

D²ST-Adapter: Disentangled-and-Deformable Spatio-Temporal Adapter for Few-shot Action Recognition

Wenjie Pei^{†1,2} Qizhong Tan¹ Guangming Lu¹ Jiandong Tian³ Jun Yu¹

¹Harbin Institute of Technology, Shenzhen ²Peng Cheng Laboratory

³Shenyang Institute of Automation, Chinese Academy of Sciences

wenjiecoder@outlook.com, 24B951007@stu.hit.edu.cn

Abstract

Adapting pre-trained image models to video modality has proven to be an effective strategy for robust few-shot action recognition. In this work, we explore the potential of adapter tuning in image-to-video model adaptation and propose a novel video adapter tuning framework, called Disentangled-and-Deformable Spatio-Temporal Adapter (D²ST-Adapter). It features a lightweight design, low adaptation overhead and powerful spatio-temporal feature adaptation capabilities. D²ST-Adapter is structured with an internal dual-pathway architecture that enables built-in disentangled encoding of spatial and temporal features within the adapter, seamlessly integrating into the single-stream feature learning framework of pre-trained image models. In particular, we develop an efficient yet effective implementation of the D²ST-Adapter, incorporating the specially devised anisotropic Deformable Spatio-Temporal Attention as its pivotal operation. This mechanism can be individually tailored for two pathways with anisotropic sampling densities along the spatial and temporal domains in 3D spatio-temporal space, enabling disentangled encoding of spatial and temporal features while maintaining a lightweight design. Extensive experiments by instantiating our method on both pre-trained ResNet and ViT demonstrate the superiority of our method over state-of-the-art methods. Our method is particularly well-suited to challenging scenarios where temporal dynamics are critical for action recognition. Code is available at <https://github.com/qizhongtan/D2ST-Adapter>.

1. Introduction

Few-shot action recognition aims to learn an action recognition model from a set of base classes of video data, which can recognize novel categories of actions using only a few support samples. To this end, learning an effective feature

extractor that is generalizable across different classes is crucial. A typical way [1, 3, 33, 35, 45] is to leverage pre-trained large image models [10, 24, 29] for feature learning by image-to-video model adaptation.

Most existing methods [23, 34, 35] seek to adapt large vision models to few-shot action recognition by fine-tuning either the entire or partial model. While such methods have achieved promising performance, there are two important limitations that hamper them from exploiting the full potential of the pre-trained models. First, fully fine-tuning pre-trained models tends to overfit in few-shot learning scenarios, whereas partial fine-tuning is less effective due to its limited capacity for adaptation. Second, most large vision models are trained on image data due to the scarcity of video data. Thus, these fine-tuning based methods require to learn the temporal features as a post-processing step after feature extraction by the pre-trained backbone, typically by constructing an auxiliary module to model the temporal dynamics [28, 33, 34, 38], which has limited effectiveness compared to methods that incorporate temporal learning throughout the entire feature learning stage.

As a prominent parameter-efficient fine-tuning technique [9, 11, 13, 50], adapter tuning [4, 11, 32] is particularly well-suited to few-shot learning scenarios due to much lower parameter-learning overhead. A straightforward way to apply adapter tuning to image-to-video model adaptation is to employ the vanilla image adapter [11] to adapt the pre-learned spatial and temporal features separately, which is exactly adopted by AIM [44] and DUALPATH [22]. As shown in Figure 1, both AIM and DUALPATH first pre-learn the spatial and temporal features separately by duplicating the multi-head self-attention modules, either in cascaded manner (AIM) or parallel manner (DUALPATH). Subsequently, they insert individual vanilla adapters in each duplicated module for specific feature adaptation. However, such methods inevitably result in a heavyweight model architecture and substantial computational overhead. As illustrated in Figure 1, ST-Adapter [21] is the first to design a specialized adapter for video data by incorporating a 3D

[†] Corresponding author.

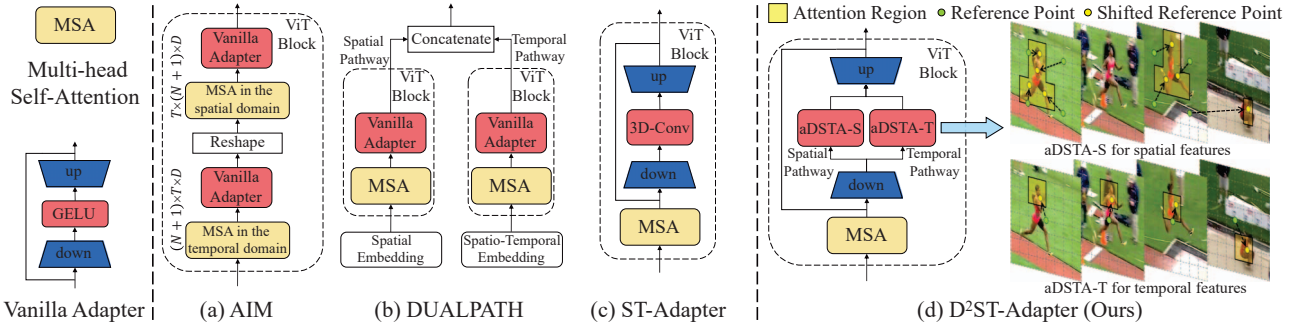


Figure 1. Comparison with previous adapter-based methods. Both AIM and DUALPATH pre-learn the spatial and temporal features separately by duplicating the whole MSA modules, and insert vanilla adapter into each duplicated module for feature adaptation, which incurs heavy-weight model design. ST-Adapter designs the adapter for video data using 3D convolution whose performance is limited by the joint learning of spatio-temporal features. In contrast, our D^2ST -Adapter is designed in a dual-pathway architecture to adapt the spatial and temporal features in a disentangled manner while maintaining the single-stream feature learning framework of the pre-trained model. Furthermore, we design the anisotropic Deformable Spatio-Temporal Attention (aDSTA), which configures anisotropic sampling densities in the spatial and temporal domains to model two pathways specifically and enables D^2ST -Adapter to adapt features in a global view.

convolution layer into the vanilla adapter to learn spatio-temporal features. Nevertheless, we investigate two potential limitations of ST-Adapter. First, it learns the spatial and temporal features jointly by 3D convolution, whereas it has been shown that in low-data scenarios, such a joint learning paradigm for spatio-temporal features in video data is inferior to learning the spatial and temporal features in a disentangled manner, such as SlowFast [6] and other two-stream designs [25, 31]. Second, the convolutional operation, especially with shallow layers for lightweight design, has a limited local receptive field in both the spatial and temporal domains, which also limits the performance of ST-Adapter.

