HIT: Holistic Interaction Transformer for Action Detection

Gueter Josmy Faure
National Tsing Hua University

Min-Hung Chen
Microsoft

Shang-Hong Lai
National Tsing Hua University

Abstract

*Actions are all about how we interact with the environment, including other people, objects, and ourselves. In this paper, we propose a novel multi-modal **H**olistic **I**nteraction **T**ransformer Network (**HIT**) that leverages the largely ignored, but critical hand and pose information essential to most human actions. The proposed **HIT** network is a comprehensive bi-modal framework that comprises an RGB stream and a pose stream. 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 glued using an Attentive Fusion Mechanism (AFM). Finally, we extract cues from the temporal context to better classify the occurring actions using cached memory. Our method significantly outperforms previous approaches on the J-HMDB, UCF101-24, and MultiSports datasets. We also achieve competitive results on AVA.*

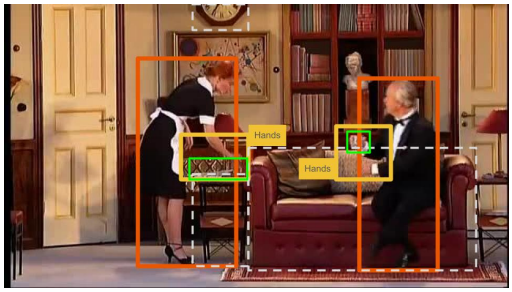


Figure 1: **Intuition**. This figure exemplifies how essential hand features are for detecting actions. Both persons in the frame are interacting with objects. Still, the instance detector fails to detect those very objects the persons are interacting with (green boxes) and, instead, picks the unimportant ones (dashed grey boxes). However, capturing the hands and everything in between (yellow boxes) gives the model a better idea of the actions being performed by the actors (red boxes); *"lift/pick up"* (left) and *"carry/hold (an object)"* (right)

1. Introduction

Spatio-temporal action detection is the task of recognizing actions in space and in time. In this regard, it is fundamentally different and more challenging than plain action detection, whose goal is to label an entire video with a single class. A sound spatio-temporal action detection framework aims to deeply learn the information in each video frame to correctly label each person in the frame. It should also keep a link between neighboring frames to better understand activities with continuous properties such as *"open"* - *"close"* [1, 5, 14, 30, 40]. In recent years, more robust frameworks have been introduced that explicitly consider the relationship between the spatial entities [28, 42] since if two persons are in the same frame, they are likely to be interacting with each other. However, using only person features is insufficient for capturing object-related action (e.g., *volleyball spiking*). Others try to understand the relationship not only between persons on the frame but also their surrounding objects [26, 39]. These methods have two main shortcomings. First, they only rely on objects with high detection confidence which might result in ignoring important objects that may be too small to be detected or unknown to the off-the-shelf detector. For example, in Figure 1, none of the objects the actors are interacting with are detected. Secondly, these models struggle to detect actions related to objects not present in the frame. For instance, consider the action *"point to (an object)"*. It is possible that the object the actor is pointing at is not in the current frame.

Figure 1 illustrates one of our motivations for undertaking this research. Most humans' actions are contingent on what they do with their hands and their poses when executing specific actions. The person on the left is *"picking up/lifting (something)"* which is not noticeable even by humans. Still, our model is able to capture this action since we consider the person's hand features and the pose of the subject (the bending position is typical of someone picking up something). A similar issue occurs with the person on the right who is *"sitting and holding (an object)"*. The man is holding a cup, but the object detector does not find the

object, probably because it is very small or highly transparent. Using hand features, our model implicitly focuses on these challenging objects.

Our proposed Holistic Interaction Transformer (HIT) network uses fine-grained context, including person pose, hands, and objects, to construct a bi-modal interaction structure. Each modality comprises three main components: person interaction, object interaction, and hand interaction. Each of these components learns valuable local action patterns. We then use an Attentive Fusion Mechanism to combine the different modalities before learning temporal information from neighboring frames that help us better detect the actions occurring in the current frame. We perform experiments on the J-HMDB [13], UCF101-24 [35], Multi-sports [18] and AVA [10] datasets, and our method achieves state-of-the-art performance on the first three while being competitive with the SOTA methods on AVA.

