D^{2}ST-Adapter: Disentangled-and-Deformable Spatio-Temporal Adapter for Few-shot Action Recognition

Supplementary Material

A. Effect of the Sampling Densities of *D^{2}ST-Adapter*

We compare the efficiency and performance of different sampling densities in 1-shot setting in Table 7. It demonstrates that sampling denser reference points yields marginal performance gain at the expense of increased memory usage. Our model is set to <2, 4, 4> & <8, 2, 2> to strike a good balance between efficiency and performance.

B. Effect of the Number of Input Frames

To analyze the impact of the number of input frames, we compare the SSv2-Full 1-shot performance when using different number of frames as input under two sampling densities settings. As shown in Table 8, more input frames and denser sampling both yield better performance while incurring more computational cost.

C. Instantiation on More Backbones

To investigate the performance of our *D^{2}ST-Adapter* on smaller backbones, we instantiate our model with ResNet-18 and ResNet-34 pre-trained on ImageNet, and conduct experiments to compare our model with other methods under the same experimental settings. Table 11 shows that our *D^{2}ST-Adapter* achieves the best performance using both backbones in all settings, which demonstrates the robustness of our method across different backbones. While DST-Adapter, the convolutional version of our model, also performs well, it is still inferior to *D^{2}ST-Adapter*, which manifests the effectiveness of the proposed anisotropic Deformable Spatio-Temporal Attention (aDSTA).

For larger backbones, we conduct experiments on CLIP-ViT-L/14, which has a total of 303.2 million parameters, significantly larger than CLIP-ViT-B/16, which has a total of 85.8 million parameters. The results in Table 12 show that our model can still consistently outperform ST-Adapter and Full Fine-tuning on the larger and stronger backbone.

D. Effect of the Inserted Position of *D^{2}ST-Adapter*

Theoretically, our *D^{2}ST-Adapter* can be inserted into any position of the backbone flexibly. To investigate the effect of the inserted position of *D^{2}ST-Adapter* on the performance of the model, we conduct experiments with four different ways of inserting *D^{2}ST-Adapters* into the pre-trained CLIP-ViT-B backbone (comprising 12 learning stages) on SSv2-Small dataset: a) early-insertion, which inserts the *D^{2}ST-Adapter* into each of first 6 stages (close to the input), b) late-insertion that inserts the *D^{2}ST-Adapter* into each of last 6 stages (close to the output), c) skip-insertion, which inserts the adapter into the backbone every two stages and d) full-insertion that inserts the adapter into each learning stage. As shown in Table 9, late-insertion, namely inserting the proposed *D^{2}ST-Adapters* into the last 6 stages,

Table 7. Effect of the sampling densities of *D^{2}ST-Adapter* in CLIP-ViT-B.

Sampling Densities (Spatial & Temporal)	Memory Usage	SSv2-Full	Kinetics
<1, 4, 4> & <4, 2, 2>	17.0 GB	65.9	89.0
<2, 4, 4> & <8, 2, 2>	17.3 GB	66.7	89.3
<2, 6, 6> & <8, 3, 3>	18.0 GB	66.9	89.2
<2, 8, 8> & <8, 4, 4>	18.8 GB	66.9	89.6
<4, 8, 8> & <16, 4, 4>	20.3 GB	67.2	89.3

Table 8. Effect of the number of input frames in SSv2-Full 1-shot setting with CLIP-ViT-B backbone.

Sampling Densities (Spatial & Temporal)	Sampling Frames 4	8	16
<1, 4, 4> & <4, 2, 2>	63.1	65.9	69.0
<2, 4, 4> & <8, 2, 2>	63.6	66.7	69.9

Table 9. Effect of the inserted position of *D^{2}ST-Adapter* in CLIP-ViT-B on SSv2-Small dataset. Skip means adding *D^{2}ST-Adapter* every other stage.

Insertion Position	Tunable Params (%)	Memory Usage	1-shot	5-shot
Early-insertion	4.1%	15.7 GB	48.4	64.6
Late-insertion	4.1%	15.3 GB	54.2	68.5
Skip-insertion	4.1%	15.4 GB	53.3	67.9
Full-insertion	7.9%	17.3 GB	**55.0**	**69.3**

Table 10. Effect of the bottleneck ratio of *D^{2}ST-Adapter* in CLIP-ViT-B on SSv2-Small dataset.

Ratio	Tunable Params (%)	Memory Usage	1-shot	5-shot
0.0625	1.4%	15.5 GB	53.4	68.1
0.125	3.2%	16.2 GB	54.2	68.6
0.25	7.9%	17.3 GB	**55.0**	69.3
0.5	20.2%	19.7 GB	54.7	**69.4**

Table 11. Performance of different methods using smaller backbones (*i.e.*, ResNet-18 and ResNet-34) on SSv2-Full dataset.

