Supplementary Material
Cross-modal Representation Learning for Zero-shot Action Recognition

Chung-Ching Lin, Kevin Lin, Lijuan Wang, Zicheng Liu, Linjie Li
Microsoft
{chungching.lin, keli, lijuanw, zliu, lindsey.li}@microsoft.com

This appendix presents more details of the limitations, implementation details, and extends the experimental section presented in the main manuscript.

1. **Limitations.** In Section 1, we discuss the limitations of the proposed method.
2. **Potential Negative Societal Impact.** We discuss the potential negative societal impact in Section 2.
3. **Implementation Details.** We provide network architecture of ResT, dataset statistics, and evaluation protocols in Section 3.
4. **Additional Ablation Study.** Additional ablation studies are presented in Section 4.
5. **Extended Experiments**. In Section 5, we extend the experimental section presented in the main manuscript. We demonstrate the generalization of the proposed method using pre-trained object features as model inputs and visualize qualitative results.

1. Limitations

The goal of this work is to provide a cleaner framework for zero-shot action recognition. In our setting, the model is not allowed to be pretrained on another dataset, and is evaluated on its ability to perform classification using unseen visual prototypes composited from seen visual prototypes. Although our method demonstrates its competitiveness on the benchmark datasets, it has limitations in some cases. For example, our model confuses similar actions, such as "laugh," "smile," and "chew." In these three classes, the actions mainly involve opening and shutting the jaws, but the muscle movements involved are subtle.

We also observe composite failures, e.g., for "hula hoop" where the class is named only by a noun of the main object, or for "playing daf" where the class is named by a general verb (ex: play, make, use) with a noun of a rare object. Our model is able to composite actions from other actions, but it exists a natural challenge to find relatedness for compositing out-of-distribution objects. However, if the setting of pure zero-shot is relaxed, our model could extend its capability via pretraining on another dataset, such as ImageNet.

Figure 1, 2, and 3 illustrate the confusion matrixes of ResT_18 model evaluated on UCF101, HMDB51, and ActivtyNet.

2. Potential Negative Societal Impact

Training and evaluating video understanding models are typically computationally intensive, which might significantly impact the environment. To alleviate this problem, we proposed a framework that reduces the computational demands for ZSAR. The potential negative impacts may include but are not limited to: (1) It poses a risk when directly applying action recognition models for decision making, especially in the health care and autonomous vehicle fields. (2) A video action recognition model can be misused, for example for unauthorized surveillance. Ethical considerations must be addressed in a real-world application.

3. Implementation Details

3.1. Network architecture of ResT

We describe the detailed network architecture of ResT in this section. ResT follows the design of the transformer encoder in [10]. As shown in Figure 4, the transformer encoder consists of alternating layers of multiheaded self-attention (MSA) and MLP blocks (Eq. 1, 2). Residual connections and layernorm (LN) are applied after every block. The MLP contains two fully-connected layers with a GeLU non-linearity. Our transformer consists of L layers. We denote $\boldsymbol{z_l}$ as the output of l^{th} layers.

$$\boldsymbol{z'_l} = LN(MSA(\boldsymbol{z_{l-1}}) + \boldsymbol{z_{l-1}}), \tag{1}$$

$$\boldsymbol{z_l} = LN(MLP(\boldsymbol{z'_l}) + \boldsymbol{z'_l}), \tag{2}$$

where $l = 1, ..., L$.

ResT uses the first token, z_0^0, to perform action classification on a source dataset. A classification head is attached to the output of the first token, z_L^0. We append a 1-hidden-layer MLP $f(\cdot)$, which is used to predict the final video classes.

