Supplementary: Unsupervised Few-Shot Action Recognition via Action-Appearance Aligned Meta-Adaptation

Jay Patravali[⋆‡] Gaurav Mittal[⋆†] Ye Yu[†] Fuxin Li[‡] Mei Chen[†]
[†]Microsoft [‡]Oregon State University
{gaurav.mittal, yu.ye, mei.chen}@microsoft.com {patravaj,lif}@oregonstate.edu

In this document, we provide additional analysis of our method MetaUVFS both qualitatively and quantitatively. We include details and analysis that was ready at the time of submission but could not be included in the paper due to space constraints. We also provide a list of classes that were explicitly removed from the large-scale unlabeled video dataset used for training our model.

1. Additional Implementation Details

We use Pytorch (v1.7) as the deep learning framework. We make use of the Pytorch Distributed Training package, `torch.distributed`, for our distributed training jobs that uses NCCL v2.7.8 as the communications library. For large-scale dataset training, our model takes close to an hour per epoch when run on 64 GPUs in parallel in a distributed setting. We set the support and query samples to be 1 for the few-shot meta-training phase. We use learn2learn [1] library to conduct our few-shot experiments with MAML [3]. We use second-order gradients for our MAML training (ablation shown later for using first-order approximation of MAML).

2. Action-Appearance Alignment Visualization: Novel Classes

To qualitatively evaluate the effectiveness of the Action-Appearance Aligned Meta-adaptation (A3M) module of MetaUVFS, we sample a few support-query pairs of novel class videos for Kinetics100, UCF101 and HMDB51 dataset. We visualize the action-appearance aligned masked frames of the video pairs based on the input frames fed to the 3D action and 2D appearance encoders. The action-appearance aligned masked frames refer to the frames fed to the 2D appearance encoder that have been overlayed with the weighted attention scores as a mask computed by the A3M module based on the action and appearance encoder embeddings. Figure 1 illustrates the videos and their corresponding action-appearance aligned masked frames. The support video samples are on the left and query video samples are on the right. For each video sample, there are three rows of frames. The top row provides the 4×4 sampled frames for the 3D action encoder at 112×112 resolution, referred to as Action Input (Fig. 6). The middle row provides the 1×8 sampled frames for the 2D appearance encoder at 224×224 resolution, referred to as Appearance Input (Fig. 6). The bottom row provides the Action-Appearance Aligned Masked Frames as described above. The weighted frames embeddings from the A3M module are then aggregated to generate the final action-appearance aligned video embedding for few-shot video action recognition.

From each video sample shown in Figure 1, we can observe that the attention mask from A3M is able to effectively focus on the frames that are most representative of the correct action in relation to the spatial 2D appearance embeddings and spatio-temporal 3D action embeddings. For instance, for the second video (UCF101, Still Rings), the A3M module learns to focus on the frames that show the characteristic pose of a person as part of Still Rings. There are frames where the person is standing with normal stance and can be ambiguous in deducing the correct action. The A3M module learns to mask these ambiguous frames so that it can learn to correctly predict the action using as few as 1 support example. Similarly, for the fourth video (HMDB51, Kick), the A3M module learns to focus on the frames demonstrating the 'Kick' action and masks the rest of the frames where the person might be just standing in a certain pose. Another key observation is that the frame may not be contiguous and A3M can decide to focus on any arbitrary set of frames based on the most representative action-oriented frames (*e.g.*, query video for HMDB51, Kick and query video for Kinetics100, Cutting Watermelon).

3. Nearest Neighbor Retrieval: Novel Classes

As illustrated in Figure 2, we perform a qualitative analysis of our approach MetaUVFS by retrieving and visualizing the top 1, 3, and 5 nearest neighbors (right side) for a

⋆ Authors with equal contribution.
Work done when Jay Patravali was a research intern at Microsoft.

