the performance of zoom-in matching at different levels?** In Tab. 4, we explore different combinations of zoom-in matching at test time. Tab. 4 (a) uses  $\mathcal{L}_{ce}$  for training. We can see temporal or spatial matching alone does not outperform global matching on Kinetics. Tab. 4 (b) employs our HCL training algorithm. Instead, the temporal or spatial matching achieves superior performance on Kinetics, which proves that HCL is beneficial to learn fine-grained alignment. The two matching’s improvement is more significant on SSv2\* and Kinetics respectively, since SSv2\* mainly focuses on temporal variation while spatial cue plays a more important role on Kinetics. In addition, the combination of global, temporal and spatial matching improves individual performances whether using  $\mathcal{L}_{ce}$  or our HCL. It shows that different levels are complementary with each other and zoom-in matching needs to equip with HCL for effective hierarchical alignment.

Table 5: Comparison with SOTA methods on Kinetics and SSv2.

Match	Method	Kinetics			SSv2 <sup>†</sup>			SSv2*		
		1-shot	2-shot	5-shot	1-shot	2-shot	5-shot	1-shot	2-shot	5-shot
Global	MAML [10]	54.2	65.5	75.3	30.9	35.1	41.9	-	-	-
	ProtoNet [32]	59.1	73.6	83.5	30.9	37.2	47.2	34.0	41.2	51.7
	TARN [3]	66.6	74.6	80.7	-	-	-	-	-	-
Temporal	CMN [48]	60.5	70	78.9	36.2	42.1	48.8	-	-	-
	TAM [4]	73.0	-	85.8	-	-	-	42.8	-	52.3
	TRX [29]	64.6	76.4	85.5	34.7	43.5	<b>56.8</b>	38.1	49.1	63.9
Hierarchical	ours	<b>73.7</b>	<b>79.1</b>	<b>85.8</b>	<b>38.7</b>	<b>45.5</b>	55.4	<b>47.3</b>	<b>54.5</b>	<b>64.9</b>

### 4.3 Comparison with State-of-the-Art Methods

In Tab. 5, we compare our method with state-of-the-art approaches on Kinetics and SSv2 benchmarks. The global matching approaches [32, 20] are less competitive to temporal matching approaches [4, 8] and our hierarchical model in general. Our proposed model outperforms previous temporal approaches by a large margin under 1-shot and 2-shot evaluations and is comparable under 5-shot setting on all datasets. When labels are extremely limited as in the 1-shot setting, our model achieves +9.1%, +4.0% and +9.2% improvements on Kinetics, SSv2<sup>†</sup> and SSv2\* respectively compared to TRX [8]. The improvements from our model are more significant on SSv2 benchmarks. For example, though our model slightly outperforms TAM [4] by 0.7% under 1-shot setting on Kinetics, it beats TAM [4] by +4.5% and +12.6% under 1-shot and 5-shot settings on SSv2\*, which indicates that our method has stronger capability of temporal reasoning. In addition, the performance is more encouraging on SSv2\*, where we obtain significant improvements under all settings from 1-shot to 5-shot. Considering SSv2\* has more training samples (more than 70,000 videos) than other datasets like Kinetics (7,600 videos), we believe that our HCL is able to benefit more from large-scale datasets compared with other approaches.

Table 6: Comparison with SOTA methods on UCF-101 and HMDB-51.

Match	Method	UCF-101		HMDB-51	
		1-shot	5-shot	1-shot	5-shot
Global	ProtoNet [32]	67.2	93.0	44.2	72.0
Temporal	TARN [3]	66.3	-	45.5	60.6
Temporal	TRX [29]	81.3	<b>95.9</b>	52.0	75.6
Hierarchical	ours	<b>82.6</b>	94.5	<b>59.1</b>	<b>76.3</b>

We further provide comparisons on UCF-101 and HMDB-51 in Tab. 6, which contain much less training data than Kinetics and SSv2. Our method significantly improves over TRX [8] on HMDB-51 e.g., +7.1% for 1-shot and +0.7%

for 5-shot. On UCF-101, HCL shows improvement for 1-shot but a slight decrease for 5-shot over TRX [8]. In general, our model is robust to various action categories, whether they focus on spatial information (e.g., Kinetics) or temporal orders (e.g., SSv2). Our model is more effective when the training classes have abundant samples in  $\mathcal{D}_{base}$  and the test classes have extremely few samples in  $\mathcal{D}_{novel}$  (e.g., SSv2\*), which is exactly the situation in real applications.

Table 7: Comparison on more challenging cross-domain evaluation setting.