The main contributions in this paper can be summarized as follows:

- We propose a novel framework that combines RGB, pose and hand features for action detection.
- We introduce a bi-modal Holistic Interaction Transformer (HIT) network that combines different kinds of interactions in an intuitive and meaningful way.
- We propose an Attentive Fusion Module (AFM) that works as a selective filter to keep the most informative features from each modality, and an Intra-Modality Aggregator (IMA) for learning useful action representations within the modalities.
- Our method achieves state-of-the-art performance on three of the most challenging spatio-temporal action detection datasets.

2. Related Work

2.1. Video Classification

Video classification consists in recognizing the activity happening in a video clip. Usually, the clip spans a few seconds and has a single label. Most recent approaches to this task use 3D CNNs [1, 5, 6, 40] since they can process the whole video clip as input, as opposed to considering it as a sequence of frames [30, 38]. Due to the scarcity of labeled video datasets, many researchers rely on models pretrained on ImageNet [1, 41, 47] and use them as backbones to extract video features. Two-stream networks [5, 6] are another widely used approach to video classification thanks to their ability to only process a fraction of the input frames, striking a good balance between accuracy and complexity.

2.2. Spatio-Temporal Action Detection

In recent years, more attention has been given to spatio-temporal action detection [5, 7, 17, 28, 39]. As the name (spatio-temporal) suggests, instead of classifying the whole video into one class, we need to detect the actions in space, i.e., the actions of everyone in the current frame, and in time since each frame might contain different sets of actions. The most recent works on spatio-temporal action detection use a 3D convolution network backbone [27, 42] to extract video features and then crop the person features from the video features either using ROI pooling [8] or ROI align [12]. Such methods discard all the other potentially useful information contained in the video.

2.3. Interaction Modeling

What if the spatio-temporal action detection task really is an interaction modeling task? In fact, most of our everyday actions are interactions with our environment (e.g., other persons, objects, ourselves) and interactions between our actions (for instance, it is very likely that"open the door" is followed by "close the door"). The interaction modeling idea spurs a wave of research about how to effectively model interaction for video understanding [28, 39, 42].

Most researches in this area use the attention mechanism. [25, 51] propose Temporal Relation Network (TRN), which learns temporal dependencies between frames or, in other words, the interaction between entities from adjacent frames. Other methods further model not just temporal but spatial interactions between different entities from the same frame [26, 39, 42, 48, 52]. But the choice of entities for which to model the interactions differs by model. Rather than using only human features, [28, 45] chose to use the background information in order to model interactions between the person in the frame and the context. They still crop the persons' features but do not discard the remaining background features. Such an approach provides rich information about the persons' surroundings. However, while the context says a lot, it might say too much and induce noise.

In an attempt to be more selective about the features to use, [26, 39] first pass the video frames through an object detector, crop both the object and person features, and then model their interactions. This extra layer of interaction provides better representations than standalone human interaction modeling models and helps with classes related to objects such as *"work on a computer"*. Nevertheless, they still fall short when the objects are small (i.e., undetected by the object detection model), faraway (not in the current frame).

2.4. Multi-modal Action Detection

Most recent action detection frameworks use only RGB features. The few exceptions such as [10, 34, 36, 37] and

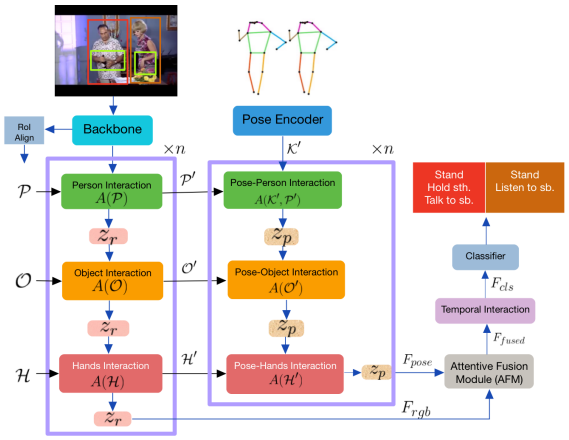


Figure 2: **Overview of our HIT Network.** On top of our RGB stream is a 3D CNN backbone which we use to extract video features. Our pose encoder is a spatial transformer model. We parallelly compute rich local information from both sub-networks using person, hands, and object features. We then combine the learned features using an attentive fusion module before modeling their interaction with the global context.