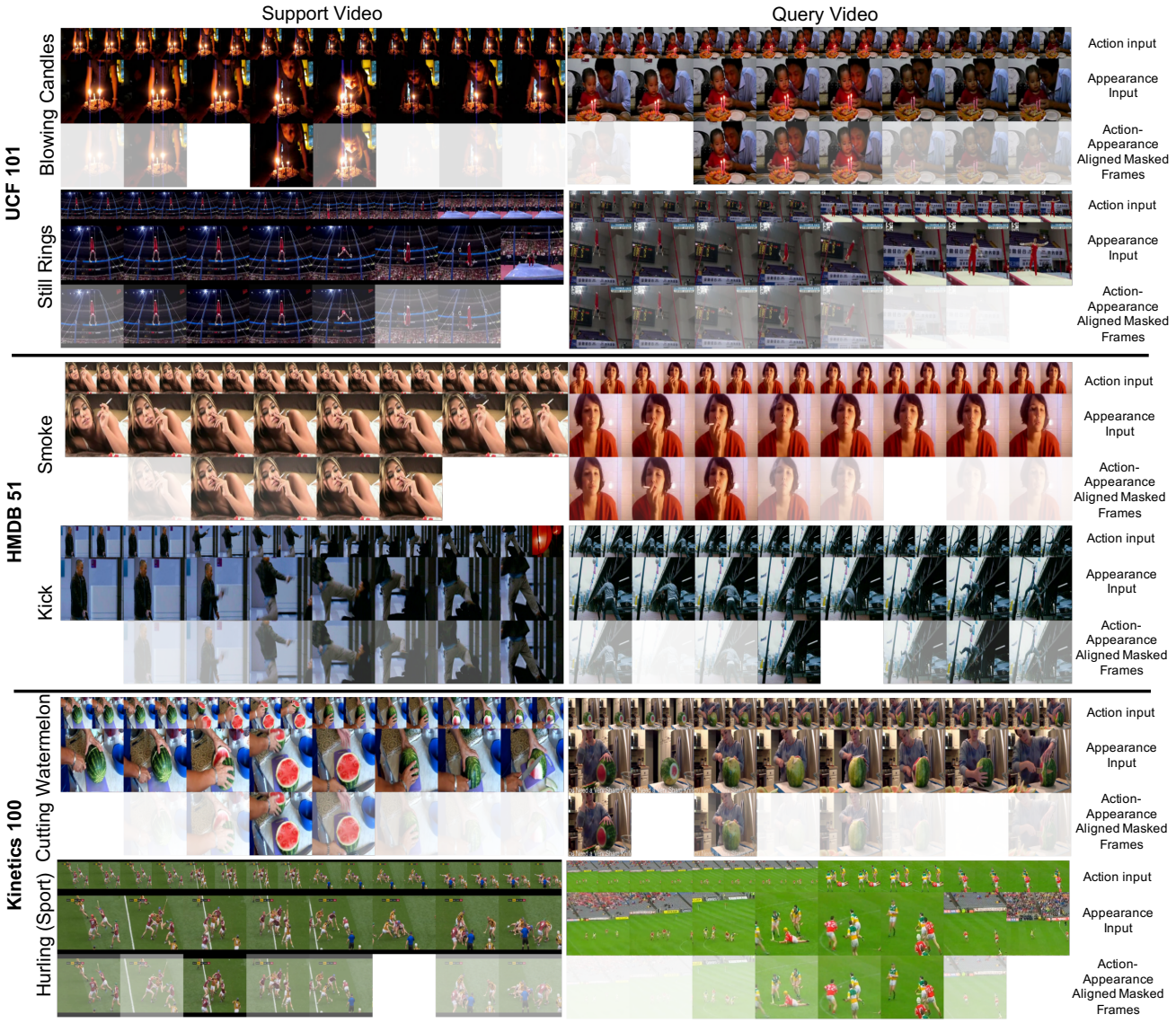

Figure 1. Support-Query novel class videos highlighting the Action-Appearance Aligned Masked Frames. For each video, support sample is on the Left and query sample is on the Right. For each video, there are three rows: $4 \times 4, 112 \times 112$ frames as Action Input fed to 3D action encoder (top), $1 \times 8, 224 \times 224$ frames as Appearance Input fed to 2D appearance encoder (middle), and Action-Appearance Aligned Masked Frames (bottom) where the Appearance Input is overlayed with the weighted attention mask from Action-Appearance Aligned Meta-adaptation (A3M) module of MetaUVFS based on Action and Appearance Input. We can observe that A3M module learns to effectively focus on the frames most representative of the action in as few as 1 support sample. For *e.g.*, in the first video (UCF101, Blowing Candles), A3M module effectively puts high attention on the frames illustrating the person blowing the candle and masking out the frames where the person is just standing in the anticipation of blowing the candles. Similarly, in the third video (HMDB51, Smoke), the A3M module learns to focus on the frames where the person is taking a drag on the cigarette and masks out the other frames.

