Holistic Interaction Transformer Network for Action Detection

Gueter Josmy Faure Wei-Jhe Huang Cheng-Yu Ho
Jheng-Hsien Yeh Qing-Wen Yang Shang-Hong Lai

Department of Computer Science, National Tsing Hua University, Taiwan

Abstract

This technical report describes our method for the ECCV DeeperAction Challenge - MultiSports track. The proposed network is a bi-modal framework comprising RGB and pose streams. Each of them separately models person, object, and hand interactions. Within each sub-network, an Intra-Modality Aggregation module (IMA) is introduced that selectively merges individual interaction units. The resulting features from each modality are then aggregated using an Attentive Fusion Mechanism (AFM). Finally, we extract cues from the temporal context to better classify the occurring actions using cached memory.

1. Method

Our Holistic Interaction Transformer (HIT) network is concurrently composed of an RGB and a pose sub-network. Each aims to learn persons' interactions with their surroundings (space) by focusing on the key entities that drive most of our actions (e.g., objects, pose, hands). After fusing the two sub-networks' outputs, we further model how actions evolve in time by looking at cached features from the past and future.

1.1. Overall Framework

Given an input video $V_{in} \in \mathbb{R}^{C\times T\times H\times W}$ we extract video features $V_b \in \mathbb{R}^{C\times T\times H\times W}$ by applying a 3D video backbone. Afterward, using ROIAlign, we crop person features $\mathcal{P}$, object features $\mathcal{O}$, and hands features $\mathcal{H}$ from the video. We also keep a cache of memory features which is denoted as $\mathcal{M} = [t-S, ..., t-1, t+1, ..., t+S]$, where $2S$ is the temporal window. Parallelly, we use a pose model to extract person keypoints $\mathcal{K}$ from each keyframe of the dataset. Further, the RGB and pose sub-networks compute the RGB feature F_{rgb} and pose feature F_{pose}, respectively. These features are fused and subsequently used as anchors for learning global context information to obtain F_{cls}. Finally, our network outputs $\hat{y} = g(F_{cls})$, where g is the classification head. The overall framework is shown in Fig. 1.

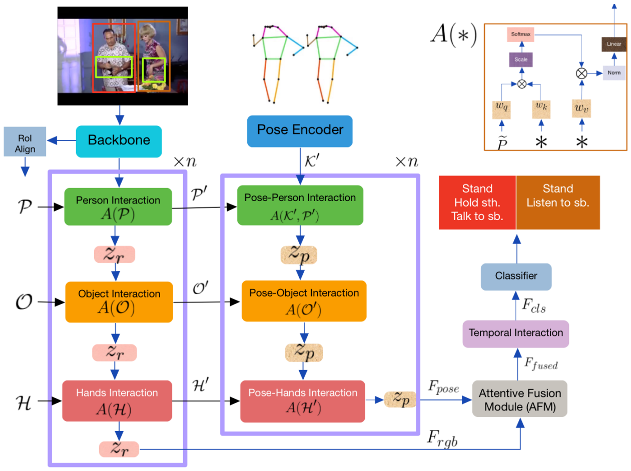


Figure 1. **Overview of our framework.** On top of our RGB stream is a 3D CNN backbone which we use to extract video features. Our pose encoder is a spatial transformer model. We compute rich local information from both sub-networks using person, hands, and object features. Then, we combine the learned features using an attentive fusion module before modeling their interaction with the global context.

1.2. Entity Selection

HIT consists of two mirroring modalities with distinct modules designed to learn different types of interactions. Human actions are largely based on their pose, hand movements (and pose), and interaction with other entities in the frame. Based on these observations, we select human poses and hands bounding boxes as entities for our model, along with object and person bounding boxes. We use Detectron [4] for human pose detection and create a bounding box encircling the location of the person's hands. Following the state-of-the-art methods, [17], [15], [13], we use Faster-RCNN [14] to compute object bounding box proposals. We use the person bounding boxes from https://github.com/MCG-NJU/MultiSports at inference time. The video feature extractor is a 3D CNN backbone network [3], and the pose encoder is a lightweight spa-