$$x = LN(z_L^0) \tag{3}$$

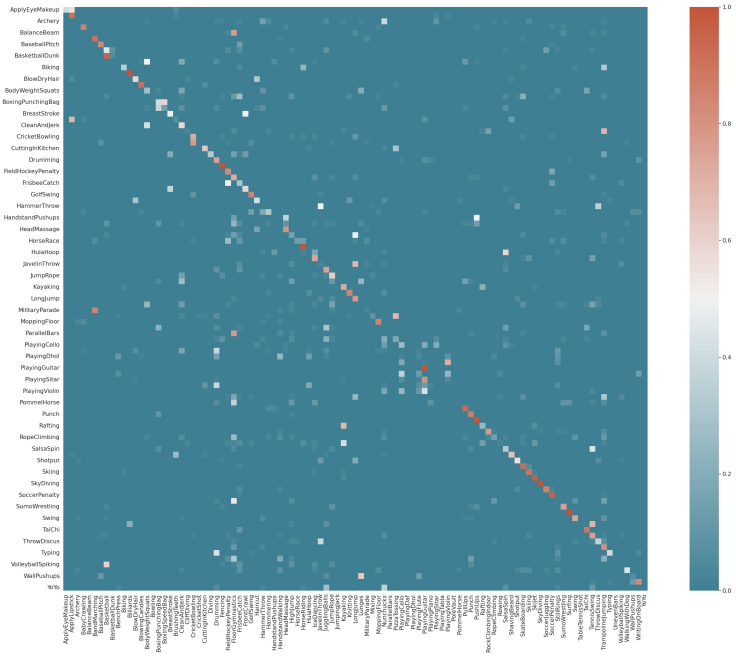


Figure 1. **Confusion matrix on UCF101 by ResT_18 (K664) model.**

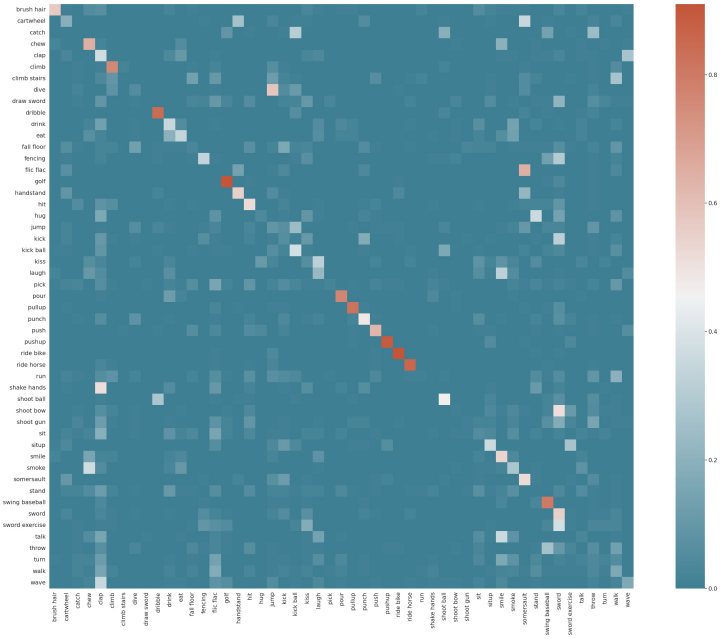


Figure 2. **Confusion matrix on HMDB51 by ResT_18 (K664) model.**

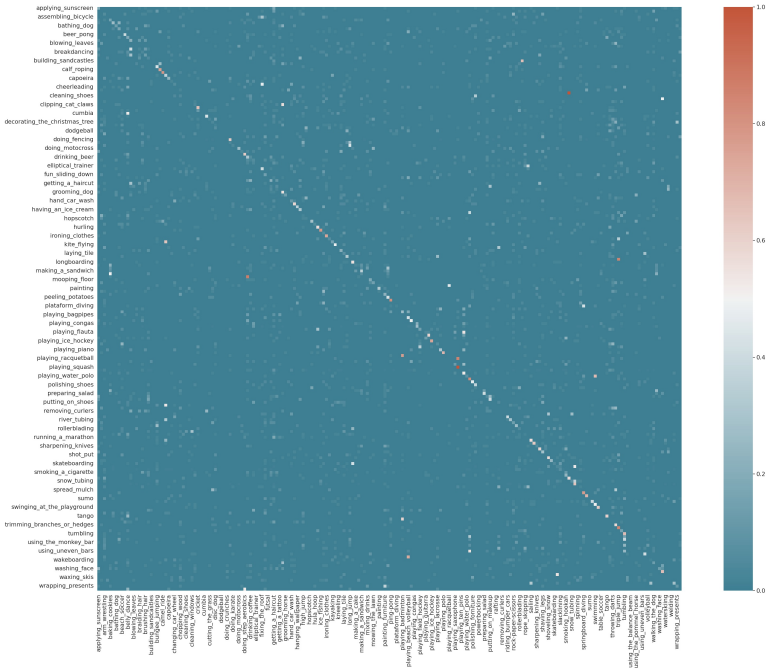


Figure 3. **Confusion matrix on ActivityNet by ResT_18 (K605) model.**

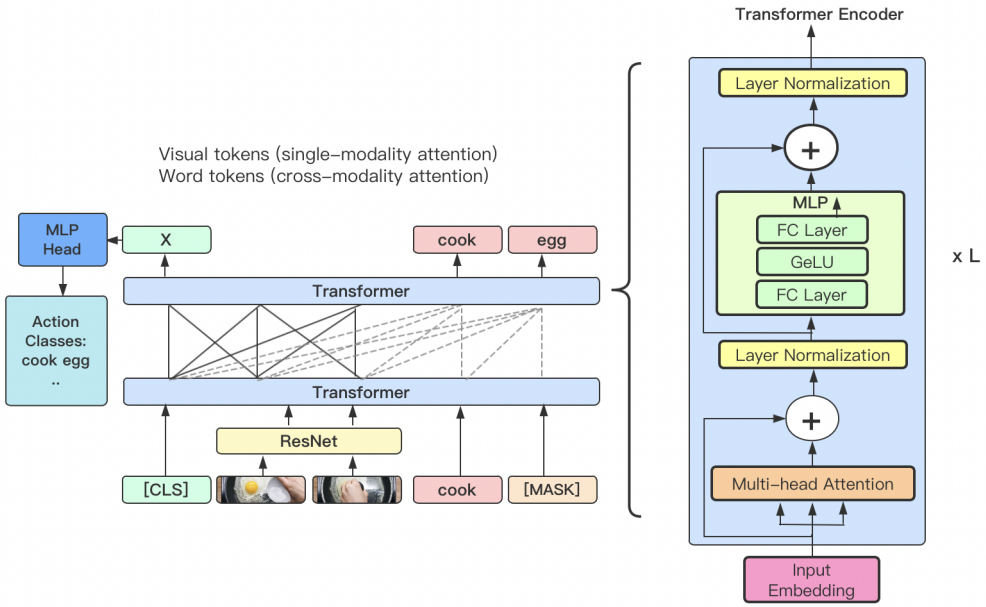


Figure 4. **Architecture of the transformer encoder in our proposed ResT.**

Our ResT consists of 12 transformer layers with a hidden size of 768D. The visual representation size is also 768D. The classifier weights are 768×664, and 768×605 corresponding to Kinetics 664 and 605 training sets.

3.2. Datasets

We train our models on a subset of the Kinetics dataset [4], and perform evaluations on three action recognition datasets: UCF101, HMDB51, and ActivityNet. UCF101 is labeled with 101 action categories with a focus on sports and contains 13,320 videos. HMDB51 has 6,767 videos with 51 classes. ActivityNet contains 200 classes and 27,801 untrimmed videos with an emphasis on daily activities. Kinetics dataset contains 700 classes with 545,317 training videos.

3.3. Evaluation protocol

In the zero-shot evaluation, we report results on half dataset (0/50 split) and full dataset (0/100). Most prior methods use pre-trained action recognition models to extract features, followed by training a ZSL model on 50% of the target dataset and testing on the other 50% of the same dataset to alleviate domain shift problems (50/50 setting). Our work follows E2E [2] to adopt a cross dataset configuration, where the models are only trained once on a source action recognition dataset and then are directly evaluated on 50% of other target datasets. The goal of 0/50 setting is to disallow tailoring ZSAR models to a specific test dataset. In the 0/50 split, we randomly choose 50% classes from the test dataset: 50 on UCF101, 25 on HMDB51, and 100 on ActivityNet. On each test set, we randomly generate 10 splits and report the averaged results. As E2E [2] and our method are trained on a separate dataset, we are able to test our models on full UCF101, HMDB51, and ActivityNet datasets (0/100).

4. Additional Ablations

In this section, we extend the experimental section presented in the main manuscript.