query video (left side) belonging to the novel classes. The nearest neighbors are obtained by comparing pair-wise cosine distances of the novel class action-appearance aligned video feature embeddings. We can observe that for UCF101 and Kinetics100 dataset, our model is able to accurately retrieve the videos of the same class as the query video. For the novel classes in the HMDB51 dataset, many of the videos belonging to a class actually belong to the same video source. Hence, in our illustration, the top-1 and top-3 retrieved videos are almost identical. Additionally, a failure case, highlighted in a red dashed-line box, occurs because the query video of class 'Kick' and the falsely-retrieved

video of class 'Run' belong to the same video source 'The Matrix' (potentially a bias towards the same video source).

4. Further Ablation

4.1. Effectiveness of large-scale unlabeled data

To demonstrate the effectiveness of performing unsupervised training on a large-scale unlabeled video dataset, we conduct a set of experiments where we train our method MetaUVFS only on the base class dataset splits of the three few-shot benchmark datasets – UCF101, HMDB51, and Kinetics100. In these experiments, we consider the base class videos to be unlabeled without ground truth annotations.

Tabel 1 summarizes the results of this study. For comparison, we also show the few-shot performance on CVRL [5] which is the best-performing baseline for unsupervised video representation learning. From the table, we can observe that the performance of our approach MetaUVFS, meta-trained on a large-scale unlabeled video dataset, is higher by at least 10% on 1-shot experiments and by at least 7% on 5-shot experiments compared to the performance when trained on individual base class splits. A similar trend can also be observed for CVRL which is the best performing unsupervised baseline. This large performance improvement can be attributed to the fact that unsupervised methods tend to learn better feature representations when subjected to large amounts of data. Another observation is that in both scenarios of training with either large-scale data or individual base class splits, our method MetaUVFS is able to significantly outperform CVRL across all few-shot experiments. This further highlights the importance of aligning action and appearance-based representations and having a specialized few-shot meta-learner, that induces a downstream few-shot task-specific prior on the model, to enhance few-shot performance.

4.2. Different backbones for 3D Action Encoder

To validate our choice of using ResNet50-3D in MetaUVFS, we train our method using different 3D CNN architectures (C3D [6], S3D [8] and R(2+1)D [7]) as the 3D action encoder in our model. Table 2 shows the comparison of the few-shot performance across the different 3D action encoders. We can observe from the table that across all few-shot settings, ResNet50-3D is able to perform the best among all the 3D CNN architectures considered. This justifies our choice of using ResNet50-3D in MetaUVFS. For R(2+1)D, we choose a 34-layer architecture that has larger number of parameters than our chosen ResNet50-3D architecture. On the other hand, C3D and S3D have relatively fewer number of parameters compared to ResNet50-3D. Since the few-shot performance is not correlated to the number of parameters in the 3D encoder, we believe that the higher performance of ResNet50-3D compared to others is due to the inherent network design and inductive bias of ResNet50-3D that enables it to learn more generalizable features over the given data distribution.

4.3. Learning Rate Analysis

As our few-shot meta-testing involves finetuning following MAML [3] protocol, we observe that the learning rate and the number of steps to train the classifier on support samples are fundamental to achieving the optimal performance. Figure 3 provides insights into the effect of different learning rates and the number of steps to fine-tune have on the few-shot performance. Note that since we use l_2-normalized representation as the input to the classifier, higher learning rates such as 1 or 10 can be used for faster convergence. However, very high learning rates such as 100 can lead to overfitting and reduced performance.

4.4. Comparison with using First-Order approximation of MAML (FOMAML)

We also conduct an experiment where we meta-train our method using First-Order approximation of MAML (also known as FOMAML [3]). On evaluating on Kinetics100 few-shot benchmark, we obtain 62.05 ± 0.46 and 78.61 ± 0.40 on 5-way, 1-shot and 5-way, 5-shot few-shot settings respectively. This is lower than using MAML with second-order gradients that achieves 62.80 ± 0.45 and 79.55 ± 0.39 on 5-way, 1-shot and 5-shot few-shot settings for Kinetics100 respectively. We conclude that, empirically, using MAML with second-order gradients helps to achieve the best performance compared to FOMAML and the rest of the baseline meta-learning algorithms (Table 4, main paper).