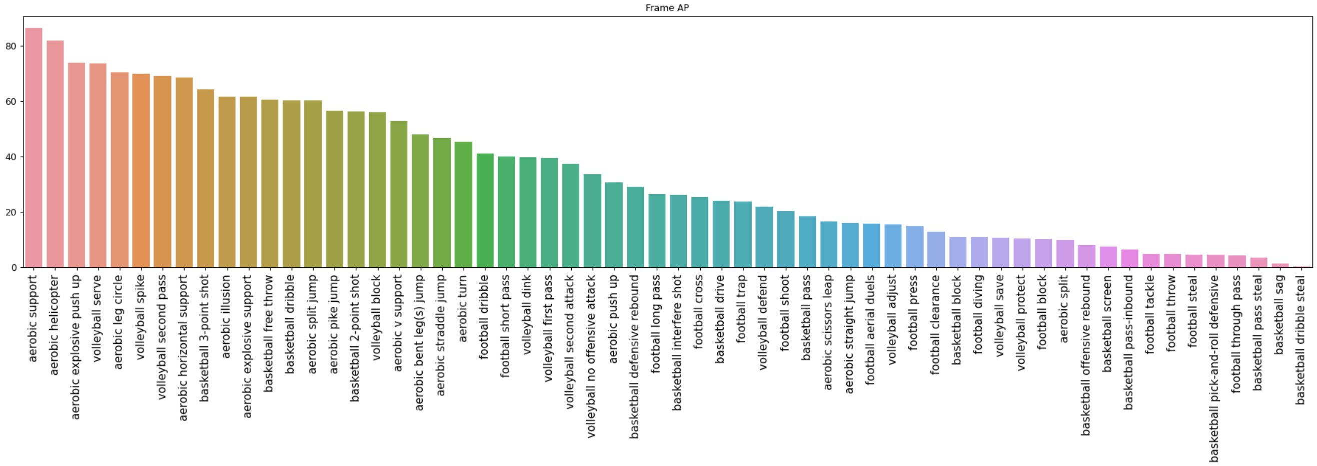


Figure 2. **Per-class frame AP results.** Our per-class frame AP score follows the long-tailed distribution pattern.

tial transformer inspired by [19]. We apply ROIAlign [5] to trim the video features and extract person and local context features (hands and objects).

1.3. The RGB Branch

The RGB branch comprises three main components, as shown in Figure 1. Each performs a series of operations to learn specific information concerning the target person. The person interaction module learns the interaction between persons in the current frame (or self-interaction when the frame contains only one subject). The object and hands interaction modules model person-object and person-hands interaction, respectively. An illustration of the interaction module is shown in Figure 1 (top-right corner). At the heart of each interaction unit is a cross-attention computation where the query is the target person (or the output of the previous unit), and the key and value are derived from the objects, or the hands features, depending on which module we are at. The following equations summarize the RGB branch's flow.

$$
\begin{aligned}
F_{rgb} = (A(\mathcal{P}) \rightarrow z_r \rightarrow A(\mathcal{O}) \rightarrow z_r \rightarrow A(\mathcal{H}) \rightarrow z_r) \\
A(*) = softmax(\frac{w_q(\widetilde{P}) \times w_k(*)}{\sqrt{d_r}}) \times w_v(*) \\
z_r = \sum_b A(b) \times softmax(\theta_b),
\end{aligned}
\tag{1}
$$

where $b \in (\widetilde{P}, \mathcal{O}, \mathcal{H}, \mathcal{M})$, d_r represents the channel dimension of the RGB features, w_q, w_k and w_v project their inputs into query, key and value, respectively. Note that $A(*)$ is the cross-attention mechanism. It only takes person features as input when computing person interaction $A(\mathcal{P})$. However, for hand interaction (objects interaction), it takes two sets of input: the output of z_r, which serves as query (denoted as $\widetilde{P}$), and the hands features (object features) from which we obtain the key and values.

The intra-modality aggregation component, z_r is the weighted sum of all interaction modules, including the temporal interaction module TI. z_r is essential for two main reasons. First, it allows the network to aggregate as much information as possible, efficiently. Secondly, the learnable parameter θ helps filter the different sets of features, hand-picking the best each of them has to offer while discarding noisy and unimportant information.