[29] use optical flow to capture motion. [37] employs an inception-like model and concatenate RGB and flow features at the $Mixed4b$ layer (early fusion) whereas [10] and [36] use an I3D backbone to separately extract RGB and flow features, then concatenate the two modalities just before the action classifier. While skeleton-based action recognition has been around for a while now [2, 11, 24], as far as we know, no previous works have tackled skeleton-based action detection.

In this paper, we propose a bi-modal approach to action detection that employs visual and skeleton-based features. Each modality computes a series of interactions, including person, object, and hands, before being fused. A temporal interaction module is then applied to the fused features to learn global information regarding neighboring frames.

3. Proposed Method

In this section, we provide a detailed walk-through of our approach. 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. This comprehensive activity understanding scheme helps us achieve superior action detection performance.

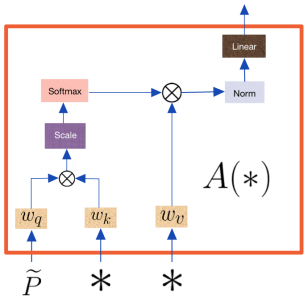


Figure 3: **Illustration of the Interaction module.** $*$ refers to the module-specific inputs while $\widetilde{P}$ refers to the person features in $A(\mathcal{P})$ or the output of the module that comes before $A(*)$.

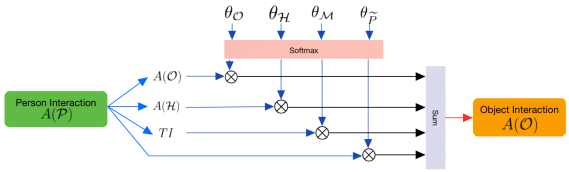


Figure 4: **Illustration of the Intra-Modality Aggregator.** Features from one unit to the next are first augmented with contextual cues then filtered.

This section is organized as follows: we first describe the entity selection process in section 3.1. In section 3.2, we elaborate on the RGB modality before introducing its pose counterpart in section 3.3. Further, in section 3.4, we explain our Attentive Fusion Module (AFM) and then the Temporal Interaction Unit (Section 3.5). The section ends with a discussion on our Intra-Modality Aggregator (IMA).

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 $2S+1$ 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 Figure 2.

3.1. 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 move-

ments (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 [9] for human pose detection and create a bounding box encircling the location of the person's hands. Following the state-of-the-art methods, [39], [32], [28], we use Faster-RCNN [31] to compute object bounding box proposals. We use the person bounding boxes from [16] at inference time. The video feature extractor is a 3D CNN backbone network[5], and the pose encoder is a lightweight spatial transformer inspired by [50]. We apply ROIAlign [12] to trim the video features and extract person and local context features (hands and objects).

3.2. The RGB Branch

The RGB branch comprises three main components, as shown in Figure 2. 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 3. 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. It is like asking "how can these particular features help detect what the target person is doing?". 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 (see Figure 4). 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. A more detailed discussion on z_r is provided in the supplementary material.

3.3. 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 [50].

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

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

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

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

$A(\mathcal{K}', \mathcal{P}')$ 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.

3.4. 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. We then reduce the magnitude of the output feature by using the projection matrix Θ_{fused}. Table 5a in our ablation study validates the superiority of our fusion mechanism compared to other fusion types used in the literature.

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

3.5. 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 $2S + 1$. Inspired by

[39], the memory cache contains the person features extracted by the video backbone. F_{fused} inquires $\mathcal{M}$ as to which of the neighboring frames contains informative features, then absorbs them. 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.

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

Finally, the classification head g is composed of two feed-forward layers with relu activation, and the output layer.

$$\hat{y} = g(F_{cls}) \tag{6}$$

4. Experiments

We perform experiments on four challenging action detection datasets: J-HMDB [13], UCF101-24 [35], MultiSports [18] and AVA [10]. The implementation details described below relate to the J-HMDB and UCF101-24 datasets. We refer the reader to the supplementary materials for details on how we train MultiSports and AVA.

4.1. Datasets

The **J-HMDB dataset** [13] has 21 action classes and up to 55 clips per class. The dataset totaled 31,838 annotated frames with a resolution of 320x240. Each video clip contains 15 to 40 frames and is trimmed to contain a single action. To be on the same page with other methods, we report frame and video mAP results on split-1 of the dataset. The IoU threshold for frame mAP is 0.5, the same as other methods in our comparison table.