5. Spatio-temporal frame sampling and augmentation

Figure 6 illustrates the spatio-temporal frame sampling strategy we employ in MetaUVFS to maximize the downstream few-shot performance. For an input video, the 2D appearance stream encodes 8 input frames where 1 frame is randomly sampled from each of 8 segments equally-partitioned along the video length at 224×224 resolution (8×1). For the 3D-action stream, we randomly sample 4 clips across 4 equidistant segments of the video to form a 16 frame input at a lower frame resolution of 112×112 (4×4).

6. T-SNE Visualization of Few-Shot Tasks

We visualize the action-appearance aligned feature representations of MetaUVFS on the 5-way and 10-way setting using T-SNE [4] as shown in Figure 4. For 5-way, we randomly sample 5 classes and encode all videos belonging to these 5 classes and then obtain the T-SNE plots. Similarly,

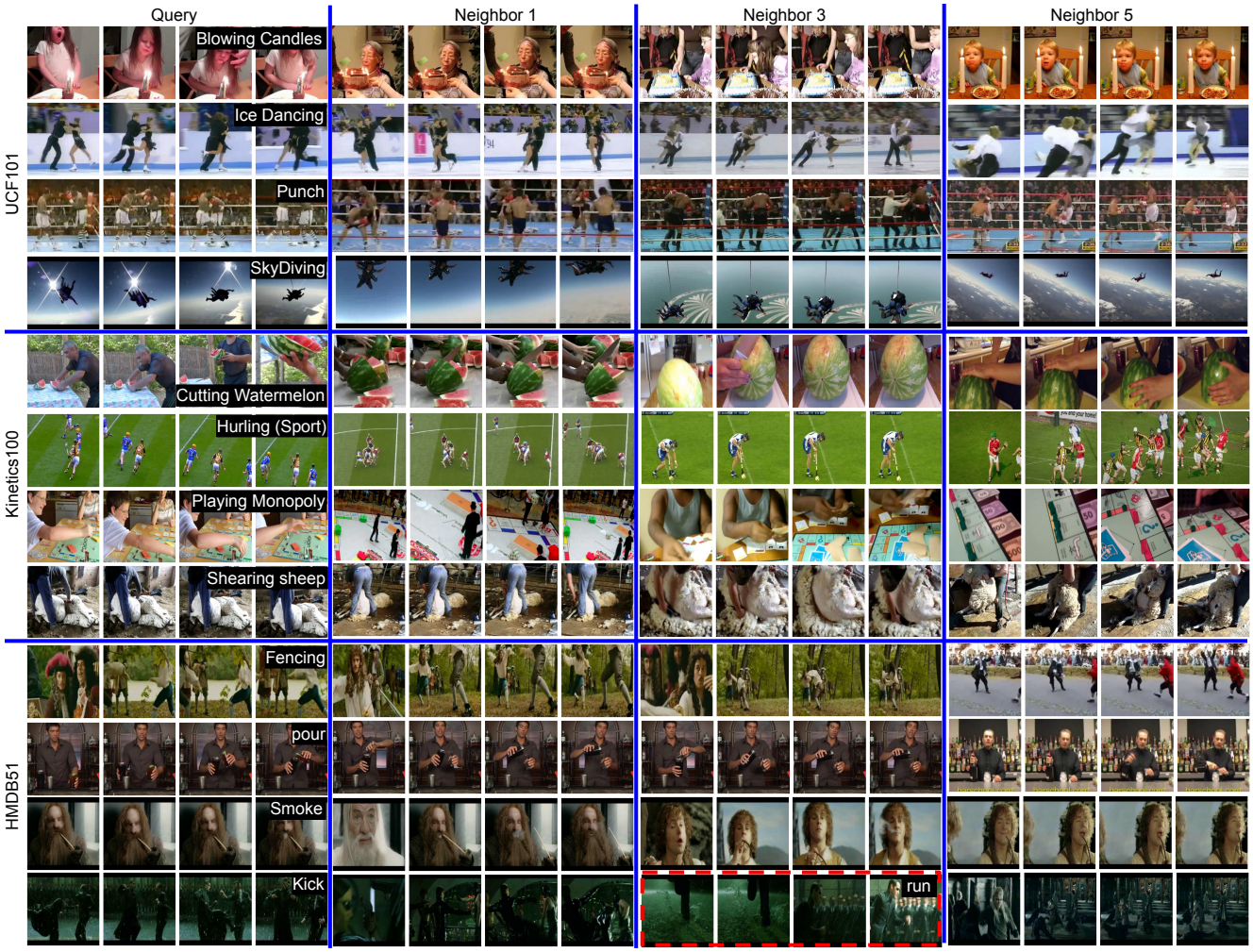

Figure 2. Nearest neighbor retrieval results with our unsupervised MetaUVFS action-appearance aligned video representations trained on our large unlabeled dataset. On the left side is a query video sampled from the novel classes in the UCF101, Kinetics100, and HMDB51 few-shot datasets. On the right side, we show among the top-5, *i.e.* the 1st, 3rd and 5th, nearest neighbors retrieved from the corresponding novel class dataset. The action class label for each video is shown in the upper right corner. The dashed-line box in red (bottom-most row) shows the model falsely retrieving another class 'run' due to query 'Kick' belonging to the same movie 'The Matrix'.

	UCF101 (unlabeled)				HMDB51 (unlabeled)				Kinetics100 (unlabeled)			
	Base Class		+ large unlabeled		Base Class		+ large unlabeled		Base Class		+ large unlabeled	
Methods	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
CVRL [5]	53.46 ± 0.46	75.41 ± 0.40	63.00 ± 0.41	87.80 ± 0.30	27.45 ± 0.38	30.91 ± 0.36	44.21 ± 0.45	60.35 ± 0.45	43.00 ± 0.41	58.35 ± 0.43	53.26 ± 0.48	71.39 ± 0.44
MetaUVFS (Ours)	65.31 ± 0.39	85.26 ± 0.47	**76.38 ± 0.40**	**92.50 ± 0.24**	36.72 ± 0.34	51.43 ± 0.45	**47.55 ± 0.45**	**66.13 ± 0.33**	50.28 ± 0.47	65.92 ± 0.44	**62.80 ± 0.45**	**79.55 ± 0.39**