1.4. The Pose Branch

The pose model is similar to its RGB counterpart and reuses most of its outputs. We first extract the pose features $\mathcal{K}'$ by using a light transformer encoder f inspired by [19].

$$
\mathcal{K}' = f(\mathcal{K}) \tag{2}
$$

Then we compute F_{pose} by mirroring the different constituents of the RGB modality. Here, $\mathcal{P}'$, $\mathcal{O}'$, and $\mathcal{H}'$ are the corresponding outputs of $A(\mathcal{P})$, $A(\mathcal{O})$, and, $A(\mathcal{H})$.

$$
F_{pose} = (A(\mathcal{K}', \mathcal{P}') \rightarrow z_p \rightarrow A(\mathcal{O}') \rightarrow z_p \rightarrow A(\mathcal{H}') \rightarrow z_p)
$$

$$
A(\mathcal{K}', \mathcal{P}') = softmax(\frac{w_q(\mathcal{K}') \times w_k(\mathcal{P}')}{\sqrt{d_p}}) \times w_v(\mathcal{P}') \tag{3}
$$

where $A(\mathcal{K}', \mathcal{P}')$ computes the cross-attention between the pose features $\mathcal{K}'$ and the enhanced person interaction features $\mathcal{P}'$. Such a cross-modal blend enforces the pose features by focusing on the key corresponding attributes of the RGB features. The other components, $A(\mathcal{O}')$ and $A(\mathcal{H}')$ take a linear projection of z_p as query while their key-value pairs stem from $A(\mathcal{O})$ and $A(\mathcal{H})$. z_p is the intra-modality aggregation component for the pose model. Similar to z_r, it filters and aggregates information from each interaction module.

1.5. The Attentive Fusion Module (AFM)

At some point in the network, the RGB and pose streams need to be combined into one set of features before being fed to the action classifier. For this purpose, we propose an Attentive Fusion Module that applies channel-wise concatenation of the two feature sets followed by self-attention for feature refinement.

$$F_{fused} = \Theta_{fused}(SelfAttention(F_{rgb}, F_{pose})) \tag{4}$$

1.6. Temporal Interaction Unit

Following the fusion module is a temporal interaction block (TI). Human actions happen in a continuum; therefore, long-term context is essential to understanding actions. Along with F_{fused}, this modules receives compressed memory data $\mathcal{M}$ with length $2S$. Inspired by [17], the memory cache contains the person features extracted by the video backbone. TI is another cross-attention module where F_{fused} is the query and two different projections of the memory $\mathcal{M}$ form the key-value pair.

$$F_{cls} = TI(F_{fused}, \mathcal{M}) \tag{5}$$

1.7. Implementation Details

Dataset: The **MultiSports** dataset [9] contains 66 fine-grained action categories from four different sports spanning more than 3200 video clips with 37701 action instances and 902k bounding boxes. Actions are annotated at 25 FPS, and each video clip lasts around 22 seconds.

Backbone Network. We use SlowFast [3] R101 instantiation pre-trained on the Kinetics-700 dataset [1].

Person and Object Detector: We extract keyframes from each video in the dataset and use detected person bounding boxes from https://github.com/MCG-NJU/MultiSports for inference. According to the authors, these boxes are generated by the person detector of Faster R-CNN with a ResNeXt-101-FPN. As object detector, we employ Faster-RCNN [14] with ResNet-50-FPN [11, 18] backbone. The model is pretrained on ImageNet [2], and fine-tuned on MSCOCO [12].

Keypoints Detection and Processing: For keypoints detection, we adopt a pose model from Detectron [4]. The authors use a Resnet-50-FPN backbone pretrained on ImageNet for object detection and fine-tuned on MSCOCO keypoints using precomputed RPN [14] proposals. Each keyframe from the target dataset is passed through the model, which outputs 17 keypoints for each detected person, corresponding to the COCO format. We further post-process the detected pose coordinates, so they match the groundtruth person bounding boxes (during training) and the detected bounding boxes from https://github.com/MCG-NJU/MultiSports (during evaluation and testing). For person hands location, we are only interested in the keypoints referring to the person's wrists; therefore, we make a bounding box out of these two keypoints to highlight the person's hands and everything in between.