UCF101-24 is a subset of the UCF101 [35] dataset suitable for Spatio-temporal action detection. It contains 24 action classes and 3207 untrimmed videos with human bounding boxes annotated frame-by-frame. The classes mainly relate to sports activities. Following the state-of-the-art methods, we test our model on the first split of the dataset and report frame mAP with an IoU threshold of 0.5 and video mAP with two different thresholds for Spatio-temporal tube overlap.

The **MultiSports** dataset [18] 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 has a duration of around 22 seconds.

AVA [10] version 2.2 consists of 430 15-minutes videos sampled from YouTube. For each video in the dataset, 900 frames are annotated with human bounding boxes and labels. The dataset contains 80 class labels divided into pose action (14), person-person interaction (49), and person-object interaction (17) classes. Following the standard practice, we report the frame mAP for 60 of the 80 classes with a spatial IoU threshold of 0.5.

4.2. Implementation Details

Person and Object Detector: We extract keyframes from each video in the dataset and use detected person bounding boxes from [16] for inference. As object detector, we employ Faster-RCNN [31] with ResNet-50-FPN [21, 46] backbone. The model is pretrained on ImageNet [3], and fine-tuned on MSCOCO [22].

Keypoints Detection and Processing: For keypoints detection, we adopt a pose model from Detectron [9]. The authors use a Resnet-50-FPN backbone pretrained on ImageNet for object detection and fine-tuned on MSCOCO keypoints using precomputed RPN [31] 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 bounding boxes from [16] (during 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.

Backbone: We employ SlowFast networks [5] as our video backbone. Our experiments and ablation study use SlowFast with a ResNet-50 instantiation pretrained on Kinetics-700 [1]. For AVA and MultiSports, we use the more powerful SlowFast-Resnet-101 pretrained on K700 as the video backbone.

Training and Evaluation: The input videos are sampled 32 frames per clip, with $\alpha = 4$ and $\tau = 1$, meaning the SlowFast backbone has a temporal stride of 4 for the slow path while the fast path takes the entire 32 frames as input. During training, random jitter augmentation is applied to the ground-truth human bounding boxes. For object boxes, we use the ones with detection score ≥ 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. The network is trained on the J-HMDB dataset for 7K iterations, with the first 700 iterations serving as linear warm-up. No weight decay was used. We use SGD as optimizer and a batch size of 8 to train the model on one 11GB GPU. We train on the UCF101-24 dataset for 50k iterations, adopting linear warm-up during the first 1k iterations. The starting learning rate of 0.0002 is reduced by a factor of 10 at iterations 25k and 35k. During inference, we predict action labels for human bounding boxes provided by [16] for both datasets.

4.3. Comparison with State-of-the-Art Methods

In Tables 1 and 2, we compare our results with other methods on the challenging J-HMDB and UCF101-24 datasets, respectively. Our method registers significant

Model	input	f@0.5	v@0.2	v@0.5
ACT [15]	V + F	65.7	74.2	73.7
Li et. al [19]	V	—	76.1	74.3
TacNet [34]	V + F	65.5	74.1	73.4
MOC [20]	V + F	70.8	77.3	77.2
AVA [10]	V + F	73.3	—	78.6
PCSC [36]	V + F	74.8	82.6	82.2
HISAN [29]	V + F	76.7	85.9	84.0
ACRN [37]	V + F	77.9	—	80.1
Context rcnn [45]	V	79.2	—	—
TubeR [49]	V + F	—	87.4	82.3
Ours	V	**83.8**	**89.7**	**88.1**

Table 1: **Frame and video-level comparison with the state-of-the-art methods on J-HMDB.** We use a SlowFast-Resnet50 as video backbone and report our results in mAP. Our model outperforms state-of-the-art methods on both frame mAP and video mAP metrics.

Model	input	f@0.5	v@0.2	v@0.5
ACT [15]	V + F	67.1	77.2	51.4
ACDNet [23]	V + F	70.9	—	—
TacNet [34]	V + F	72.1	77.5	52.9
HISAN [29]	V + F	73.7	80.4	49.5
MOC [20]	V + F	78.0	82.8	53.8
AIA [39]	V	78.8	—	—
PCSC [36]	V + F	79.2	84.3	61.0
TubeR [49]	V + F	83.2	83.3	58.4
ACAR [28]	V	84.3	—	—
Ours	V	**84.8**	**88.8**	**74.3**

Table 2: **Comparison with the State-of-the-art methods on UCF101-24**. Like other methods in our comparison table, we evaluate frame mAP on split 1 with an IoU threshold of 0.5 and video mAP with thresholds of 0.2 and 0.5.