Method	Kinetics $\rightarrow$ UCF-101		Kinetics $\rightarrow$ HMDB-51	
	1-shot	5-shot	1-shot	5-shot
MAML [10]	62.4	80.7	43.9	59.3
ProtoNet [32]	67.8	84.2	48.5	63.0
ours	76.1	90.6	54.2	69.5

#### 4.4 Cross-Domain Evaluation

To further validate the generalization capability of our model, we design a new cross-domain FSAR setting similar to [6]. We use the training split in Kinetics as  $\mathcal{D}_{base}$  and the testing splits in UCF-101 and HMDB-51 as  $\mathcal{D}_{novel}$ . Then we remove overlapped classes between Kinetics training set and the testing set. Such evaluation is more challenging, which requires the learned model not only generalizes on new action classes but also on new video domains. We compare our model with an optimization-based model MAML [10] and a metric-based model ProtoNet [32]. Table 7 presents the cross-domain results. We achieve significantly better performances than the other methods, with 8.3% and 5.7% gains under 1-shot setting on UCF-101 and HMDB-51 datasets respectively. It proves that our model can adapt well to novel actions in different domains from the base classes in the training set.

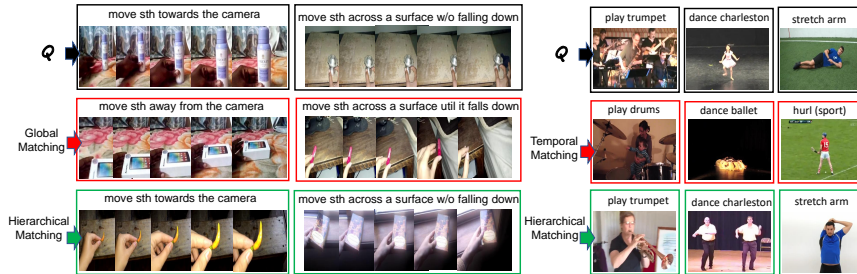


Fig. 4: Global/temporal matching vs. hierarchical matching. We show the most similar video in support sets for each query video using the matching approach.



Fig. 5: Discriminative patch pairs between query and support videos.  $Q$  denotes the query,  $S_1, S_2$  and  $S_3$  are three distinct support videos from the same class.

#### 4.5 Quality Analysis

Fig. 4 provides qualitative comparisons for global, temporal matching with our hierarchical matching. Global matching fails to differentiate videos with similar appearances but different temporal orders, while temporal matching fails in recognizing detailed spatial information. Our hierarchical matching considers both temporal orders and discriminative spatial patches, and thus it can classify videos more accurately. Fig. 5 presents examples of discriminative patch pairs between query and support videos in spatial matching with our model. First, our model is able to select semantically relevant pairs in matching. For example, it selects patches of the person’s hand and the instrument in “play ukulele” action, and patches of the person and an car in “push car” action. Secondly, our model can effectively align patches with other videos in the support set.

### 5 Conclusion

In this paper, we propose a hierarchical matching approach for few-shot action recognition. Our model, equipped with a zoom-in matching module, hierarchically build coarse-to-fine alignment between videos without complex computation. Therefore, video similarities on few sets can be measured from multiple levels. Moreover, to learn discriminative temporal and spatial associations, we propose a mixed-supervised hierarchical contrastive learning (HCL) algorithm, which utilizes cycle consistency as weak supervision to combine with supervised learning. We carry out extensive experiments to evaluate our proposed model on four benchmark datasets. Our model achieves the state-of-the-art performances especially under 1-shot setting. It shows better generalization capacity in a more challenging cross-domain evaluation as well.

**Acknowledgment** This work was partially supported by National Natural Science Foundation of China (No. 62072462), National Key R&D Program of China (No. 2020AAA0108600), and Large-Scale Pre-Training Program 468 of Beijing Academy of Artificial Intelligence (BAAI).

