Appendix for Few-shot Action Recognition with Hierarchical Matching and Contrastive Learning

Anonymous ECCV submission

Paper ID 4804

This document provides supplementary material that has been omitted from the main paper due to space limitations. Section. 1 provides more implementation details about our model. Section. 2 presents the performance comparison under different setups from 1-shot to 5-shot and the computational cost comparison between previous works and ours. Section. 3 shows the additional ablation of different clip and patch number we select in temporal and spatial matching. Section. 4 provides the detailed description of the multi-scale strategy to represent patches and clips.

1 Additional Implementation Details

During training, we normalize the video frames before feeding them into the model. During inference, we simply resize the frame scale into 224×224 without augmentation. We adopt 5-way K-shot setup to evaluate our model and each episode contains 5 query videos for each class. We use a simple yet effective multi-scale strategy to enhance temporal and spatial representations with multiple clip scales $r_t \in \{2, 3\}$ and patch scales $r_s \in \{1, 2, 4\}$ respectively, which will be described in Section. 4 in detail. More details can be found in our codes.

2 Additional Comparison

Experimental Performance. Due to the space limitation, we only demonstrate partial comparison results on five dataset splits including Kinetics [10], SSv2† [2], SSv2* [11], HMDB-51 [9] and UCF-101 [9] in the main paper. For a comprehensive comparison with previous state-of-the-art works, we provide more comparisons with existing few-shot works under the setups from 1-shot to 5-shot, which are presented in Tab. 1. The global-matching and temporal-matching approaches are presented in the first and second block of the tables. We achieve state-of-the-art results on all metrics under all setups on Kinetics, SSv2* and HMDB-51 datasets. We achieve superior results on SSv2 † under 1-shot, 2-shot and 3-shot setups, and best 1-shot result on UCF-101.

Computational Cost. Since our method consists of matching at three levels, the computational cost is larger than previous works that only use matching at the global or temporal level. Specifically, HCL takes around 800ms to evaluate the distance of each video pair, where the global, temporal and spatial matching takes around 100ms, 200ms and 500ms respectively. The computation cost at global and temporal matching is similar to previous works, e.g., TRX [6] (temporal matching) takes around 200ms.

Table 1: Performance compared with other works on different datasets.

Dataset	Match	Method	1-shot	2-shot	3-shot	4-shot	5-shot
Kinetics	Global	Matching Net [8]	53.3	64.3	69.2	71.8	74.6
		MAML [3]	54.2	65.5	70.0	72.1	75.3
		ProtoNet [7]	59.1	73.6	78.7	81.7	83.5
		TRAN [1]	66.6	74.6	77.3	78.9	80.7
	Temporal	CMN [10]	60.5	70	75.6	77.3	78.9
		TAM [2]	73.0	-	-	-	85.8
		TRX [6]	64.6	76.4	80.5	83.6	85.5
	Hierarchical	ours	**73.7**	**79.1**	**82.4**	**84.0**	**85.8**
$SSv2^*$	Global	MAML [3]	30.9	35.1	-	-	41.9
		ProtoNet [7]	34.0	41.2	45.5	48.9	51.7
		TRAN [1]	66.6	74.6	77.3	78.9	80.7
	Temporal	TSN++ [2]	33.6	-	-	-	43.0
		CMN++ [2]	34.4	-	-	-	43.8
		TRN++ [2]	38.6	-	-	-	48.9
		TAM [2]	42.8	-	-	-	52.3
		TRX [6]	38.1	49.1	55.7	60.4	63.9
	Hierarchical	ours	**47.3**	**54.5**	**59.0**	**62.4**	**64.9**
$SSv2^{\dagger}$	Global	Matching Net [8]	31.3	35.9	39.8	40.5	45.5
		MAML [3]	30.9	35.1	38.6	40	41.9
		ProtoNet [7]	30.9	37.2	41.8	44.5	47.2
	Temporal	CMN [10]	36.2	42.1	44.6	47	48.8
		TRX [6]	34.7	43.5	49.0	**52.9**	**56.8**
	Hierarchical	ours	**38.7**	**45.5**	**49.1**	51.8	55.4
HMDB-51	Global	GenAPP [5]	-	-	-	-	52.5
		ProtoGAN [4]	34.7	-	-	-	54.0
		ProtoNet [7]	44.2	57.3	64.6	68.2	72.0
	Temporal	ARN [1]	45.5	-	-	-	60.6
		TRX [6]	52.0	64.2	70.6	73.3	75.6
	Hierarchical	ours	**59.1**	**66.5**	**71.2**	**73.8**	**76.3**
UCFUCF-101	Global	GenAPP [5]	-	-	-	-	78.6
		ProtoGAN [4]	57.8	-	-	-	80.2
		ProtoNet [7]	67.2	82.4	88.1	90.9	93.0
	Temporal	ARN [9]	66.3	-	-	-	84.8
		TRX [6]	81.3	**90.2**	**93.1**	**94.9**	**95.9**
	Hierarchical	ours	**82.5**	88.6	91.0	92.4	93.9