Model	f@0.5	v@0.2	v@0.5
ROAD [33]	3.9	0.0	0.0
YOWO[16]	9.2	10.7	0.8
MOC [20]	25.2	12.8	0.6
MultiSports [18]	27.7	24.1	**9.6**
Ours	**33.3**	**27.8**	8.8

Table 3: **Comparison with the State-of-the-art on MultiSports.** Our model significantly outperforms the other methods on two metrics.

gains compared to the state-of-the-art methods both in terms of frame and video mAP. Such a performance demonstrates our bi-modal framework's ability to capture more diverse clues about human actions by taking a closer look at the human's pose and environment.

Model	Pretrain	frame mAP
SlowFast, R-101+NL [5]	K600	29.0
X3D-L[4]	K600	29.4
AIA [39]	K700	32.3
Object Transformer[43]	K600	31.0
Beyond Short Clips[48]	K700	31.6
ACAR[28]	K700	33.3
MeMViT [44]	K700	33.5
*TubeR [49]	IG + 400	**33.6**
Ours	K700	32.6

Table 4: **Comparison with the State-of-the-art on AVA v2.2.** Our model has comparable results compared to the SOTA methods.

In Table 3, we report our results on 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 2. As Table 4 shows, we achieve competitive results on the most challenging fine-grained action detection dataset (AVA). With ACAR [28] using pretrained features as memory and TubeR [49] using a backbone pretrained on the IG + K400 dataset, the only comparable method that outperforms ours is MeMViT [44]. Overall, our results on four action detection datasets exhibit the generalization capabilities of our method.

4.4. Ablation Study

We perform ablation experiments on the J-HMDB dataset to illustrate the effectiveness of our model and its constituents. 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 5b, a two-layer setting improves the mAP by more than 4% compared to having just one, while adding a third induces overfitting. This is due to our method blending a lot of information within one layer. Therefore, for the remaining experiments, we report results using two layers. By two layers, we mean that the RGB sub-network is repeated twice, and so is the pose sub-network.

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,

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 Temporal Interaction	82.16
Before Temporal Interaction	**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[39]	77.25
Backbone + Pose Encoder	80.44
Backbone + Ours	**83.81**

(e) **Interaction modeling methods**

Hands	RGB	Pose	mAP
			58.85
	✓		79.11
	✓	✓	79.62
		✓	80.19
✓	✓		80.82
✓		✓	80.90
✓	✓	✓	**83.81**

(f) **The importance of each modality and the hand features**

$A(\mathcal{H})$	$A(\mathcal{O})$	TI	mAP
✓			81.44
	✓		78.86
		✓	79.73
✓		✓	79.36
✓	✓		80.23
	✓	✓	79.62
✓	✓	✓	**83.81**

(g) **Importance of individual interaction units**

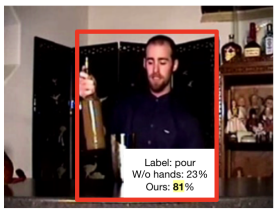
Table 5: **Ablation Study on J-HMDB** We use a SlowFast-Resnet50 as video backbone and report our results in mAP. *backbone* refers to the video *backbone* followed by the action classifier. For *Backbone + Encoder* we directly use our AFM to fuse the pose and RGB features extracted from the pose encoder and video backbone, then apply the action classifier.

$Concat$, $WeightedSum$, and $Average$.

In the Sum fusion, refers to element-wise addition of the features. Such a method yields the worst result since we end up with significantly magnified result. The $Concat$ fusion stands for channel-wise concatenation of the RGB and pose features. It is slightly better than the Sum fusion but still falls short of the desired outcome since it does not enhance the results. $WeightedSum$ yields a marginally higher mAP than the two previous fusion methods. However, it does not challenge our AFM since our intra-modality aggregator (IMA: z_r, z_p) already selects the best features from each modality. A better fusion method is the $Average$ fusion, which takes the average of the RGB and pose streams. Such a fusion approach solves the shortcomings of Sum but does not enhance the resulting feature. As shown in table 5a, 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 5c 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 3, 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 5d, without the intra-modality aggregation module, important information would be wasted, holding back the model's performance. Therefore, we present the features from each interaction unit and let the IMA component choose and aggregate information as it pleases.