Table 1. **Efficacy of acquiring and training on large unlabeled video dataset**. We compare the 1 and 5 shot 5-way performance of training models individually on 'Base Class' videos and in addition with the 'large-unlabeled' video dataset (total ∼550K videos). We compare our method against CVRL [5] which is the best-performing unsupervised baseline method. Unsupervised training on large-scale unlabeled dataset outperforms training on just base class video by a significant margin. For both data scenarios, MetaUVFS also significantly outperforms state-of-the-art CVRL that highlights the significance of having action-appearance aligned representations and an explicit few-shot learner to meta-learn from unlabeled videos.

we do this for the 10-way setting. As the 10-way task is expectedly harder than the 5-way task, 10-way T-SNE plots for both Kinetics100 and UCF101 look more scattered and mixed than for the 5-way task. The clusters formed for UCF101 look more compact and distinct as compared to Kinetics100. This observation is consistent with our quantitative analysis where the few-shot performance of MetaUVFS on UCF101 is higher than Kinetics100. Kinetics100 is also a relatively harder dataset than UCF101 as it is less constrained in terms of overall background setting and the

3D Action Encoder	Params	UCF101		HMDB51		Kinetics100	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
S3D [8]	9.6M	66.09 ± 0.46	88.42 ± 0.31	40.92 ± 0.47	61.03 ± 0.46	54.83 ± 0.48	73.64 ± 0.42
C3D [6]	27.7M	72.00 ± 0.42	90.88 ± 0.27	43.51 ± 0.47	63.70 ± 0.46	58.52 ± 0.46	76.93 ± 0.40
R(2+1)D-34 [7]	64.0M	67.06 ± 0.45	89.06 ± 0.30	41.51 ± 0.46	61.72 ± 0.46	55.63 ± 0.47	74.09 ± 0.42
Resnet50-3D (Ours)	44.5M	**76.38 ± 0.40**	**92.50 ± 0.24**	**47.55 ± 0.45**	**66.13 ± 0.33**	**62.80 ± 0.45**	**79.55 ± 0.39**

Table 2. Ablation study of MetaUVFS highlighting the choice of using ResNet-3D-50 as the 3D Action encoder compared to contemporary 3D architecture designs used by previous methods. ResNet-3D-50 is able to outperform all other backbones across all few-shot benchmarks.