3 Additional Ablation of Clip and Patch Number

We empirically set the number of clips T and patches S selected in temporal and spatial matching as 10 according to the default setting. Additionally, we provide ablations of these two hyper-parameters in Tab. 2. As spatial matching contributes more for Kinetics while temporal matching matters for SSV2*, we evaluate S on Kinetics and T on SSV2*.

Table 2: Ablation experiments with different number of clips T and patches S.

S	Kinetics			T	SSV2*		
	1-shot	2-shot	5-shot		1-shot	2-shot	5-shot
5	71.8	77.8	84.9	5	45.9	53.6	63.3
10	**73.7**	**79.1**	**85.8**	**10**	**47.3**	**54.5**	**64.9**
15	74.6	78.6	85.1	20	46.1	53.8	64.1

Table 3: Ablation experiments with the multi-scale patches r_s and clips r_t.

			Kinetics		SSv2*	
		multi-scale	1-shot	5-shot	1-shot	5-shot
1	clips (r_t)	$\{2\}$	72.9	85.5	45.7	63.1
2		$\{3\}$	72.8	85.3	46.6	64.2
3		$\{2,3\}$	73.7	85.8	47.3	64.9
4	patches (r_s)	$\{1\}$	72.1	84.5	46.5	64.2
5		$\{1,2\}$	73.3	85.6	46.7	64.7
6		$\{1,2,4\}$	73.7	85.8	47.3	64.9

4 Additional Ablation of Multi-scale Representation

In order to better capture diverse sizes of objects and various durations of actions, we further exploit a multi-scale strategy to enhance both spatial and temporal representations. Given a video with t frames, we extract $t \times h \times w$ patches from it. In the spatial dimension, besides the original $h \times w$ patches, we use average pooling to get downsampled feature maps, e.g., $\frac{h}{r_s} \times \frac{w}{r_s}$, where r_s denotes a downsample rate. In the temporal dimension, we exhaustively enumerate all possible clips from the video with r_t frames to construct the clip set C.

The additional experiments when using different scales of clips and patches are presented in Tab. 3. The results show that multi-scale clips r_t=$\{2,3\}$ bring stable improvements compared with single scale r_t=$\{2\}$ or r_t=$\{3\}$, which improves the 1-shot performance on Kinetics from 72.9% to 73.7% and SSv2* from

63.1% to 64.9%. Similar improvement of multi-scale patches can also be observed in row 4-6 of Tab. 3 and our model achieves the best performance using r_s=$\{1, 2, 4\}$. The multi-scale clips and patches enable our model to better adapt to different durations of actions and sizes of objects.

References

1. Bishay, M., Zoumpourlis, G., Patras, I.: Tarn: Temporal attentive relation network for few-shot and zero-shot action recognition. BMVC (2019) 2
2. Cao, K., Ji, J., Cao, Z., Chang, C.Y., Niebles, J.C.: Few-shot video classification via temporal alignment. CVPR (2020) 1, 2
3. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. ICML (2017) 2
4. Kumar Dwivedi, S., Gupta, V., Mitra, R., Ahmed, S., Jain, A.: Protogan: Towards few shot learning for action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. pp. 0–0 (2019) 2
5. Mishra, A., Verma, V.K., Reddy, M.S.K., Arulkumar, S., Rai, P., Mittal, A.: A generative approach to zero-shot and few-shot action recognition. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 372–380. IEEE (2018) 2
6. Perrett, T., Masullo, A., Burghardt, T., Mirmehdi, M., Damen, D.: Temporal-relational crosstransformers for few-shot action recognition. CVPR (2021) 1, 2
7. Snell, J., Swersky, K., Zemel, R.S.: Prototypical networks for few-shot learning. NeurIPS (2017) 2
8. Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., Wierstra, D.: Matching networks for one shot learning. NeurIPS (2016) 2
9. Zhang, H., Zhang, L., Qi, X., Li, H., Torr, P.H., Koniusz, P.: Few-shot action recognition with permutation-invariant attention. ECCV (2020) 1, 2
10. Zhu, L., Yang, Y.: Compound memory networks for few-shot video classification. ECCV (2018) 1, 2
11. Zhu, L., Yang, Y.: Label independent memory for semi-supervised few-shot video classification. IEEE Annals of the History of Computing (2020) 1