In this work we present the Disentangled-and-Deformable Spatio-Temporal Adapter (D^2ST -Adapter) to tackle the aforementioned limitations. Specifically, we design our D^2ST -Adapter as a dual-pathway architecture, as illustrated in Figure 1, in which the spatial pathway is responsible for capturing the appearance features while the temporal pathway focuses on learning the temporal dynamics. This built-in disentangled encoding of spatio-temporal features equips our D^2ST -Adapter with superior image-to-video model adaptation capabilities while maintaining a lightweight design and low adaptation overhead. As a result, it benefits from a more efficient model design compared to AIM and DUALPATH, which perform separate feature pre-learning and individual adaptation for spatial and temporal features by duplicating learning modules. Meanwhile, our D^2ST -Adapter demonstrates superior feature learning capability for video data compared to ST-Adapter, owing to its disentangled adaptation scheme.

We develop an efficient yet effective implementation of the D^2ST -Adapter, featuring the specially devised anisotropic Deformable Spatio-Temporal Attention (aDSTA) as the pivotal operation to enable built-in disentangled encoding capability. It adapts the deformable attention [40]

from 2D image space to 3D spatio-temporal space, which allows our D^2ST -Adapter to encode features in both spatial and temporal domains in a global view while keeping lightweight modeling. One novel design of our aDSTA is that it can be tailored with anisotropic sampling densities along spatial and temporal domains, enabling specialized versions of aDSTA to model the spatial and temporal pathways separately. As shown in Figure 1, we tailor aDSTA-T with denser sampling along temporal domain than spatial domain for temporal pathway since it focuses on capturing temporal features. In contrast, the tailored aDSTA-S for spatial pathway samples denser points along spatial domain.

To conclude, we make the following contributions:

- We propose D^2ST -Adapter, a novel video adapter tuning framework for few-shot action recognition. It is designed in a dual-pathway architecture featuring built-in disentangled adaptation for spatial and temporal features, enabling effective yet efficient image-to-video model adaptation for few shot action recognition.
- We devise aDSTA, which can be tailored with anisotropic sampling densities along spatial and temporal domains to model the spatial and temporal pathways separately, yielding a lightweight implementation of D^2ST -Adapter.
- Extensive experiments on five benchmarks with instantiations of our method on pre-trained ResNet [10] and ViT [24], demonstrate the superiority of our method over other methods, particularly in challenging scenarios like SSv2 benchmark where the temporal dynamics are critical for action recognition.

2. Related Work

Few-shot Action Recognition. Most existing few-shot action recognition methods adopt the metric-based paradigm to classify videos, and primarily focus on two directions to deal with this task. The first one is to investigate the

spatio-temporal modeling [16, 28, 34, 36, 39, 43, 45, 52]. STRM [28] enriches local patch features and global frame features for joint spatio-temporal modeling. MoLo [34] designs a motion autodecoder to explicitly extract motion dynamics in a unified network. Our method also aims to model spatio-temporal features effectively and efficiently based on adapter-tuning technique. Another direction focuses on designing effective metric learning strategies [3, 12, 20, 23, 33, 38]. OTAM [3] utilizes the DTW [19] algorithm to calculate video distances with strict temporal alignment. TRX [23] exhaustively enumerates all sub-sequences of support and query videos and matches them using attention mechanism. HyRSM [33] applies a novel bidirectional mean Hausdorff metric (referred to as Bi-MHM) to alleviate the strictly ordered constraints. We evaluate our D^2ST -Adapter using the above three classical matching metrics to demonstrate its effectiveness and robustness.

Adapter Tuning. As a classical parameter-efficient fine-tuning method, adapter tuning is first proposed in [11], and quickly draws attention in many other research areas [4, 9, 32]. A common practice is to build up a lightweight module (named *Adapter*) which only consisting of negligible learnable parameters, and selectively plug it into a pre-trained model. During training, only the parameters of inserted adapters are updated while the original model remains frozen, leading to efficient task adaptation. Recently, some works apply this method to adapt image models for action recognition. AIM [44] duplicates multi-head self-attention modules and plugs adapters after them to separately learn the spatial and temporal features in a cascaded manner. Similarly, DUALPATH [22] explicitly builds two-stream architecture upon ViT and utilizes adapters to parallelly learn spatial and temporal features. ST-Adapter [21] employs depth-wise 3D convolution to construct adapter for feature adaptation, which endows it with spatio-temporal modeling capability. Our D^2ST -Adapter follows the basic framework of ST-Adapter, and meanwhile optimize the essential technical designs for few-shot action recognition.

3. Method

3.1. Overview

Problem Formulation of Few-shot Action Recognition.

The task of few-shot action recognition typically follows episodic paradigm [3, 30, 33, 34]. An episode includes a support set \mathcal{S} consisting of N classes and K labeled samples for each class (referred to as the N -way K -shot task), as well as a query set \mathcal{Q} that contains unlabeled samples to be classified. In each episode, we aim to classify every query into one of the N classes with the guidance of the support set. Such episodic task setting is consistently adopted during all the training, validation, and test stages.

Adapter Tuning Framework. We design a plug-and-play and lightweight adapter for video data, dubbed D^2ST -

Adapter, which can be integrated into most existing large image models. Thus we can leverage the powerful feature encoding capability of pre-trained models by efficient image-to-video model adaptation with only a small amount of parameter-tuning overhead. It is particularly well-suited to few-shot scenarios. Figure 2 (a) illustrates the overall adapter tuning framework of our method. Given a pre-trained feature extractor from a large model, our designed lightweight D^2ST -Adapter can be selectively plugged into the middle layers of the feature extractor to perform feature adaptation. Only the inserted adapters are tuned during training while the pre-trained backbone keeps frozen. The learned features by such adapter tuning framework are further used for few-shot action recognition based on the metric-based strategies [3, 23, 33]. We instantiate the backbone of feature extractor with ResNet and ViT respectively.

3.2. D^2ST -Adapter

Dual-pathway Adapter Architecture. Successful action recognition from video data entails effective feature learning of both spatial semantic and temporal dynamic features. Most large vision models are typically pre-trained on image data, thus our D^2ST -Adapter should be capable of capturing both the spatial features and temporal features. To this end, we design the D^2ST -Adapter as a dual-pathway architecture shown in Figure 2 (c), in which the spatial pathway is responsible for capturing the spatial semantics while the temporal pathway focuses on learning the temporal dynamics. As a result, our model is able to encode the spatio-temporal features for video data in a disentangled manner.