Training and Evaluation: The input videos are sampled 64 frames per clip, with $\alpha = 8$ and $\tau = 4$. During training, random jitter augmentation is applied to the ground-truth human bounding boxes. For object boxes, we use the ones with detection score $\geq$ 0.25 and whose IoU with any person bounding box in the same frame is positive. This is to ensure that only the objects with relatively high confidence scores and those with which humans directly interact are included in our sample. We use a memory span of $S = 30$ for Temporal Interaction. The network is trained for 150k iteration with the first 2000 iterations serving as linear warm-up. The starting learning rate of 0.0004 is reduced by a factor of 10 at iterations 90k and 110k. We use SGD as optimizer and a batch size of 16 to train the model on 8 GPUs. At inference/test time, we predict action labels for human bounding boxes provided by https://github.com/MCG-NJU/MultiSports with a confidence threshold of 0.8. Softmax focal loss is used as activation function for the classifier. Our model outputs frame detection results and we create action tubes using the format provided by ACT [7].

1.8. Ablation Study

Since the MultiSports dataset is heavy, we first perform ablation experiments on the J-HMDB dataset [6] to confirm the effectiveness of our model and its constituents, then transport the best configuration to MultiSports. All ablations are performed using the SlowFast-Resnet50 video backbone. We use frame mAP with an IoU threshold of 0.5 as evaluation metric.

Network Depth: Two layers of our network are enough to learn valuable features conducing to accurate action detection. As shown in Table 1b, a two-layer setting improves the mAP by more than 4% compared to having just one, while adding a third induces overfitting.

Attentive Fusion Module (AFM): We used an Attentive Fusion Mechanism (AFM) to combine features from the two modalities. Equipped with self-attention, it helps smoothen the fusion process between different modalities. We corroborate this choice by comparing it with Sum, $Concat$, $WeightedSum$, and $Average$.

Bi-modal fusion	mAP
Sum	78.60
Concat	78.77
WeightedSum	80.21
Average	81.35
AFM	**83.81**

(a) **Bi-modal fusion methods**

Depth	mAP
1 layer	79.21
2 layers	**83.81**
3 layers	81.54

(b) **Network Depth**

	mAP
After TI	82.16
Before TI	**83.81**

(c) **Late versus early fusion**

	mAP
w/o IMA	79.80
w/ IMA	**83.81**

(d) **Importance of IMA**

	mAP
Backbone	58.85
Backbone + AIA [17]	77.25
Backbone + Pose Encoder	80.44
Backbone + Ours	**83.81**

(e) **Interaction modeling methods**

Table 1. **Ablation Study on J-HMDB** We use a SlowFast-Resnet50 as video backbone and report our results in mAP. For *Backbone + Encoder* we directly use our AFM to fuse the pose and RGB features extracted from the pose encoder and video backbone.

The Sum fusion, refers to element-wise addition of the features. The $Concat$ fusion stands for channel-wise concatenation of the RGB and pose features. $WeightedSum$ yields a marginally higher mAP than the two previous fusion methods. However, it does not challenge our AFM. A better fusion method is the $Average$ fusion, which takes the average of the RGB and pose streams. As shown in table 1a, our AFM works better than the other approaches by virtue of its ability to enhance the combined features.

Late vs. Early Fusion: Late/early fusion refers to whether we fuse the two modalities before or after the Temporal Interaction module. Table 1c reports our results trying both structures. As we expected, temporal interaction works best when it's done on the full feature map, instead of features from each modality independently. It should also be more efficient since we only need one temporal interaction unit.

The Intra-Modality Aggragator (IMA): In section 1, we describe the use of the intra-modality component z_r for the RGB modality and z_p for the pose model. We notice that better feature selection is achieved when the network learns by itself how to do that. As shown in Table 1d, without the intra-modality aggregation module, important information would be wasted, holding back the model's performance.

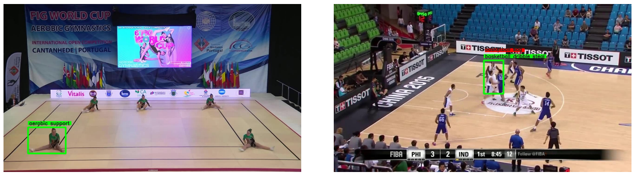
Interaction Modeling methods: To validate our interaction modeling scheme, we re-implement another interaction method found in the literature on top of the video backbone network. Table 1e contains results obtained with the bare backbone, with the backbone and our pose encoder, and the implementation of AIA [17]. For the *Backbone + Pose Encoder* framework, we directly fuse the outputs of the video backbone and the pose encoder. The table shows that our pose encoder is stronger than AIA, which aggregates person, object, and memory interaction.