## References

1. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C.: Vivit: A video vision transformer. ICCV (2021)
2. Bertasius, G., Wang, H., Torresani, L.: Is space-time attention all you need for video understanding? ICML (2021)
3. Bishay, M., Zoumpourlis, G., Patras, I.: Tarn: Temporal attentive relation network for few-shot and zero-shot action recognition. BMVC (2019)
4. Cao, K., Ji, J., Cao, Z., Chang, C.Y., Niebles, J.C.: Few-shot video classification via temporal alignment. CVPR (2020)
5. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. CVPR (2017)
6. Chen, W.Y., Liu, Y.C., Kira, Z., Wang, Y.C.F., Huang, J.B.: A closer look at few-shot classification. ICLR (2019)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. NAACL (2019)
8. Doersch, C., Gupta, A., Zisserman, A.: Crosstransformers: spatially-aware few-shot transfer. NeurIPS (2020)
9. Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., Zisserman, A.: Temporal cycle-consistency learning. CVPR (2019)
10. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. ICML (2017)
11. Fu, Y., Zhang, L., Wang, J., Fu, Y., Jiang, Y.G.: Depth guided adaptive meta-fusion network for few-shot video recognition. ACMMM (2020)
12. Gidaris, S., Bursuc, A., Komodakis, N., Pérez, P., Cord, M.: Boosting few-shot visual learning with self-supervision. ICCV (2019)
13. Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al.: The” something something” video database for learning and evaluating visual common sense. ICCV (2017)
14. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. CVPR (2006)
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. CVPR (2016)
16. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. CVPR (2014)
17. Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., Krishnan, D.: Supervised contrastive learning. arXiv preprint arXiv:2004.11362 (2020)
18. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. NeurIPS (2012)
19. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. ICCV (2011)
20. Kumar Dwivedi, S., Gupta, V., Mitra, R., Ahmed, S., Jain, A.: Protogan: Towards few shot learning for action recognition. ICCV Workshops (2019)
21. Laenen, S., Bertinetto, L.: On episodes, prototypical networks, and few-shot learning. In: Thirty-Fifth Conference on Neural Information Processing Systems (2021)
22. Lake, B., Salakhutdinov, R., Gross, J., Tenenbaum, J.: One shot learning of simple visual concepts. CogSci (2011)

23. Li, W., Wang, L., Xu, J., Huo, J., Gao, Y., Luo, J.: Revisiting local descriptor based image-to-class measure for few-shot learning. CVPR (2019)
24. Liu, C., Xu, C., Wang, Y., Zhang, L., Fu, Y.: An embarrassingly simple baseline to one-shot learning. CVPR (2020)
25. Majumder, O., Ravichandran, A., Maji, S., Polito, M., Bhotika, R., Soatto, S.: Supervised momentum contrastive learning for few-shot classification. arXiv preprint arXiv:2101.11058 (2021)
26. Miller, E.G., Matsakis, N.E., Viola, P.A.: Learning from one example through shared densities on transforms. CVPR (2000)
27. Misra, I., Maaten, L.v.d.: Self-supervised learning of pretext-invariant representations. CVPR (2020)
28. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
29. Perrett, T., Masullo, A., Burghardt, T., Mirmehdi, M., Damen, D.: Temporal-relational crosstransformers for few-shot action recognition. CVPR (2021)
30. Qiu, Z., Yao, T., Mei, T.: Learning spatio-temporal representation with pseudo-3d residual networks. ICCV (2017)
31. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. NeurIPS (2014)
32. Snell, J., Swersky, K., Zemel, R.S.: Prototypical networks for few-shot learning. NeurIPS (2017)
33. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
34. Su, J.C., Maji, S., Hariharan, B.: When does self-supervision improve few-shot learning? ECCV (2020)
35. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H.S., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. CVPR (2018)
36. Sung, F., Zhang, L., Xiang, T., Hospedales, T.M., Yang, Y.: Learning to learn: Meta-critic networks for sample efficient learning. IEEE Access (Volume: 7) (2019)
37. Tian, Y., Krishnan, D., Isola, P.: Contrastive multiview coding. ECCV (2019)
38. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. ICCV (2015)
39. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. CVPR (2018)
40. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. NeurIPS (2017)
41. Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., Wierstra, D.: Matching networks for one shot learning. NeurIPS (2016)
42. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks for action recognition in videos. TPAMI (2018)
43. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. CVPR (2018)
44. Wang, Y., Hebert, M.: Learning to learn: Model regression networks for easy small sample learning. ECCV (2016)
45. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. CVPR (2018)
46. Ye, M., Zhang, X., Yuen, P.C., Chang, S.F.: Unsupervised embedding learning via invariant and spreading instance feature. CVPR (2019)
47. Zhang, H., Zhang, L., Qi, X., Li, H., Torr, P.H., Koniusz, P.: Few-shot action recognition with permutation-invariant attention. ECCV (2020)



48. Zhu, L., Yang, Y.: Compound memory networks for few-shot video classification. ECCV (2018)
49. Zhu, L., Yang, Y.: Label independent memory for semi-supervised few-shot video classification. IEEE Annals of the History of Computing pp. 1–1 (2020)