As a common practice of typical Adapters [11, 21], our D^2ST -Adapter adopts the bottleneck architecture for reducing the computational complexity. It first downsamples the feature map into a low-dimensional feature space, then the downsampled features are fed into the spatial and temporal pathways concurrently to perform disentangled feature adaptation. Finally, both the adapted spatial and temporal features are fused by simple element-wise addition and up-sampled back to the initial size. Formally, given the feature maps $\mathbf{F}_i^{\text{in}} \in \mathbb{R}^{T \times H \times W \times C}$ obtained from the i -th stage of the pre-trained backbone, containing C channels of feature maps with spatial size $H \times W$ for each of T frames, the feature adaptation by D^2ST -Adapter can be formulated as:

$$\mathbf{F}_i^{\text{out}} = \text{GELU}(\mathcal{F}_S(\mathbf{F}_i^{\text{in}} \cdot \mathbf{W}_{\text{down}}) \oplus \mathcal{F}_T(\mathbf{F}_i^{\text{in}} \cdot \mathbf{W}_{\text{down}})) \cdot \mathbf{W}_{\text{up}}. \quad (1)$$

Herein, $\mathbf{W}_{\text{down}} \in \mathbb{R}^{C \times C'}$ and $\mathbf{W}_{\text{up}} \in \mathbb{R}^{C' \times C}$ are the transformation matrices of two linear layers for downsampling and upsampling, respectively. \mathcal{F}_S and \mathcal{F}_T denote the disentangled feature adaptation by the spatial and temporal pathways respectively. Below we discuss two feasible modeling mechanisms for \mathcal{F}_S and \mathcal{F}_T , which leverages 3D convolution and our proposed aDSTA module, respectively.

Modeling with 3D Convolution. A straightforward way to model the disentangled spatio-temporal feature adaptation

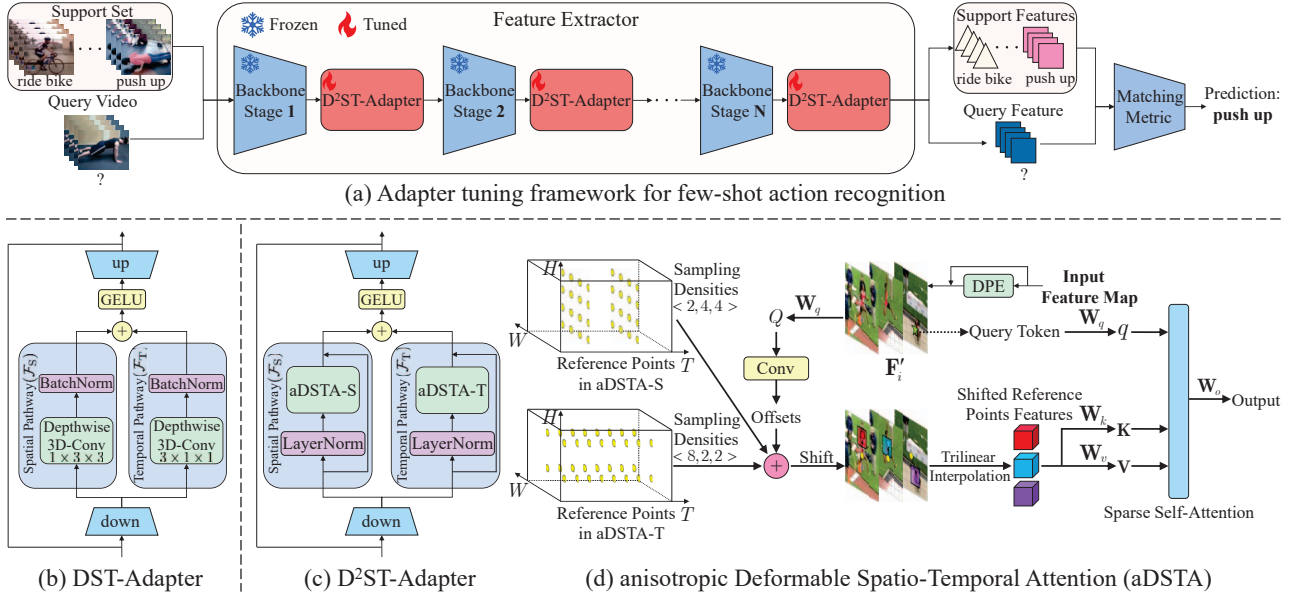


Figure 2. (a) Overall adapter tuning framework of our method. (b) DST-Adapter, a convolutional version of our model, which follows dual-pathway architecture, whereas both pathways are constructed based on 3D convolution. (c) D²ST-Adapter, which is designed in a dual-pathway architecture and both pathways are modeled based on the proposed aDSTA illustrated in (d).

(\mathcal{F}_S and \mathcal{F}_T in Equation 1) is to employ 3D convolutional network, which can be tailored by configuring the shape of convolutional kernels to focus on learning either the spatial or the temporal features. Specifically, we model both \mathcal{F}_S and \mathcal{F}_T using 3D depth-wise convolutional layers followed by a batch normalization layer and a GELU layer, as shown in Figure 2 (b). The only difference between the modeling of them is that \mathcal{F}_S uses $1 \times 3 \times 3$ convolutional kernel for capturing spatial features while \mathcal{F}_T is constructed with $3 \times 1 \times 1$ convolutional layer to learn the temporal dynamics. The resulting version is called *DST-Adapter*.

An important limitation of using 3D convolutions to construct *D²ST-Adapter*, which is also suffered by ST-Adapter, is that the convolutional operation has limited local receptive field either in the spatial or the temporal domain, especially with shallow layers for lightweight design.

Modeling with aDSTA. To address the limitation of the modeling mechanism based on 3D convolution (DST-Adapter), we devise anisotropic Deformable Spatio-Temporal Attention (aDSTA) which performs feature adaptation in a global view by sparse self-attention while maintaining high computational efficiency. It adapts the deformable attention [40] from 2D image space to 3D spatio-temporal space. As shown in Figure 2 (d), aDSTA first samples a group of reference points with anisotropic sampling densities in the spatio-temporal space and then learns the offset for each reference point to shift them to the more informative regions. As a result, aDSTA can learn a set of informative tokens which are further used as key and value pairs shared for all queries for sparse self-attention.

We conduct several adaptations from the deformable attention in 2D space to our aDSTA in spatio-temporal 3D space. First, we employ Dynamic Position Embedding (DPE) [17] implemented as a 3D depth-wise convolution layer to learn the semantic-conditioned spatio-temporal position information for each token and incorporate it into the semantic features by simple element-wise addition. Thus, the feature map $\mathbf{F}'_i \in \mathbb{R}^{T \times H \times W \times C'}$ fused with the position information at the i -th stage is obtained by:

$$\mathbf{F}'_i = \mathcal{F}_{\text{DPE}}(\mathbf{F}_i^{\text{in}} \cdot \mathbf{W}_{\text{down}}) \oplus (\mathbf{F}_i^{\text{in}} \cdot \mathbf{W}_{\text{down}}). \quad (2)$$