4.1. Influence of removing overlapping classes

Table 1. **Accuracy comparisons on models trained on Kinetics 664 and full Kinetics 400/700 datasets.** All models are evaluated on 25 clips on the 50% of UCF101 and HMDB51 datasets.

	UCF (0/50)	
ResT_101 Model	Top-1	Top-5
664 classes	58.7	75.9
400 classes	61.1	79.2
700 classes	69.2	83.8

In this experiment, we compare our models trained on Kinetics 664 (with overlapping classes removed) with the models trained on the full Kinetics 400 and 700 datasets [4] (without overlapping classes removed) to demonstrate that removing overlapping classes is a non-trivial learning constraint. The results are reported in Table 1. It can be seen that the models trained on the full Kinetics dataset obtain higher Top-1 accuracy than the models trained on the sets without overlapping classes (e.g., 2.4% absolute gains in Top-1 accuracy on 0/50 configuration from K664 to K400 and 10.5% gains from K664 to K700). As discussed in the main manuscript, one has to ensure that the seen and unseen classes are disjoint and the zero-shot setting is maintained when external datasets are involved.

4.2. Importance of constraints in semantic relatedness transfer

The design of the transfer scheme aims to embed a combination of the most representative and distinctive information for effective knowledge transfer. It follows, the proposed framework is thus less prone to the hubness problem and the bias with NN search. In this section, we discuss the importance of the constraints in the semantic relatedness transfer.

The hubness problem is related to the high-dimensional nearest neighbor search. That is, some points (hubs) frequently occur in the k-nearest neighbor set of other points. The skewness of an empirical ψ_k distribution is typically used to measure the degree of hubness [7, 8]. The distribution ψ_k is the distribution of the number of times ($\psi_k(j)$) the j^{th} prototype is in the top k nearest neighbors of the test samples. The skewness of the distribution is defined as:

$$\psi_{k\_skewness} = \frac{\sum_{j=1}^{\gamma}(\psi_k(j) - E[\psi_k])^3}{Var[\psi_k]^{\frac{3}{2}}}, \tag{4}$$

where γ is the total number of test prototypes. A higher skewness value indicates a more severe hubness issue.