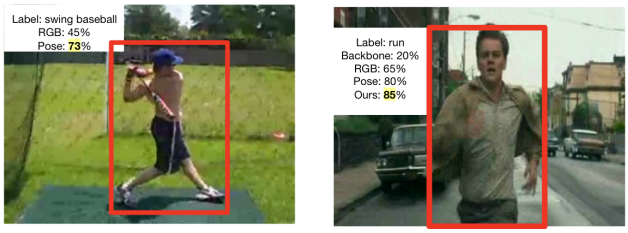


(a) Hand features is essential for detecting the action class "pour".

(b) The "pick up" class has a clear pose signature.

Figure 5: **Qualitative results on the hand and pose features importance.**

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 5e contains results obtained with the bare backbone, with the backbone and our pose encoder, and the implementation of AIA [39]. For the *Backbone + Pose En-*



(a) Another action with clear pose signature.

(b) A neutral class. The accuracy increases as a we plug in more modules.

Figure 6: **The modalities' importance.**

coder 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. This proves that a person's pose contains rich information about what the person is doing. Such a result also confirms that pose information works well, whether used as a supplement or as a standalone network.

The importance of each modality and the hand features: In Table 5f, we present a detailed ablation of the different building blocks of our model. Using only the RBG or pose modality, the action detection mAP jumps 20 points compared to the backbone and keeps increasing from there. Hand features excluded, the pose-only model is stronger than the RGB-only model, which confirms our assumption that hand features are more valuable to the RGB sub-network since the pose sub-network implicitly contains hand information (hand keypoints). That being said, the pose-only model still benefits from hand features, as evidenced by the mAP increase from 80.19% without hands to 80.90% with them. The RGB-only model registers a higher gain when hands are added (79.11% versus 80.82%). These experiments underline the importance of hand interaction for action detection. With all these components pulling the strings, the model trained with both modalities with hand interaction registers the highest accuracy. Such an outcome indicates the harmony between all parts of our framework as well as their independent contributions.

Importance of Different Types of Interactions: Since our framework is composed of three auxiliary types of interaction units, we wanted to quantify their different contributions. While it is feasible, we did not consider removing $A(\mathcal{P})$ in this ablation since our model is person-centric. As Table 5g shows, hands interaction ($A(\mathcal{H})$) alone yields higher accuracy than either $A(\mathcal{O})$ or TI. It is also better than any other combination. We suspect this is a byproduct of our Intra-Modality Aggregator not having enough features to work with. Without other interaction types as enforcers, $A(\mathcal{O})$ returns the lowest accuracy. However, when paired with hand interaction, the model's accuracy jumps from 78.86% to 80.23%, outlining their complementarity. This ablation proves that the previously ignored hand features provide essential information for accurate action detection.

4.5. Qualitative Results

To further assess our framework's performance and understand what it "sees", in Figure 5, we present qualitative results on select frames from the J-HMDB dataset with action classes we consider as hand-related. Figure 5a illustrates how using hand features can help for classes related to hands, such as "pour". A model without hands would struggle to detect such an action due to the poor disparity between the background and the actor, as evidenced by the low confidence score it gives to the correct action label. Our model easily spots the action since it, among other things, focuses on the person's hand. In Figure 5b, the pose-only model is even more powerful than the complete bi-modal framework due to the person's bending, which is a strong pose feature. Even though the action of "picking up something" is hand-related, hand detection features for this frame might be noisy because of the blurriness of the frame. Such a result demonstrates the subtleties our pose modality is able to identify.

Figure 6a confirms that our pose-only model does an exceptional job classifying actions with typical pose signatures. The person uses his hand to "swing a baseball"; however, the pose signature is still more evident than the RGB hand features. Figure 6b further confirms the significance of each modality of our model. For a neutral class like "run", the model's confidence keeps increasing as we add the modalities, reaching its peak with both RGB and pose combined. With such an outcome, we can argue that the different modalities of our network work in tandem to help us achieve superior video action detection performance.

5. Conclusion

Learning the nature of interactions between person and other instances is vital for detecting actions. In this paper, we demonstrate that the choice of instances is at least as important as the instance modeling framework. In our Holistic Interaction Transformer Network, we integrate previously ignored instances such as person pose and hands and construct a bi-modal framework to model and aggregate interactions effectively. After learning modality-specific interaction features, the two streams of our model are combined using an Attentive Fusion Mechanism. We also present detailed ablations validating our design choices. Our results on three public action detection benchmarks demonstrate our framework's superiority over the state-of-the-art methods.