Our aDSTA learns the offsets in 3D space for the reference points by a 3D convolutional network consisting of a 3D depth-wise convolution layer and a $1 \times 1 \times 1$ 3D convolution layer as well as a GELU in between. After shifting the reference points according to the learned offsets, we derive the features for each shifted reference point by performing trilinear interpolation among neighboring tokens within a 3D volume space around the point. For instance, the features for the shifted point \mathbf{p} located at (p_t, p_h, p_w) is calculated via the trilinear interpolation $\mathcal{F}_{\text{tri-int}}$ by:

$$\mathcal{F}_{\text{tri-int}}(\mathbf{p}) = \sum_{\mathbf{r}} g(p_t, r_t) \cdot g(p_h, r_h) \cdot g(p_w, r_w) \cdot \mathbf{F}'_i(\mathbf{r}), \quad (3)$$

where $g(a, b) = \max(0, 1 - |a - b|)$ defines an interpolating cubic volume space centered at a and $\mathbf{r} = (r_t, r_h, r_w)$ indexes all tokens in the whole spatio-temporal 3D space. All shifted reference points are used as the keys and values shared for sparse self-attention. For instance, a token in \mathbf{F}'_i located at (u_t, u_h, u_w) serves as a query and the output $\mathbf{Z}_{i,(u_t, u_h, u_w)}$ of aDSTA is calculated by:

$$\mathbf{q} = \mathbf{F}'_{i,(u_t, u_h, u_w)} \cdot \mathbf{W}_q, \mathbf{K} = \mathbf{P} \cdot \mathbf{W}_k, \mathbf{V} = \mathbf{P} \cdot \mathbf{W}_v, \quad (4)$$

$$\mathbf{Z}_{i,(u_t, u_h, u_w)} = \text{softmax}(\mathbf{q}\mathbf{K}/\sqrt{C'})\mathbf{V} \cdot \mathbf{W}_o,$$

where $\mathbf{P} \in \mathbb{R}^{M \times C'}$ is feature matrix of M shifted points and $\mathbf{W}_q, \mathbf{W}_k, \mathbf{W}_v, \mathbf{W}_o$ are projection matrices for the query, key, value and output respectively.

Anisotropic sampling densities. The learned offsets for reference points are restricted in a limited range for two reasons. First, it can prevent the potential training collapse that all reference points are shifted to the same point. Second, such restriction can ensure that each reference point can be shifted to a unique location within a local region. As a result, denser sampling of reference points typically leads to finer-grained representations. The classical deformable attention samples reference points uniformly in 2D feature space with the same sampling densities along two spatial dimensions. In contrast, we configure anisotropic sampling densities in the spatial and temporal domains. Consequently, our $D^2ST\text{-Adapter}$ can leverage aDSTA to model both spatial feature adaptation \mathcal{F}_S in the spatial pathway and temporal feature adaptation \mathcal{F}_T in the temporal pathway.

We denote the sampling densities of reference points of aDSTA along the dimensions of time, height and width as $\langle n_t, n_s, n_s \rangle$, thereby the total number of sampled reference points is $n_t \times n_s \times n_s$. Intuitively, an aDSTA module which samples large n_t and small n_s is more capable of learning the temporal features than learning the spatial features. On the other hand, sampling more reference points in the spatial domain than the temporal domain ($n_s > n_t$) generally makes aDSTA focus on learning the spatial features. Thus, we can tailor aDSTA by configuring the sampling densities to model \mathcal{F}_S and \mathcal{F}_T correspondingly:

- **aDSTA-S** for modeling \mathcal{F}_S in the spatial pathway, which is sampled with larger n_s and smaller n_t ($n_s > n_t$).
- **aDSTA-T** for modeling \mathcal{F}_T in the temporal pathway, which samples denser reference points in temporal domain than in spatial domain ($n_t > n_s$).

The values of n_t and n_s are tuned as hyper-parameters.

3.3. End-to-End Adapter Tuning

The learned features by our adapter tuning framework are further used for few-shot action recognition based on the metric-based strategy. Following ProtoNet [26], it first calculates the frame-wise L_2 distance matrix between the query video and each class prototype derived by averaging the corresponding support samples. Then the distance matrix is used to calculate the matching similarity between the query and each class for prediction. Formally, the similarity between query q and class prototype c can be expressed as:

$$s(\mathbf{F}_q, \mathbf{F}_c) = \mathbf{M}([\mathbf{F}_q^1, \dots, \mathbf{F}_q^{T_q}], [\mathbf{F}_c^1, \dots, \mathbf{F}_c^{T_c}]), \quad (5)$$

where \mathbf{F}_q^i and \mathbf{F}_c^j denote the features of the i -th frame in query q and the j -th frame in class prototype c , and \mathbf{M} is the matching metric such as Bi-MHM [33]. During training,

we take the similarities for each class as logits and optimize the model in an end-to-end manner using the same cross-entropy loss as in [3, 23, 33]. For inference, we classify the query as the support class closest to it.

4. Experiments

4.1. Experimental Setup

Datasets. We conduct experiments on five standard few-shot action recognition benchmarks, including SSv2-Full [8], SSv2-Small [8], Kinetics [14], HMDB51 [15], and UCF101 [27]. Following the typical data split [3, 34, 51], for SSv2-Full, SSv2-Small and Kinetics, we select 64/12/24 classes from the datasets as the training/validation/test set, respectively. As for HMDB51 and UCF101, we adopt the same data split as [34, 45].

Implementation details. To evaluate $D^2ST\text{-Adapter}$, we instantiate our method with both ResNet-50 [10] and CLIP-ViT-B/16 [5]. We insert one $D^2ST\text{-Adapter}$ module in each stage of the pre-trained model (4 stages for ResNet-50 and 12 stages for CLIP-ViT-B/16), as shown in Figure 2 (a). For a fair comparison with previous methods [33, 34], we uniformly sample 8 frames (*i.e.* $T = 8$) as the input of a video. Bi-MHM [33] is used as the matching metric in comparison with state-of-the-art methods. Besides, OTAM [3] and TRX [23] are additionally employed in the ablation study. We report the average classification accuracy of 10,000 episodes randomly selected from the test set. Other implementation details follow HyRSM [33], which are provided in the supplementary material.

4.2. Comparison with State-of-the-Art Methods

We compare our method with other methods on two types of datasets: temporal-related benchmarks including SSv2-Full and SSv2-Small datasets, and spatial-related benchmarks including Kinetics, HMDB51, and UCF101 datasets.

Benchmarks sensitive to temporal features. Table 1 shows the comparative results of different methods on two temporal-related datasets, *i.e.* SSv2-Full and SSv2-Small, which are quite challenging since both the temporal and spatial features are crucial to the action recognition. Our $D^2ST\text{-Adapter}$ achieves the best performance and outperforms other methods by a large margin in most few-shot settings. These results demonstrate the advantages of our method over other methods in dealing with challenging scenarios where the temporal features are critical for action recognition. In particular, our $D^2ST\text{-Adapter}$ substantially outperforms all adapter tuning methods for video data including AIM, DUALPATH, and ST-Adapter, highlighting the advantages of its core design, namely built-in disentangled adaptation of spatial and temporal features.

Benchmarks relying on spatial features. Experimental results on the spatial-related benchmarks including Kinetics, HMDB51, and UCF101, are also presented in Table 1. Our method still performs best in most settings, which mani-

Table 1. Classification accuracy (%) comparison on 5 datasets, under 5-way 1-shot and 5-way 5-shot settings. The highest results are highlighted in **bold** and the second best results are underlined. “PEFT” indicates parameter-efficient fine-tuning. “*” denotes multi-modal methods incorporating additional text input and text encoder.