Here, we summarize the three constraints imposed in the semantic relatedness transfer. Constraint I is to ensure the composited unseen prototypes are representative. Constraint II and III together promote the composited visual prototypes of unseen classes to be distinctive from one another.

In Table 2, we discuss the effect of the constraints in terms of the degree of hubness ($\psi_{1\_skewness}$) and classification accuracy (Top-1/ Top-5). The accuracy is evaluated using one clip with ResT_18 (664/605) model. We consider five combinations: (1) Reverse transfer direction (composite semantic representation and perform ZSAR in the semantic space), (2) No constraints imposed, (3) Only constraint I, (4) Only constraint II and III, (5) All constraints.

Table 2. **Effect of constraints in the semantic relatedness transfer scheme**

(a) Evaluation on UCF101 dataset

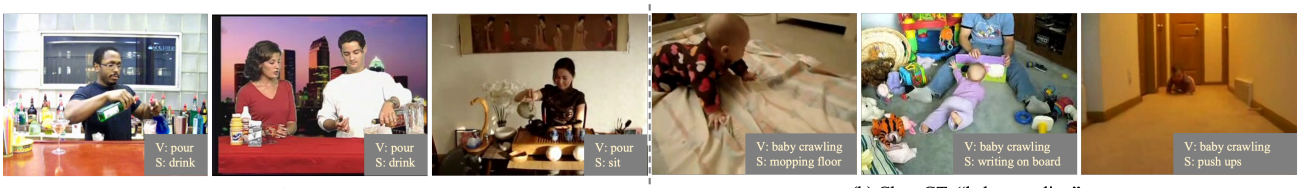
			UCF (0/50)	
ResT_18 (664)	ZSAR in V or S space	Skewness	Top-1	Top-5
Reverse transfer	S	3.350	36.3	69.2
No constraint	V	2.235	38.2	73.9
Constraint I	V	1.342	50.7	81.5
Constraint II + III	V	1.290	51.8	74.4
All constraints	V	1.228	54.0	74.6

(b) Evaluation on HMDB51 dataset

			HMDB (0/50)	
ResT_18 (664)	ZSAR in V or S space	Skewness	Top-1	Top-5
Reverse transfer	S	3.712	35.0	63.9
No constraint	V	1.688	35.1	64.7
Constraint I	V	1.379	37.9	68.5
Constraint II + III	V	0.849	38.1	64.5
All constraints	V	1.418	39.2	66.9

(c) Evaluation on ActivityNet dataset

			ActivityNet (0/50)	
ResT_18 (605)	ZSAR in V or S space	Skewness	Top-1	Top-5
Reverse transfer	S	2.742	21.9	40.1
No constraint	V	1.827	21.2	45.9
Constraint I	V	1.173	25.1	51.5
Constraint II + III	V	1.228	25.4	40.8
All constraints	V	1.020	26.2	47.4



(a) Class GT: "pour" (b) Class GT: "baby crawling"

Figure 5. **Sample results of ZSAR in visual space (V) and semantic space (S).**

We draw several conclusions from Table 2: (1) In general, we observe Top-1 accuracy is negatively affected by the presence of hubs, and the hubness problem is more likely to arise in the semantic space than the visual space. We visualize some qualitative results of ZSAR in different spaces in Figure 5. (2) Compared to 'No constraint,' all combinations of the constraints help improve the classification accuracy and alleviate the effect of the hubness problem. (3) When applying constraint I only, Top-5 accuracy is consistently higher because the constraint filters out the less related classes. (4) Constraint II and III are effective, ensuring the distinction of the composited unseen prototypes. In general, it obtains a low hubness value with these two constraints. (5) Combining all three constraints yields a filter-and-refine methodology. Overall, it achieves the best Top-1 accuracy with a relatively low degree of hubness because these three constraints together consider both representative and distinctive.

5. Extended Experiments

Although we propose a framework where no pre-training on additional datasets is performed to ensure no prior knowledge of unseen classes is acquired during training, our model is flexible and capable of cooperating with existing pre-trained models.