Method	ResNet-18					ResNet-34				
	1-shot	2-shot	3-shot	4-shot	5-shot	1-shot	2-shot	3-shot	4-shot	5-shot
OTAM	39.4	45.0	46.6	47.4	49.0	40.6	45.2	48.0	48.9	49.2
TRX	29.9	38.2	44.0	48.2	50.3	32.4	41.6	47.7	52.0	53.5
HyRSM	46.6	54.7	58.7	60.7	61.1	50.0	57.5	61.9	63.3	64.8
MoLo	50.0	57.2	61.6	63.6	64.6	54.1	61.1	65.9	67.3	67.8
ST-Adapter	47.3	54.3	58.1	61.3	62.2	48.9	55.6	59.4	62.5	64.1
DST-Adapter (Ours)	50.0	56.9	61.7	63.4	65.3	52.1	59.3	63.4	65.8	67.5
D^{2}ST-Adapter (Ours)	**53.0**	**60.4**	**65.0**	**67.1**	**68.6**	**54.4**	**62.5**	**66.0**	**69.1**	**70.6**

Table 12. Performance of different methods using a larger backbone (CLIP-ViT-L/14). The number of backbone parameters (Params) is also provided as reference.

Method	Backbone (Params)	SSv2-Full		Kinetics	
		1-shot	5-shot	1-shot	5-shot
Full Fine-tuning	CLIP-ViT-L/14 (303.2 M)	55.5	73.0	91.7	96.8
ST-Adapter	CLIP-ViT-L/14 (303.2 M)	67.0	83.8	92.2	96.9
D^{2}ST-Adapter	CLIP-ViT-L/14 (303.2 M)	**69.6**	**85.6**	**92.8**	**97.2**

yields better performance than early-insertion, which implies that adapter tuning is more effective for task adaptation in deeper layers than in the shallower layers. It is reasonable since deeper layers generally capture the high-level semantic features, which are more relevant to task adaptation. Besides, full-insertion achieves the best performance at the expense of slightly more tunable parameters and memory usage.

E. Effect of the Bottleneck Ratio of *D^{2}ST-Adapter*

The tunable parameter size is mainly determined by the bottleneck ratio of *D^{2}ST-Adapter*, defined as the ratio of downsampled channel numbers to the initial size. Thus, we can balance between the model efficiency in terms of tunable parameter size and model effectiveness in terms of classification accuracy by tuning the bottleneck ratio. As shown in Table 10, larger bottleneck ratios typically yield more performance improvement while introducing more tunable parameters, and we set the bottleneck ratio to 0.25 in all the experiments based on the results.

F. More Visualizations

Consistent with Figure 5 in the paper, we provide more visualization of the shifted reference points in Figures 6 and 7 to evaluate the anisotropic Deformable Spatio-Temporal Attention (aDSTA). Note that only the top-100/50 most important points from the aDSTA modules across all inserted D^{2}ST-Adapters are visualized. The results show that our model is able to capture the salient objects through the shifted reference points in both spatial and temporal domains.

G. More Implementation Details

Matching Metric. In the few-shot action recognition task, the classification of a query sample is based on the similarities between it and each support class prototype. Typically, a matching metric is first used to temporally align the frames or segments within two videos. Then, the overall similarity between the query and each prototype can be calculated by averaging or summing the similarities of all aligned pairs. We adopt three classic matching metrics in our experiments, including OTAM [3], TRX [23], and Bi-MHM [33]. Details of these matching metrics are provided as follows.

OTAM [3] explicitly leverages the temporal ordering information in videos through tight temporal alignment, which ensures that the alignment results for any two frame pairs do not overlap. It extends the Dynamic Time Warping (DTW) algorithm [19] to compute the alignment path of two videos in the temporal dimension, and employs continuous relaxation to make the model differentiable.

TRX [23] aligns segments rather than individual frames, thus effectively matching actions performed at varying speeds and in different locations across two videos. It first exhaustively enumerates all subsequences of two to four frames as potential actions in both videos. Then, for each action in the query video, the cross-attention mechanism is employed to compute the corresponding action prototype in the support video based on feature similarities, serving as its match.

Bi-MHM [33] stands for Bidirectional Mean Hausdorff Metric, which treats the similarity measurement between two videos as a set matching problem. This alignment strategy eliminates the constraints of temporal order and focuses

Table 13. Hyper-parameter settings of our method with ResNet-50 backbone.