Method	Pre-training	Fine-tuning	SSv2-Full		SSv2-Small		Kinetics		HMDB51		UCF101	
			1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
CMN [51]	INet-RN50	Full	34.4	43.8	33.4	46.5	57.3	76.0	—	—	—	—
OTAM [3]	INet-RN50	Full	42.8	52.3	36.4	48.0	73.0	85.8	54.5	68.0	79.9	88.9
AMeFu-Net [7]	INet-RN50	Full	—	—	—	—	74.1	86.8	60.2	75.5	85.1	95.5
ITANet [46]	INet-RN50	Full	49.2	62.3	39.8	53.7	73.6	84.3	—	—	—	—
TRX [23]	INet-RN50	Full	42.0	64.6	36.0	59.1	63.6	85.9	53.1	75.6	78.2	96.1
TA ² N [18]	INet-RN50	Full	47.6	61.0	—	—	72.8	85.8	59.7	73.9	81.9	95.1
MTFAN [38]	INet-RN50	Full	45.7	60.4	—	—	74.6	87.4	59.0	74.6	84.8	95.1
STRM [28]	INet-RN50	Full	43.1	68.1	37.1	55.3	62.9	86.7	52.3	77.3	80.5	<u>96.9</u>
HyRSM [33]	INet-RN50	Full	54.3	69.0	40.6	56.1	73.7	86.1	60.3	76.0	83.9	94.7
Nguyen <i>et al.</i> [20]	INet-RN50	Full	43.8	61.1	—	—	74.3	87.4	59.6	76.9	84.9	95.9
Huang <i>et al.</i> [12]	INet-RN50	Full	49.3	66.7	38.9	61.6	73.3	86.4	60.1	77.0	71.4	91.0
HCL [48]	INet-RN50	Full	47.3	64.9	38.7	55.4	73.7	85.8	59.1	76.3	82.5	93.9
SA-CT [47]	INet-RN50	Full	48.9	69.1	—	—	71.9	87.1	60.4	78.3	85.4	96.4
SloshNet [42]	INet-RN50	Full	46.5	68.3	—	—	70.4	87.0	59.4	<u>77.5</u>	86.0	97.1
GgHM [43]	INet-RN50	Full	54.5	69.2	—	—	74.9	87.4	61.2	76.9	85.2	96.3
MoLo [34]	INet-RN50	Full	<u>56.6</u>	70.6	<u>42.7</u>	56.4	74.0	85.6	60.8	77.4	86.0	95.5
Zheng <i>et al.</i> [49]	INet-RN50	Full	55.4	<u>71.4</u>	41.3	57.5	<u>75.6</u>	<u>87.6</u>	<u>61.5</u>	76.0	87.8	96.0
ST-Adapter [21]	INet-RN50	PEFT	52.2	68.7	41.9	55.7	73.0	85.1	60.3	74.7	84.6	94.5
D²ST-Adapter	INet-RN50	PEFT	57.0	73.6	45.8	<u>60.9</u>	75.8	87.7	61.6	76.6	<u>86.9</u>	95.6
AIM [44]	CLIP-ViT-B	PEFT	63.7	79.2	52.8	67.5	88.4	95.3	74.2	86.9	95.4	98.5
DUALPATH [22]	CLIP-ViT-B	PEFT	<u>64.5</u>	79.8	<u>53.5</u>	<u>68.1</u>	88.8	<u>95.4</u>	<u>74.9</u>	<u>87.5</u>	95.7	98.7
ST-Adapter [21]	CLIP-ViT-B	PEFT	64.2	79.5	53.1	68.0	88.5	95.1	74.1	87.3	<u>95.9</u>	<u>98.9</u>
D²ST-Adapter	CLIP-ViT-B	PEFT	66.7	81.9	55.0	69.3	89.3	95.5	77.1	88.2	96.4	99.1
CLIP-FSAR [35]*	CLIP-ViT-B	Full	62.1	72.1	54.6	61.8	94.8	95.4	77.1	87.7	97.0	99.1
EMP-Net [37]*	CLIP-ViT-B	PEFT	63.1	73.0	57.1	65.7	89.1	93.5	76.8	85.8	94.3	98.2
MA-FSAR [41]*	CLIP-ViT-B	PEFT	63.3	72.3	59.1	64.5	<u>95.7</u>	96.0	83.4	87.9	97.2	99.2
Task-Adapter [2]*	CLIP-ViT-B	PEFT	<u>71.3</u>	<u>74.2</u>	<u>60.2</u>	<u>70.2</u>	95.0	<u>96.8</u>	<u>83.6</u>	<u>88.8</u>	<u>98.0</u>	<u>99.4</u>
D²ST-Adapter-MM*	CLIP-ViT-B	PEFT	74.1	83.6	63.6	71.9	96.2	97.2	84.3	89.0	98.3	99.5

Table 2. Comparison with previous adapter-based methods on traditional full-shot action recognition task with CLIP-ViT-B/16 backbone. The testing views (# frames per clip × # temporal clip × # spatial crop) are set as 8×1×3 for K400 and 8×3×1 for SSv2.

Method	K400	SSv2
AIM [44]	83.9	66.4
DUALPATH [22]	84.1	67.3
ST-Adapter [21]	82.0	67.1
D²ST-Adapter (Ours)	84.5	68.0

fest the effectiveness and robustness of our method in these scenarios. Besides, the performance improvement by our *D²ST-Adapter* over other method, especially ST-Adapter, is smaller than that on SSv2 datasets. It is reasonable since the action recognition in these datasets relies less on the temporal features and thus the disentangled feature encoding of our method yields limited performance gain.

Comparison with multimodal methods. To have a fair comparison with the multimodal methods for few-shot action recognition, we extend our model to multi-modality (*D²ST-Adapter-MM*). This extension draws on the multimodal framework of CLIP-FSAR [35] to align with other methods. Table 1 shows that our method outperforms all other multimodal methods, especially on SSv2-Full and

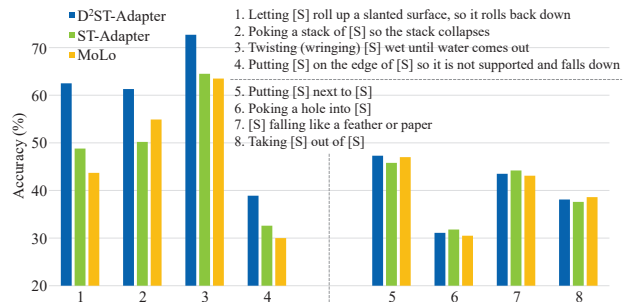


Figure 3. Illustration of actions our method excels at (1-4) and performs poorly at (5-8), respectively. The performance is evaluated on SSv2-Small benchmark in 1-shot setting.

SSv2-Small datasets. These findings consistently highlight the effectiveness and robustness of our *D²ST-Adapter*.