2. Main Results

In Table 2, we report our results on the validation set of the MultiSports dataset. Our method outperforms other methods in terms of frame mAP with an IoU threshold of 0.5, and video mAP when the spatio-temporal tube threshold is 0.2. Note that these numbers we compare ours to are taken from the MultiSports paper [9]. The per-class frame AP is illustrated in Fig. 2.

Model	f@0.5	v@0.2	v@0.5
ROAD [16]	3.9	0.0	0.0
YOWO [8]	9.2	10.7	0.8
MOC [10]	25.2	12.8	0.6
MultiSports [9]	27.7	24.1	**9.6**
Ours	**33.3**	**27.8**	8.8

Table 2. **Comparison with the State-of-the-art.**



(a) Aerobic-related sports are easy to spot, especially when there is no overlap between the subjects.

(b) Basketball-related classes, on the other hand, are more challenging due to frequent other-persons-induced occlusion.

Figure 3. A correctly classified image on the left and an incorrectly classified one on the right.

2.1. Final Submission

We train our model on the training and validation data combined for the results submitted to the challenge's test server. The results are obtained with a single model (the one we describe throughout this report). No ensemble of models was used. Furthermore, our framework ended the development stage of the challenge at the top of the leaderboard with a v@0.10:0.90 score of 13.1. The f@0.5, v@0.2, and v@0.5 were 35.4, 28.9, and 9.6, respectively.

It is to be noted that we did not use additional data to improve our results on MultiSports. J-HMDB was used independently and only for ablation purposes.

3. Acknowledgments

We thank the National Center for High-performance Computing (NCHC) for providing computational and storage resources. We also want to extend our gratitude to the MultiSports team for providing the dataset and organizing this challenge.

References

[1] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 3

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3

[3] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 1, 3

[4] R Girshick, I Radosavovic, G Gkioxari, P Dollár, and K He. Detectron. *URL: https://github.com/facebookresearch/detectron*, 2011. 1, 3

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2

[6] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *International Conf. on Computer Vision (ICCV)*, pages 3192–3199, Dec. 2013. 3

[7] Vicky Kalogeiton, Philippe Weinzaepfel, Vittorio Ferrari, and Cordelia Schmid. Action tubelet detector for spatio-temporal action localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4405–4413, 2017. 3

[8] Okan Köpüklü, Xiangyu Wei, and Gerhard Rigoll. You only watch once: A unified cnn architecture for real-time spatiotemporal action localization. *arXiv preprint arXiv:1911.06644*, 2019. 4

[9] Yixuan Li, Lei Chen, Runyu He, Zhenzhi Wang, Gangshan Wu, and Limin Wang. Multisports: A multi-person video dataset of spatio-temporally localized sports actions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13536–13545, 2021. 3, 4

[10] Yixuan Li, Zixu Wang, Limin Wang, and Gangshan Wu. Actions as moving points. In *European Conference on Computer Vision*, pages 68–84. Springer, 2020. 4

[11] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 3

[12] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 3

[13] Junting Pan, Siyu Chen, Mike Zheng Shou, Yu Liu, Jing Shao, and Hongsheng Li. Actor-context-actor relation network for spatio-temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 464–474, 2021. 1

[14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015. 1, 3

[15] Hongje Seong, Junhyuk Hyun, and Euntai Kim. Video multitask transformer network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 1

[16] Gurkirt Singh, Suman Saha, Michael Sapienza, Philip HS Torr, and Fabio Cuzzolin. Online real-time multiple spatiotemporal action localisation and prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3637–3646, 2017. 4

[17] Jiajun Tang, Jin Xia, Xinzhi Mu, Bo Pang, and Cewu Lu. Asynchronous interaction aggregation for action detection. In *European Conference on Computer Vision*, pages 71–87. Springer, 2020. 1, 3, 4

[18] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. 3

[19] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 3d human pose estimation with spatial and temporal transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11656–11665, 2021. 2