In the ablation study, we show the generalization of the proposed model by taking pre-trained object region features as inputs. Considering the essence of zero-shot setting, it might be arguable if using object information is incongruous with the idea of pure ZSAR because it is highly likely that some major objects occur in seen and unseen classes, and some unseen classes are simply named for objects (e.g., "yo-yo," "uneven bars," and "pommel horse"). However, modeling objects helps with the model interpretability. In this experiment, to prevent the model from achieving high accuracy by matching object names instead of recognizing actions, we only use object region features as model inputs to examine the capability of the proposed model. **Detection outputs (object labels) are not used in the experiment.**

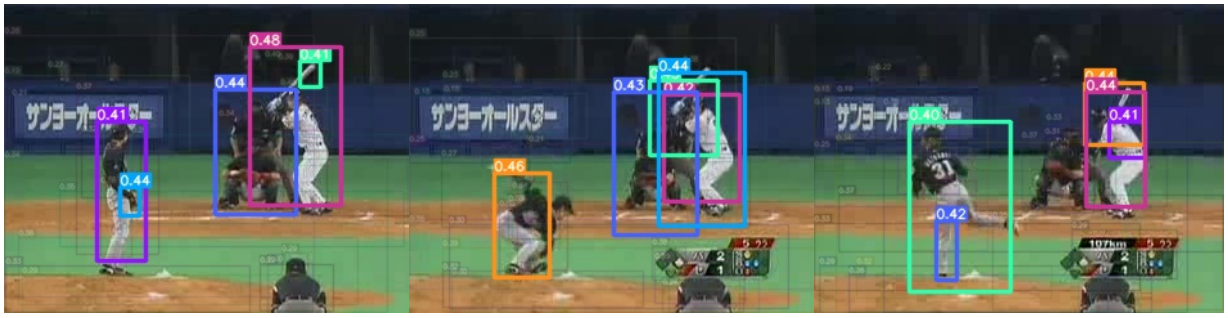
In this experiment, we replace frame-level features with object region features. We start with object feature extractors, an off-the-shelf detection network, UpDown [1]. The UpDown detector was trained on Visual Genome dataset [5]. For a frame I^t sampled at time t in a video v, an amount $r'^t = [r'^t_1, ..., r'^t_{N^t_r}]$ of N^t_r object features are extracted by the detector, where $r'^t_k \in \mathbb{R}^p$ is a p-dimension vector. To encode spatiotemporal information, we construct a 7-d vector s^t_k from the region position (normalized four corner coordinates, width, and height) and the frame index (normalized frame index offset). We concatenate object feature r'^t_k and the spatiotemporal vector s^t_k in order to form a spatiotemporally sensitive region vector r^t_k.

We report the results of our model using object region features as model inputs in Table 3. It shows that our model is able to handle contextual information in object features and make the classification of actions relatively effective.

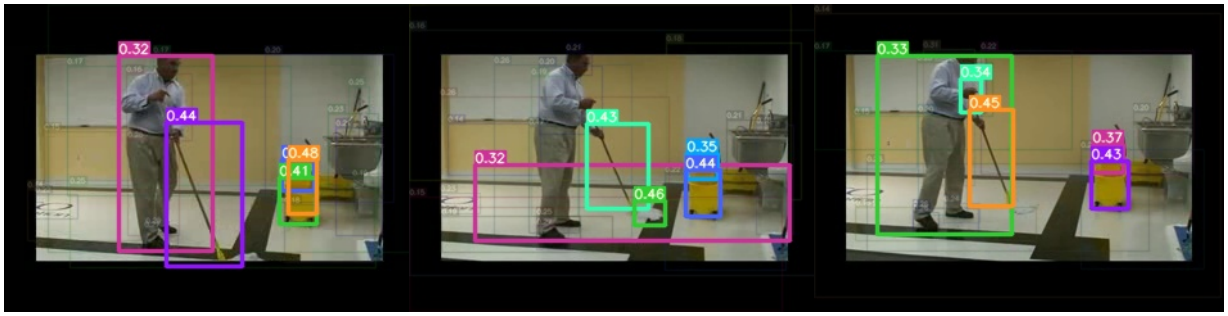
Table 3. **ZSAR performance with ResNet and object features on the 50% of UCF101, HMDB51, and ActivityNet datasets.**

Model	UCF101	HMDB51	ActivityNet
K664			
ResT_18	54.7	39.3	-
ResT_101	58.7	41.1	-
Ours_obj	57.3	39.6	-
K605			
ResT_18	50.9	37.6	29.2
ResT_101	55.9	40.8	32.5
Ours_obj	55.0	40.5	34.2