Comparison on traditional (full-shot) action recognition task. Our model is particularly effective in the few-shot scenario while it can also be applied to traditional action recognition tasks using full training samples. To obtain a more comprehensive assessment of our model, we conduct experiments on two traditional action recognition benchmarks, Kinetics-400 (K400) [14] and Something-Something-v2 (SSv2) [8], using the same settings as ST-

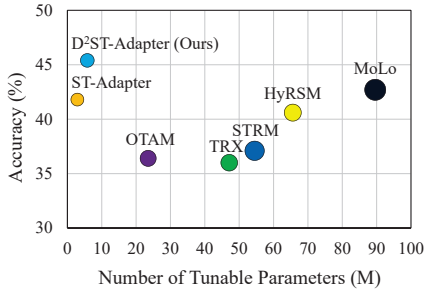


Figure 4. Performance and efficiency of different methods with ResNet-50 backbone on SSv2-Small benchmark in 1-shot setting. Circle size indicates the memory usage.

Table 3. Comparison with other adapter-based methods using CLIP-ViT-B/16 as backbone on efficiency. Accuracy on SSv2-Full benchmark in 1-shot setting is also provided.

Method	Tunable Params	Memory Usage	GFLOPs	Latency	Acc.
AIM [44]	14.3 M	19.0 GB	208	14.1 ms	63.7
DUALPATH [22]	14.3 M	16.7 GB	171	11.6 ms	64.5
ST-Adapter [21]	14.3 M	16.1 GB	163	12.6 ms	64.2
D²ST-Adapter	7.3 M	17.3 GB	150	18.2 ms	66.7

Adapter [21]. Table 2 shows that our method still outperforms these adapter-based methods on both benchmarks.

Case Study of the Pros and Cons of D^2ST -Adapter. To gain more insights into the strengths and weaknesses of our D^2ST -Adapter, we select four categories of actions where our method excels at, as well as four categories where our model struggles in comparison to the state-of-the-art MoLo and ST-Adapter. Figure 3 shows that our model achieves large performance superiority over other methods on the relatively complex actions requiring careful reasoning via learning temporal features for recognition, whilst performing on par with other methods on simple actions that can be recognized primarily based on spatial features. Such results are consistent with the conclusion that our model excels at recognizing actions involving complex temporal dynamics.

Comparison of efficiency. We compare the efficiency between our D^2ST -Adapter and other methods in terms of tunable parameter size and memory usage in Figure 4. It shows that the tunable parameter size of both our model and ST-Adapter is significantly smaller than other methods that are based on full fine-tuning, which reveals the advantage of the adapter tuning paradigm over the full fine-tuning paradigm for task adaptation. Meanwhile, our method outperforms these full fine-tuning based methods substantially with much lower parameter-finetuning overhead, which validates the superiority of our method.

We further compare our model with other adapter-based methods on efficiency in Table 3. Our D^2ST -Adapter performs best while maintaining relatively high efficiency due to the lightweight design of the proposed aDSTA module and the smaller number of inserted adapters (only one per

Table 4. Comparison between different adapters.

Method	SSv2-Full		Kinetics	
	1-shot	5-shot	1-shot	5-shot
Full Fine-tuning	44.9	57.0	72.4	84.3
Vanilla-Adapter	45.3	57.6	72.6	84.8
ST-Adapter	52.2	68.7	73.0	85.1
DST-Adapter (Ours)	53.7	69.6	74.4	86.4
D²ST-Adapter (Ours)	57.0	73.6	75.8	87.7

Table 5. Effect of the anisotropic sampling of aDSTA.

Method	Configurations	SSv2-Full		Kinetics	
		1-shot	5-shot	1-shot	5-shot
Single branch	aDSTA-S	55.2	71.9	74.1	86.0
	aDSTA-T	56.3	72.5	73.8	85.8
	aDSTA-Uniform	55.8	72.2	73.9	86.0
Dual branches	aDSTA-S & aDSTA-T	57.0	73.6	75.8	87.7

block in our method vs. two or four per block in other adapter-based methods [21, 22, 44]).

4.3. Ablation study

We conduct ablation study on SSv2-Full and Kinetics datasets, using ResNet-50 as the pre-trained backbone and Bi-MHM [33] as the matching metric.

Comparison between different Adapters. We first compare the performance of different Adapters in Table 4. Vanilla-Adapter [11] only contains the downsampling and upsampling layers with a GELU nonlinearity in between. As described in Section ‘Method’, DST-Adapter is the convolutional version of our D^2ST -Adapter which models both the spatial and temporal pathways with 3D convolution. The performance of full fine-tuning is also provided for reference. We make following observations.

- Effect of learning temporal features by adapters. ST-Adapter outperforms Vanilla-Adapter distinctly, which indicates that using 3D convolution to learn joint spatio-temporal features facilitates the feature adaptation.
- Effect of disentangled encoding of the spatial and temporal features is demonstrated by the performance improvement from ST-Adapter to DST-Adapter benefited from the dual-pathway adapter architecture.
- Effect of the proposed aDSTA. Comparing our D^2ST -Adapter with DST-Adapter, the proposed aDSTA yields a large performance gain, especially on SSv2-Full which demands effective encoding of temporal features.

Effect of anisotropic sampling of aDSTA. To investigate the effect of proposed anisotropic sampling scheme of aDSTA, we compare the performance of our D^2ST -Adapter using four different configurations of the aDSTA: 1) only ‘aDSTA-S’ modeling the spatial pathway is used; 2) only ‘aDSTA-T’ for the temporal pathway is preserved; 3) ‘aDSTA-Uniform’ performs sampling with uniform density in both the spatial and temporal domains; and 4) ‘aDSTA-S & aDSTA-T’ corresponds the fully functional D^2ST -Adapter. Table 5 shows that the distinct performance gap