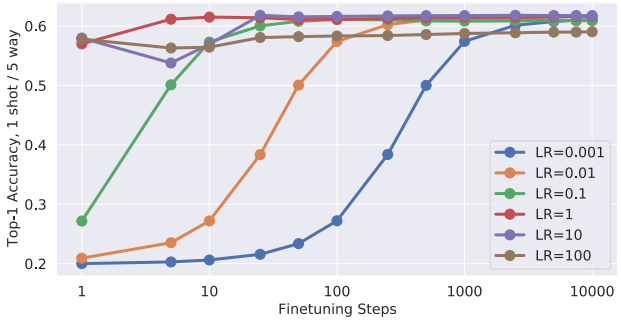

Figure 3. Comparison of accuracy for 1-shot, 5-way task on Kinetics100 for finetuning with different learning rate and number of steps.

temporal positioning of the action in the videos.

7. Videos for Unsupervised Hard Episode Generation

Our sampling and augmentation scheme prepare various augmentations for the same video as input to the action and appearance streams. Some augmentations are relatively harder (cosine distance similarity) for the InfoNCE discriminative learning. Figure 5 shows examples of 'hard' action and appearance augmentations drawn from unlabelled video samples. These augmented videos are sampled for generating episodes in our meta-learning A3M module.

8. Class splits

This section provides the list of classes whose videos were used for unsupervised training that comprises our 'large unlabeled video data'. These unlabeled videos are acquired from Kinetics700 [2], base classes videos from UCF101 and HMDB51 [9], and Kinetic100 [10]. We take extra precautions to ensure that there is no video in the training dataset belonging to the union of all the novel classes across all three evaluation datasets. This is to ensure that our testing is truly on a disjoint set of unseen classes.

8.1. Class Removal for Kinetics700

We remove the following 58 classes from Kinetics700 dataset:
blasting sand, busking, clean and jerk, cutting watermelon, dancing ballet, dancing charleston, dancing macarena, diving cliff, fencing (sport), filling eyebrows, folding paper, gymnastics tumbling, high jump, high kick, hula hooping, hurling (sport), ice skating, kicking field goal, kicking soccer ball, paragliding, playing drums, playing kickball, playing monopoly, playing tennis, playing trumpet, playing volleyball, pouring beer, pouring milk, pouring wine, punching bag, punching person (boxing), push up, pushing car, riding elephant, riding or walking with horse, rock climbing, running on treadmill, salsa dancing, shearing sheep, side kick, ski ballet, ski jumping, skiing crosscountry, skiing mono, skiing slalom, skipping rope, skydiving, smoking, smoking hookah, smoking pipe, springboard diving, standing on hands, stretching arm, surfing water, talking on cell phone, tap dancing, throwing axe, unboxing.

8.2. Class Removal for UCF101

We remove 4 base classes from UCF101 split:
Drumming, PushUps, Fencing, WallPushups.