Backbone	Hyper-parameter	SSv2-Full	SSv2-Small	Kinetics	HMDB51	UCF101
ResNet-50	base learning rate	2e-3	2e-3	2e-3	2e-3	2e-3
	warmup start learning rate	4e-4	4e-4	4e-4	4e-4	4e-4
	training episodes	120000	40000	10000	8000	8000
	warmup episodes	12000	4000	1000	800	800
	weight decay	5e-4	5e-4	5e-4	5e-4	5e-4
	batch size	8	8	8	8	8
	num. adapters per block	1	1	1	1	1
	num. input frames	8	8	8	8	8
	bottleneck ratio	0.25	0.25	0.25	0.25	0.25
	training crop size	224	224	224	224	224
CLIP-ViT-B	base learning rate	1e-3	1e-3	5e-4	1e-3	5e-4
	warmup start learning rate	2e-4	2e-4	1e-4	2e-4	1e-4
	training episodes	120000	40000	3000	3000	2400
	warmup episodes	12000	4000	300	300	240
	weight decay	5e-4	5e-4	5e-4	5e-4	5e-4
	batch size	8	8	8	8	8
	num. adapters per block	1	1	1	1	1
	num. input frames	8	8	8	8	8
	bottleneck ratio	0.25	0.25	0.25	0.25	0.25
	training crop size	224	224	224	224	224

Table 14. Tuned sampling densities (in the form of <T, H, W>) for aDSTA-S in the spatial pathway and aDSTA-T in the temporal pathway of our *D*^{2}*ST-Adapter*. Besides, the sampling densities for aDSTA-Uniform used in Table 5 for ablation study is also provided.

Backbone	Feature Map	Sampling Densities in aDSTA-S	Sampling Densities in aDSTA-T	Sampling Densities in aDSTA-Uniform
ResNet-50	<8, 56, 56>	<2, 8, 8>	<8, 4, 4>	<4, 4, 4>
ResNet-50	<8, 28, 28>	<2, 4, 4>	<8, 2, 2>	<4, 4, 4>
ResNet-50	<8, 14, 14>	<2, 4, 4>	<8, 2, 2>	<4, 4, 4>
ResNet-50	<8, 7, 7>	<2, 2, 2>	<8, 1, 1>	<4, 4, 4>
CLIP-ViT-B	<8, 14, 14>	<2, 4, 4>	<8, 2, 2>	<4, 4, 4>

solely on the appearance similarities between frames.

Data Pre-processing. We use the same data pre-processing methods as in HyRSM [33]. A video is uniformly sampled 8 frames as input. During training, each frame is first resized to 256×256 and then randomly cropped to 224×224. Some basic data augmentation methods are adopted, such as color jitter and horizontal flip. Note that for temporal-related datasets, *i.e*. SSv2-Full and SSv2-Small, we do not use horizontal flip, since the recognition of some classes in these two datasets requires distinguishing between left and right, *e.g*. "pulling something from left to right".

Tuned hyper-parameters. For ease of the reproduction, we list the tuned values for all the hyper-parameters in our experiments for each dataset with both backbones in Table 13.

Tuned sampling densities of aDSTA. The sampling densities of aDSTA-S in the spatial pathway and aDSTA-T in the temporal pathway of our *D*^{2}*ST-Adapter* can be tuned on a held-out validation set. Generally, aDSTA-S should sample denser reference points in the spatial domain while aDSTA-T samples more points in the temporal domain. For the instantiation of our model with ResNet, we configure the sampling densities of aDSTA in each convolutional stage individually since the feature maps in different convolutional stages may have different size. In contrast, we only need to tune one configuration for sampling densities when using ViT as the backbone since the feature map always has fixed size in different stages. Table 14 shows the tuned configurations of the sampling densities of aDSTA-S and aDSTA-T with different backbones. Besides, we also provide the configurations of sampling densities for aDSTA-Uniform module which is constructed to validate the effect of configuring the sampling densities of aDSTA in Table 5.