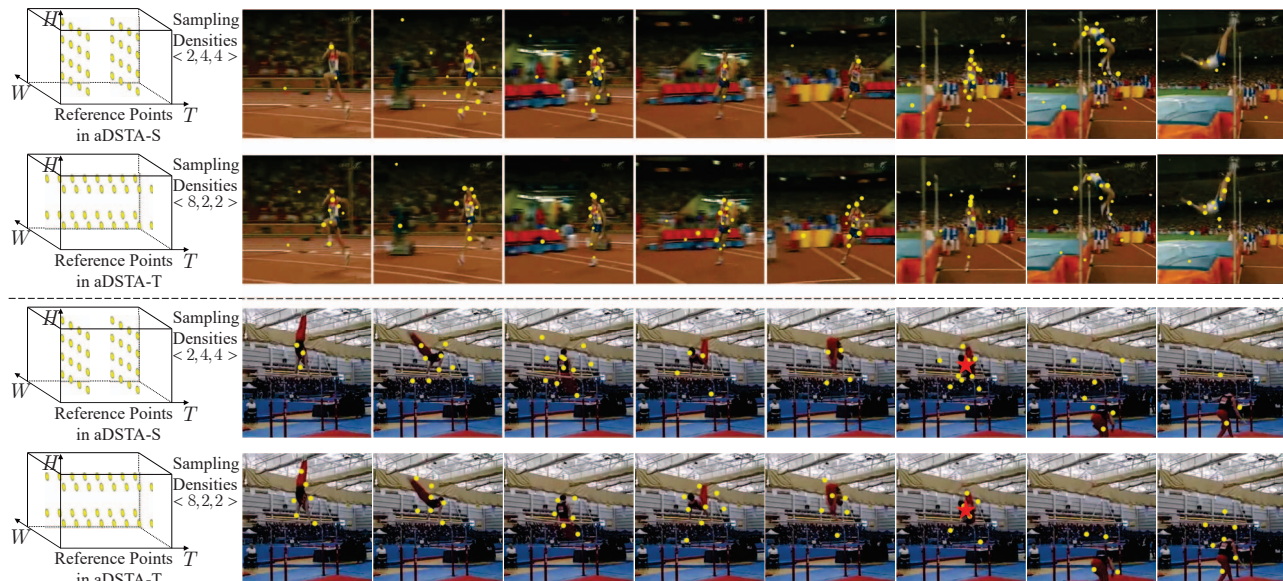


Figure 5. Visualization of important shifted reference points *w.r.t* all queries (first sample), and important shifted reference points *w.r.t* a given query, marked with a red star (second sample). More visualization results are provided in the supplementary material.

Table 6. Evaluation of generalization of our D^2ST -Adapter across different matching metrics on SSv2-Full and Kinetics. The state-of-the-art methods for each metric are involved into comparison.

Matching Metric	Method	SSv2-Full		Kinetics	
		1-shot	5-shot	1-shot	5-shot
OTAM	RFPL	47.0	61.0	74.6	86.8
	MoLo	55.0	69.6	73.8	85.1
	D^2ST-Adapter	56.0	72.8	74.7	86.8
TRX	RFPL	44.6	64.6	66.2	87.3
	MoLo	45.6	66.1	64.8	86.3
	D^2ST-Adapter	47.7	68.6	66.4	87.6
Bi-MHM	HyRSM	54.3	69.0	73.7	86.1
	MoLo	56.6	70.6	74.0	85.6
	D^2ST-Adapter	57.0	73.6	75.8	87.7

between these two settings reveals the advantage of configuring specialized sampling densities for the spatial and temporal pathways.

Generalization Across Different Matching Metrics. To evaluate the generalization of our D^2ST -Adapter across different matching metrics, we conduct the experiments using three classical matching metrics on SSv2-Full and Kinetics, including OTAM [3], TRX [23], and Bi-MHM [33], and compare our model with the state-of-the-art methods corresponding to each metric. Table 6 presents the results, which show that our D^2ST -Adapter consistently outperforms other methods, especially on the challenging SSv2-Full benchmark, validating the well generalization of the proposed D^2ST -Adapter across different matching metrics.

Visualization. We visualize the shifted reference points learned by our D^2ST -Adapter to investigate the proposed aDSTA qualitatively. Following Deformable Attention [40], for each shifted reference point serving as key and value, we

accumulate the attention weights assigned by all queries as its relative importance. Then we visualize top-100 most important shifted reference points in each pathway, with circle size indicating their importance, as shown in the first sample of Figure 5. We also provide visualization of the most relevant reference points *w.r.t* a certain salient region (token) in the second sample of Figure 5. To be specific, we manually select a salient region (indicated by a red star marker) as the query token and visualize top-50 relevant shifted reference points to the query in terms of attention weights.

The results show that the shifted reference points learned by our model can always focus on the salient targets that are critical for action recognition in both the spatial and temporal pathways. Another observation is that the shifted points in the spatial pathway are distributed densely in some specific frames to capture the spatial appearance features for the targets, while the shifted points in the temporal pathway are distributed uniformly among all frames to capture the temporal dynamics. This is consistent with the anisotropic sampling of aDSTA for different pathways.

5. Concluding Remarks

We present D^2ST -Adapter, which is a novel adapter tuning method for few-shot action recognition. It is designed in a dual-pathway architecture, which allows for disentangled encoding of spatial and temporal features. Moreover, we design the aDSTA module, which enables D^2ST -Adapter to encode features in a global view while maintaining lightweight design. Extensive experiments demonstrate the superiority of our model over state-of-the-art methods, especially in challenging scenarios involving complex temporal features.

Acknowledgements. This work was supported in part by the National Natural Science Foundation of China (62372133, 62125201, U24B20174), in part by Shenzhen Fundamental Research Program (Grant NO. JCYJ20220818102415032).

References

- [1] Mina Bishay, Georgios Zoumpourlis, and Ioannis Patras. TARN: Temporal attentive relation network for few-shot and zero-shot action recognition. In *BMVC*, 2019. 1
- [2] Congqi Cao, Yueran Zhang, Yating Yu, Qinyi Lv, Lingtong Min, and Yanning Zhang. Task-adapter: Task-specific adaptation of image models for few-shot action recognition. In *ACMMM*, pages 9038–9047, 2024. 6
- [3] Kaidi Cao, Jingwei Ji, Zhangjie Cao, Chien-Yi Chang, and Juan Carlos Niebles. Few-shot video classification via temporal alignment. In *CVPR*, pages 10618–10627, 2020. 1, 3, 5, 6, 8, 2
- [4] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *NeurIPS*, 35:16664–16678, 2022. 1, 3
- [5] 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. 5
- [6] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. SlowFast networks for video recognition. In *ICCV*, pages 6202–6211, 2019. 2
- [7] Yuqian Fu, Li Zhang, Junke Wang, Yanwei Fu, and Yungang Jiang. Depth guided adaptive meta-fusion network for few-shot video recognition. In *ACMMM*, pages 1142–1151, 2020. 6
- [8] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The” something something” video database for learning and evaluating visual common sense. In *ICCV*, pages 5842–5850, 2017. 5, 6
- [9] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. In *ICLR*, 2021. 1, 3
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 2, 5
- [11] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *ICML*, pages 2790–2799, 2019. 1, 3, 7
- [12] Yifei Huang, Lijin Yang, Yoichi Sato, et al. Compound prototype matching for few-shot action recognition. In *ECCV*, pages 351–368, 2022. 3, 6
- [13] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, pages 709–727, 2022. 1
- [14] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 5, 6
- [15] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. HMDB: a large video database for human motion recognition. In *ICCV*, pages 2556–2563, 2011. 5
- [16] Pulkit Kumar, Namitha Padmanabhan, Luke Luo, Sai Saketh Rambhatla, and Abhinav Shrivastava. Trajectory-aligned space-time tokens for few-shot action recognition. In *ECCV*, 2024. 3
- [17] 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. 4
- [18] Shuyuan Li, Huabin Liu, Rui Qian, Yuxi Li, John See, Mengjuan Fei, Xiaoyuan Yu, and Weiyao Lin. TA2N: Two-stage action alignment network for few-shot action recognition. In *AAAI*, pages 1404–1411, 2022. 6
- [19] Meinard Müller. Dynamic time warping. *Information Retrieval for Music and Motion*, pages 69–84, 2007. 3, 2
- [20] Khoi D Nguyen, Quoc-Huy Tran, Khoi Nguyen, Binh-Son Hua, and Rang Nguyen. Inductive and transductive few-shot video classification via appearance and temporal alignments. In *ECCV*, pages 471–487, 2022. 3, 6
- [21] Junting Pan, Ziyi Lin, Xiatian Zhu, Jing Shao, and Hongsheng Li. ST-Adapter: Parameter-efficient image-to-video transfer learning. In *NeurIPS*, 2022. 1, 3, 6, 7
- [22] Jungin Park, Jiyoung Lee, Kwanghoon Sohn, et al. Dual-path adaptation from image to video transformers. In *CVPR*, pages 2203–2213, 2023. 1, 3, 6, 7
- [23] Toby Perrett, Alessandro Masullo, Tilo Burghardt, Majid Mirmehdi, and Dima Damen. Temporal-relational CrossTransformers for few-shot action recognition. In *CVPR*, pages 475–484, 2021. 1, 3, 5, 6, 8, 2
- [24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 1, 2
- [25] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *NeurIPS*, 27, 2014. 2
- [26] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *NeurIPS*, 30, 2017. 5
- [27] 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. 5
- [28] Anirudh Thatipelli, Sanath Narayan, Salman Khan, Rao Muhammad Anwer, Fahad Shahbaz Khan, and Bernard Ghanem. Spatio-temporal relation modeling for few-shot action recognition. In *CVPR*, pages 19958–19967, 2022. 1, 3, 6