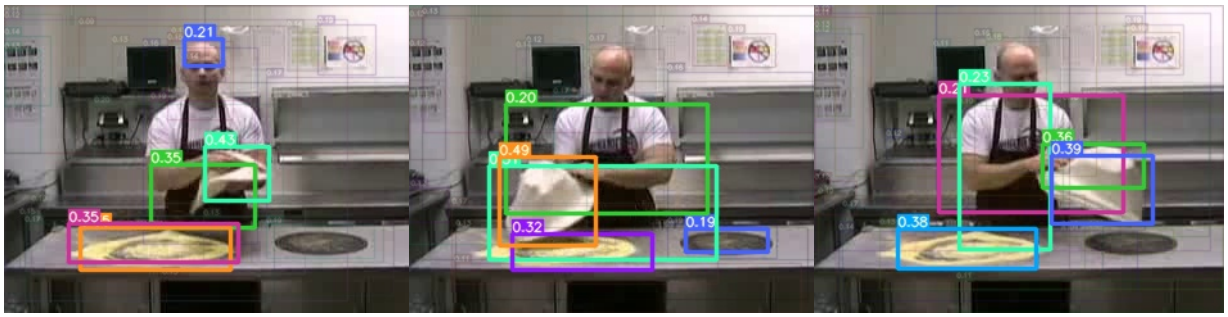
Figure 6, 7, and 8 illustrate snapshots of action samples on the UCF101 [9], HMDB51 [6] and ActivityNet [3] that are correctly classified by our model. Each subfigure presents three sample video frames from one action clip. Each frame highlights the five most attended object regions by our network for action recognition. We observe that our model focuses on the active objects where an action is taking place and attends to the most indicative objects, e.g.,"mop handle and head" and "water bucket" in Figure 6(b) or "pizza dough" in Figure 6(c). For example, in Figure 7(c), a sample from the class "eat" on HMDB51, our model attends to the mouth, spoon, and hand. Similarly, in Figure 8(c), a sample from the class "playing beach volleyball" on ActivityNet, our model focuses on the player who sets the volleyball in the first frame, the player who steps toward the ball and bends the knee in the middle frame, and then the same player jumping and preparing for spiking the ball in the last frame. These examples demonstrate the effectiveness of our transformer-based framework that learns to capture the evolution of human actions by observing the most relevant and visually descriptive objects.



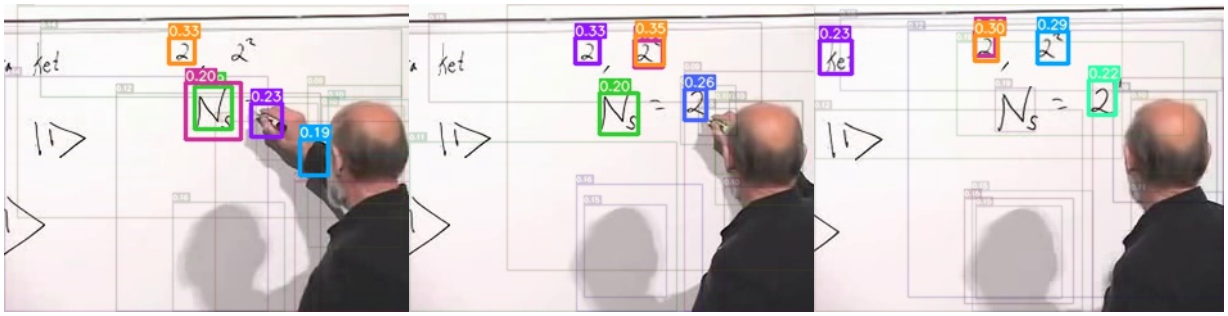
(a) Action class: "**Baseball pitch**".



(b) Action class: "**Mopping floor**".

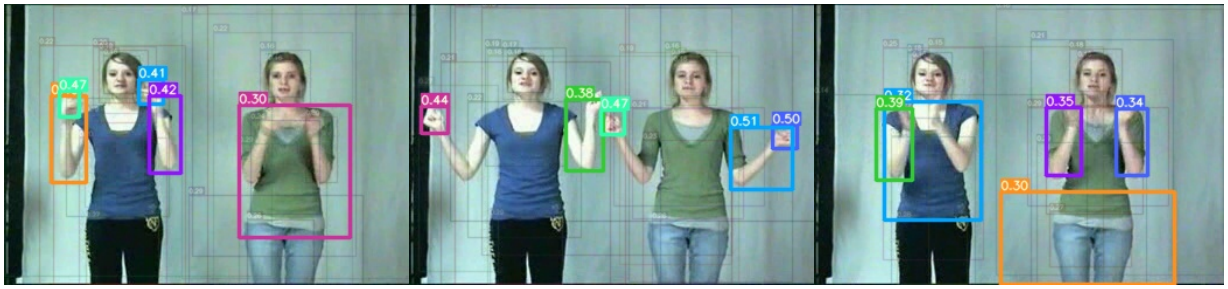


(c) Action class: "**Pizza tossing**".



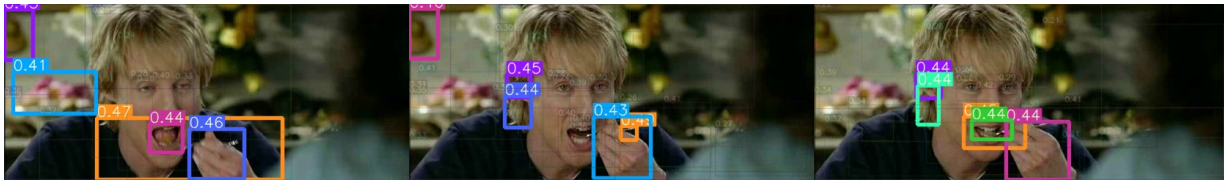
(d) Action class: "**Writing on board**".