8.3. Class Removal for HMDB51

We remove 4 base classes from HMDB51 split:
punch, handstand, push, throw

8.4. Class Removal for Kinetics100

We remove 1 base classes from Kinetics100 split:
blowing candles

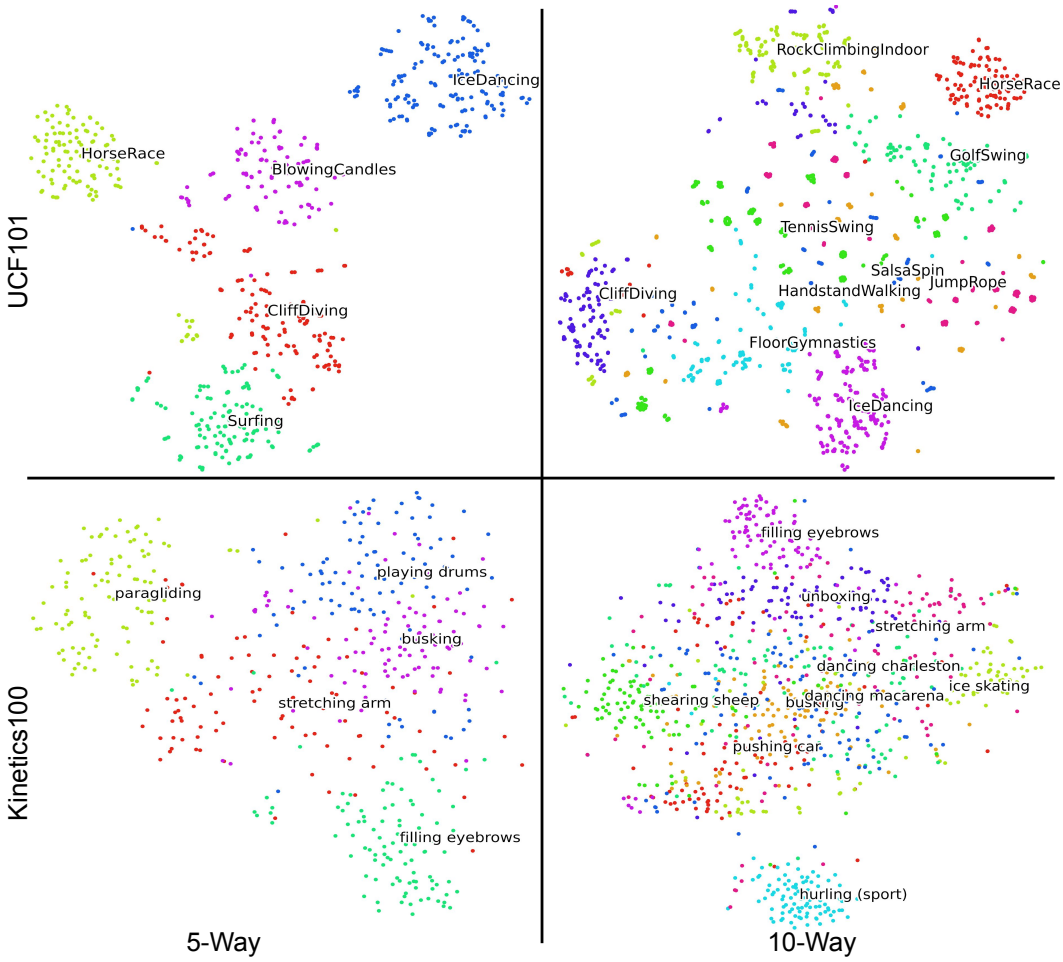

Figure 4. T-SNE plot for randomly chosen 5-way and 10-way settings on novel classes for UCF101 and Kinetics100 dataset. The clusters formed for UCF101 are more compact and well-separated than Kinetic100. This is consistent with the observation that the few-shot performance of MetaUVFS on UCF101 is higher than on Kinetics100.

References

[1] Sébastien MR Arnold, Praateek Mahajan, Debajyoti Datta, Ian Bunner, and Konstantinos Saitas Zarkias. learn2learn: A library for meta-learning research. *arXiv preprint arXiv:2008.12284*, 2020. 1

[2] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019. 5

[3] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017. 1, 3

[4] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008. 3

[5] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. *arXiv preprint arXiv:2008.03800*, 2020. 3, 4

[6] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 3, 5

[7] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 3, 5

[8] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 305–321, 2018. 3, 5

[9] Hongguang Zhang, Li Zhang, X Qui, Hongdong Li, Philip HS Torr, and Piotr Koniusz. Few-shot action recog-