- [29] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015. 1
- [30] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *NeurIPS*, 29, 2016. 3
- [31] 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*, pages 20–36, 2016. 2
- [32] Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuan-Jing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou. K-adapter: Infusing knowledge into pre-trained models with adapters. In *ACL*, pages 1405–1418, 2021. 1, 3
- [33] Xiang Wang, Shiwei Zhang, Zhiwu Qing, Mingqian Tang, Zhengrong Zuo, Changxin Gao, Rong Jin, and Nong Sang. Hybrid relation guided set matching for few-shot action recognition. In *CVPR*, pages 19948–19957, 2022. 1, 3, 5, 6, 7, 8, 2
- [34] Xiang Wang, Shiwei Zhang, Zhiwu Qing, Changxin Gao, Yingya Zhang, Deli Zhao, and Nong Sang. Molo: Motion-augmented long-short contrastive learning for few-shot action recognition. In *CVPR*, pages 18011–18021, 2023. 1, 3, 5, 6
- [35] Xiang Wang, Shiwei Zhang, Jun Cen, Changxin Gao, Yingya Zhang, Deli Zhao, and Nong Sang. Clip-guided prototype modulating for few-shot action recognition. *IJCV*, 132(6): 1899–1912, 2024. 1, 6
- [36] Yuyang Wanyan, Xiaoshan Yang, Chaofan Chen, and Changsheng Xu. Active exploration of multimodal complementarity for few-shot action recognition. In *CVPR*, pages 6492–6502, 2023. 3
- [37] Cong Wu, Xiao-Jun Wu, Linze Li, Tianyang Xu, Zhenhua Feng, and Josef Kittler. Efficient few-shot action recognition via multi-level post-reasoning. In *ECCV*, 2024. 6
- [38] Jiamin Wu, Tianzhu Zhang, Zhe Zhang, Feng Wu, and Yongdong Zhang. Motion-modulated temporal fragment alignment network for few-shot action recognition. In *CVPR*, pages 9151–9160, 2022. 1, 3, 6
- [39] Haifeng Xia, Kai Li, Martin Renqiang Min, and Zhengming Ding. Few-shot video classification via representation fusion and promotion learning. In *ICCV*, pages 19311–19320, 2023. 3
- [40] Zhuofan Xia, Xuran Pan, Shiji Song, Li Erran Li, and Gao Huang. Vision transformer with deformable attention. In *CVPR*, pages 4794–4803, 2022. 2, 4, 8
- [41] Jiazheng Xing, Mengmeng Wang, Xiaojun Hou, Guang Dai, Jingdong Wang, and Yong Liu. Multimodal adaptation of clip for few-shot action recognition. *arXiv preprint arXiv:2308.01532*, 2023. 6
- [42] Jiazheng Xing, Mengmeng Wang, Yong Liu, and Boyu Mu. Revisiting the spatial and temporal modeling for few-shot action recognition. In *AAAI*, pages 3001–3009, 2023. 6
- [43] Jiazheng Xing, Mengmeng Wang, Yudi Ruan, Bofan Chen, Yaowei Guo, Boyu Mu, Guang Dai, Jingdong Wang, and Yong Liu. Boosting few-shot action recognition with graph-guided hybrid matching. In *ICCV*, pages 1740–1750, 2023. 3, 6
- [44] Taojiannan Yang, Yi Zhu, Yusheng Xie, Aston Zhang, Chen Chen, and Mu Li. AIM: Adapting image models for efficient video action recognition. In *ICLR*, 2022. 1, 3, 6, 7
- [45] Hongguang Zhang, Li Zhang, Xiaojuan Qi, Hongdong Li, Philip HS Torr, and Piotr Koniusz. Few-shot action recognition with permutation-invariant attention. In *ECCV*, pages 525–542, 2020. 1, 3, 5
- [46] Songyang Zhang, Jiale Zhou, Xuming He, et al. Learning implicit temporal alignment for few-shot video classification. In *IJCAI*, 2021. 6
- [47] Yilun Zhang, Yuqian Fu, Xingjun Ma, Lizhe Qi, Jingjing Chen, Zuxuan Wu, and Yu-Gang Jiang. On the importance of spatial relations for few-shot action recognition. In *ACMMM*, pages 2243–2251, 2023. 6
- [48] Sipeng Zheng, Shizhe Chen, and Qin Jin. Few-shot action recognition with hierarchical matching and contrastive learning. In *ECCV*, pages 297–313, 2022. 6
- [49] Shuo Zheng, Yuanjie Dang, Peng Chen, Ruohong Huan, Dongdong Zhao, and Ronghua Liang. Saliency-guided fine-grained temporal mask learning for few-shot action recognition. In *ACMMM*, pages 1024–1033, 2024. 6
- [50] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 130(9):2337–2348, 2022. 1
- [51] Linchao Zhu and Yi Yang. Compound memory networks for few-shot video classification. In *ECCV*, pages 751–766, 2018. 5, 6
- [52] Linchao Zhu and Yi Yang. Label independent memory for semi-supervised few-shot video classification. *TPAMI*, 44(1):273–285, 2020. 3