Figure 6. Example results from our model with object region features as inputs on the classification of the "baseball pitch," "mopping floor," "pizza tossing," and "writing on board" actions on UCF101 dataset. Different bounding boxes are coded with different colors. Brighter colors depict the most attended objects. Best viewed in color.

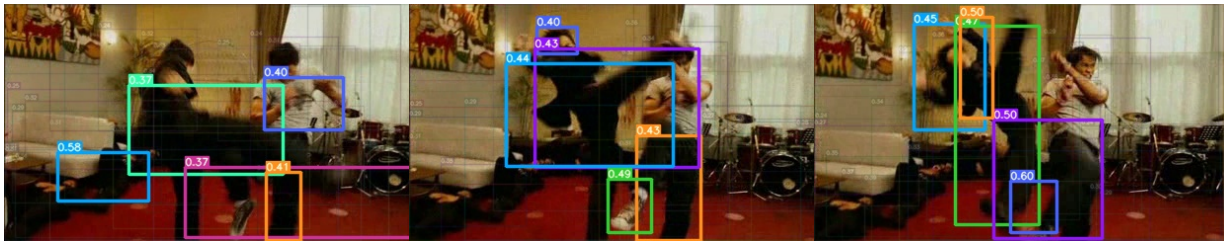


(a) Action class: "**Clap**".

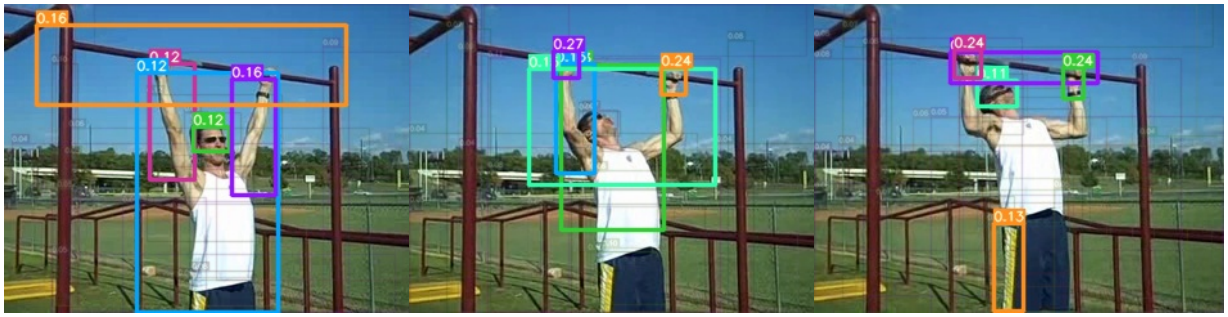
(b) Action class: "**Drink**".



(c) Action class: "**Eat**".

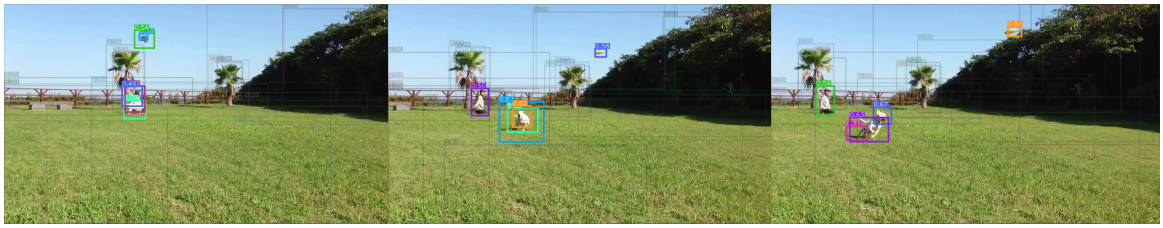


(d) Action class: "**Kick**".

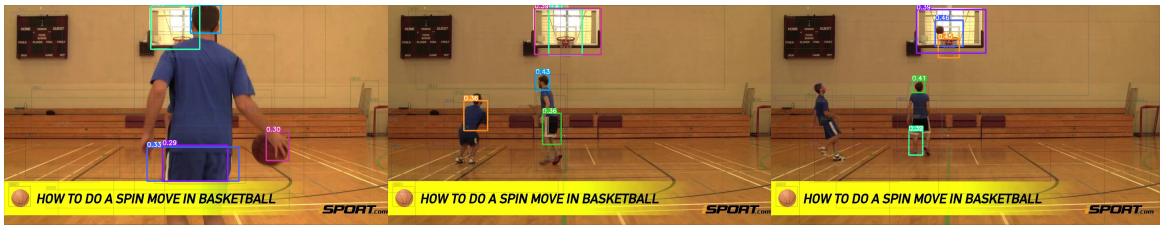


(e) Action class: "**Pullup**".