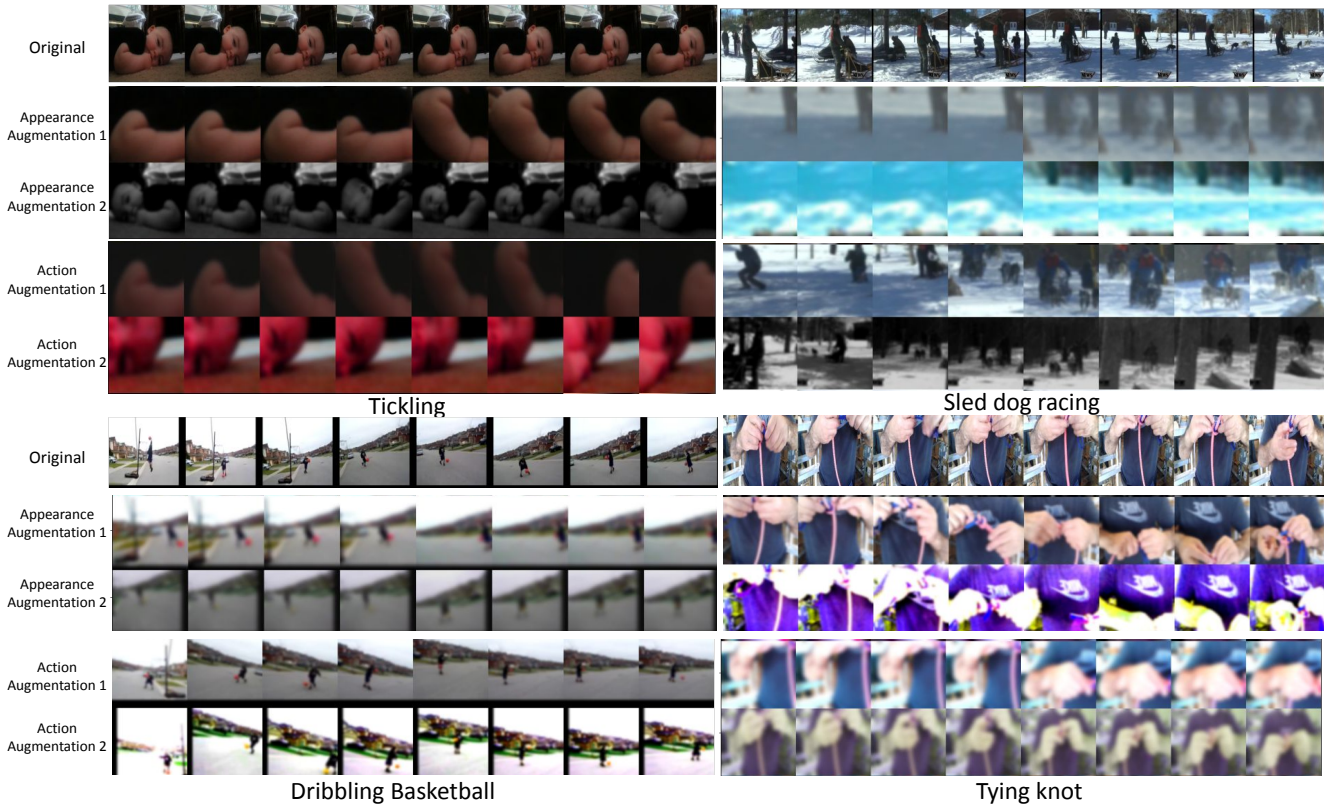

Figure 5. Figure showing some videos that are sampled to generate hard episodes to meta-train A3M module in MetaUVFS. For each video, top row shows original video frames. Next two rows show the frames comprising of the augmentations fed to the 2D appearance encoder. After that, the next two rows show the frames comprising of the augmentations fed to the 3D action encoder. For easy viewing in this figure, the action augmentations are shown with 8 frames, picking every other of the 16 frames, and resized to be the same size as appearance augmentations. Under each video is the label for the video from Kinetics700 dataset. Labels are not used when training MetaUVFS in an unsupervised manner.

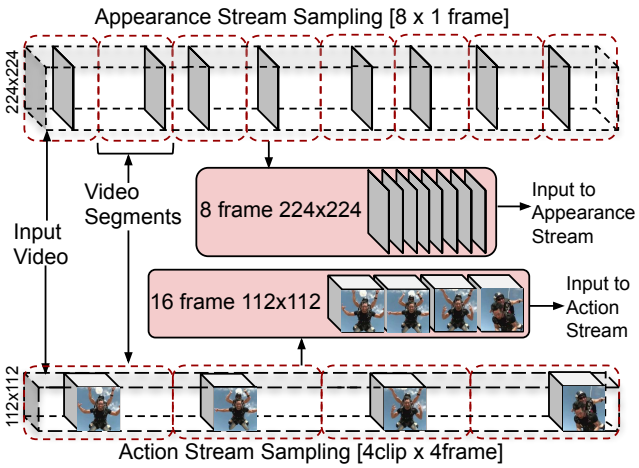

Figure 6. Sampling scheme for Appearance and Action streams.

nition with permutation-invariant attention. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 5

[10] Linchao Zhu and Yi Yang. Compound memory networks for few-shot video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 751–766, 2018. 5