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.

[2] Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13359–13368, 2021.

[3] 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.

[4] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 203–213, 2020.

[5] 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.

[6] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[7] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 244–253, 2019.

[8] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

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

[10] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6047–6056, 2018.

[11] Pranay Gupta, Anirudh Thatipelli, Aditya Aggarwal, Shubh Maheshwari, Neel Trivedi, Sourav Das, and Ravi Kiran Sarvadevabhatla. Quo vadis, skeleton action recognition? *International Journal of Computer Vision*, 129(7):2097–2112, 2021.

[12] 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.

[13] 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.

[14] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2012.

[15] 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.

[16] 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.

[17] Chao Li, Qiaoyong Zhong, Di Xie, and Shiliang Pu. Collaborative spatiotemporal feature learning for video action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7872–7881, 2019.

[18] 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.

[19] Yuxi Li, Weiyao Lin, Tao Wang, John See, Rui Qian, Ning Xu, Limin Wang, and Shugong Xu. Finding action tubes with a sparse-to-dense framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11466–11473, 2020.

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

[21] 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.

[22] 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.

[23] Yu Liu, Fan Yang, and Dominique Ginhac. Acdnet: An action detection network for real-time edge computing based on flow-guided feature approximation and memory aggregation. *Pattern Recognition Letters*, 145:118–126, 2021.

[24] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 143–152, 2020.

[25] Chih-Yao Ma, Asim Kadav, Iain Melvin, Zsolt Kira, Ghassan AlRegib, and Hans Peter Graf. Attend and interact: Higher-order object interactions for video understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6790–6800, 2018.

[26] Joanna Materzynska, Tete Xiao, Roei Herzig, Huijuan Xu, Xiaolong Wang, and Trevor Darrell. Something-else: Compositional action recognition with spatial-temporal interaction networks. In *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, pages 1049–1059, 2020.

[27] Jingcheng Ni, Jie Qin, and Di Huang. Identity-aware graph memory network for action detection. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 3437–3445, 2021.

[28] 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.

[29] Rizard Renanda Adhi Pramono, Yie-Tarng Chen, and Wen-Hsien Fang. Hierarchical self-attention network for action localization in videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 61–70, 2019.

[30] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *proceedings of the IEEE International Conference on Computer Vision*, pages 5533–5541, 2017.

[31] 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.

[32] 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.

[33] 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.

[34] Lin Song, Shiwei Zhang, Gang Yu, and Hongbin Sun. Tacnet: Transition-aware context network for spatio-temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11987–11995, 2019.

[35] 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.

[36] Rui Su, Wanli Ouyang, Luping Zhou, and Dong Xu. Improving action localization by progressive cross-stream cooperation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12016–12025, 2019.

[37] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 318–334, 2018.

[38] Lin Sun, Kui Jia, Dit-Yan Yeung, and Bertram E Shi. Human action recognition using factorized spatio-temporal convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4597–4605, 2015.

[39] 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.

[40] 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.

[41] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.

[42] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 284–293, 2019.

[43] Chao-Yuan Wu and Philipp Krahenbuhl. Towards long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1884–1894, 2021.

[44] Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13587–13597, 2022.

[45] Jianchao Wu, Zhanghui Kuang, Limin Wang, Wayne Zhang, and Gangshan Wu. Context-aware rcnn: A baseline for action detection in videos. In *European Conference on Computer Vision*, pages 440–456. Springer, 2020.

[46] 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.

[47] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning for video understanding. *arXiv preprint arXiv:1712.04851*, 1(2):5, 2017.

[48] Xitong Yang, Haoqi Fan, Lorenzo Torresani, Larry S Davis, and Heng Wang. Beyond short clips: End-to-end video-level learning with collaborative memories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7567–7576, 2021.

[49] Jiaojiao Zhao, Yanyi Zhang, Xinyu Li, Hao Chen, Bing Shuai, Mingze Xu, Chunhui Liu, Kaustav Kundu, Yuanjun Xiong, Davide Modolo, et al. Tuber: Tubelet transformer for video action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13598–13607, 2022.

[50] 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.

[51] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 803–818, 2018.

[52] Jiaming Zhou, Kun-Yu Lin, Haoxin Li, and Wei-Shi Zheng. Graph-based high-order relation modeling for long-term action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8984–8993, 2021.