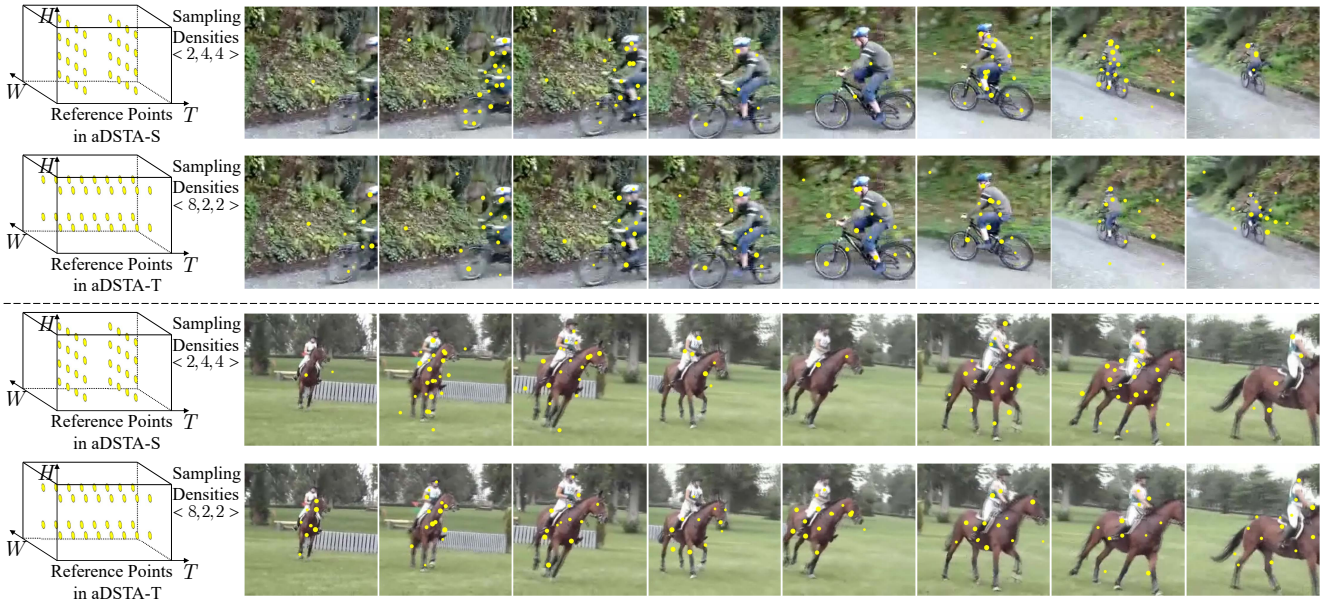

Figure 6. Visualization of top-100 important shifted reference points in both pathways for two video samples. Circle size indicates the importance for each point calculated by aggregating the attention weights from all queries.

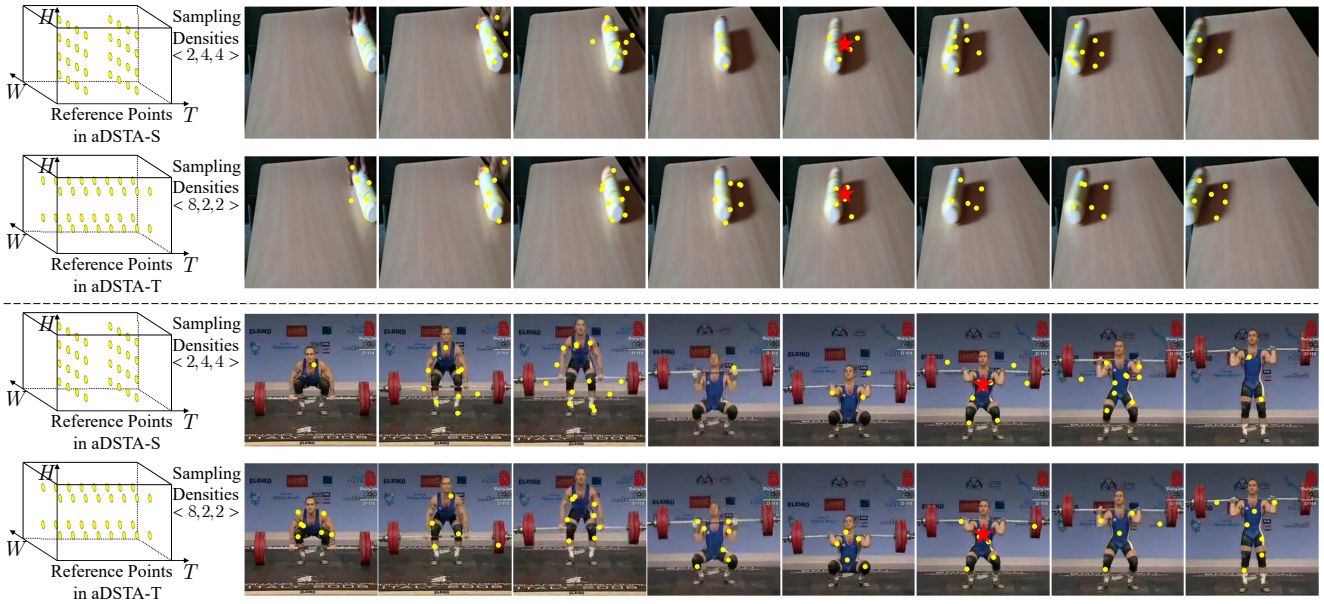

Figure 7. Given a selected query within the salient object indicated by a red star marker, top-50 relevant shifted reference points in terms of attention weight are visualized.