Figure 7. Example results from our model with object region features as inputs on the classification of the "clap," "drink," "eat," "kick," and "pullup" actions on HMDB51 dataset. Different bounding boxes are coded with different colors. Brighter colors depict the most attended objects. Best viewed in color.

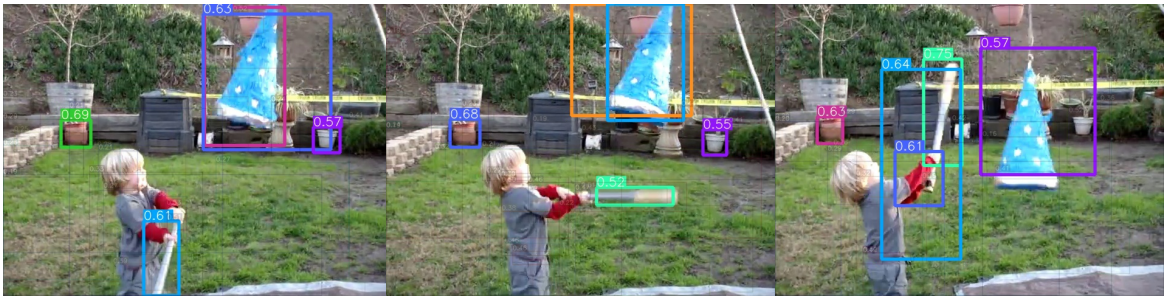


(a) Action class: "**Disc dog**".

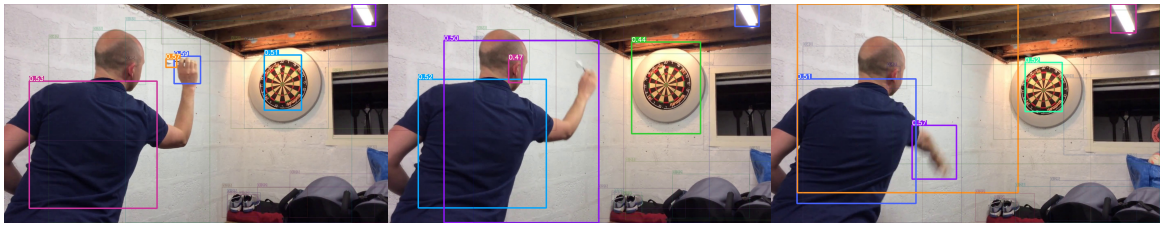


(b) Action class: "**Layup drill in basketball**".

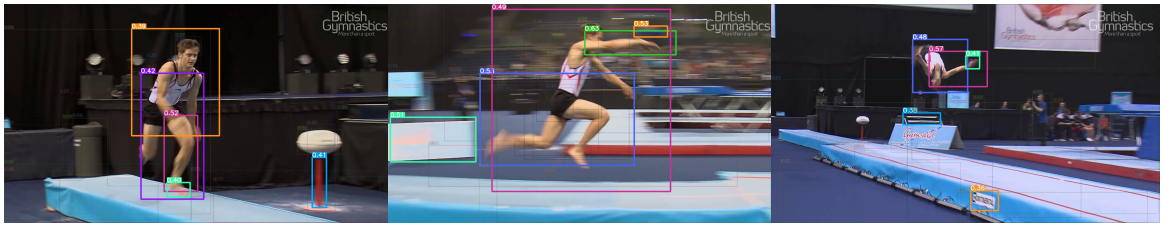
(c) Action class: "**Playing beach volleyball**".



(d) Action class: "**Hitting a pinata**".



(e) Action class: "**Throwing darts**".



(f) Action class: "**Tumbling**".

Figure 8. Example results from our model with object region features as inputs on the classification of the "disc dog," "layup drill in basketball," "playing beach volleyball," "hitting a pinata," "throwing darts," and "tumbling" actions on ActivityNet dataset. Different bounding boxes are coded with different colors. Brighter colors depict the most attended objects. Best viewed in color.

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086, 2018. 6

[2] Biagio Brattoli, Joseph Tighe, Fedor Zhdanov, Pietro Perona, and Krzysztof Chalupka. Rethinking zero-shot video classification: End-to-end training for realistic applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4613–4623, 2020. 4

[3] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970, 2015. 6

[4] 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. 4

[5] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. 2016. 6

[6] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563. IEEE, 2011. 6

[7] Milos Radovanovic, Alexandros Nanopoulos, and Mirjana Ivanovic. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research*, 11(sept):2487–2531, 2010. 4

[8] Yutaro Shigeto, Ikumi Suzuki, Kazuo Hara, Masashi Shimbo, and Yuji Matsumoto. Ridge regression, hubness, and zero-shot learning. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 135–151. Springer, 2015. 4

[9] 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. 6

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017. 1