

Hand-Object Interaction Reasoning

Jian Ma
 University of Bristol
 jian.ma@bristol.ac.uk

Dima Damen
 University of Bristol
 dima.damen@bristol.ac.uk

Abstract

This paper proposes an interaction reasoning network for modelling spatio-temporal relationships between hands and objects in video. The proposed interaction unit utilises a Transformer module to reason about each acting hand, and its spatio-temporal relation to the other hand as well as objects being interacted with. We show that modelling two-handed interactions are critical for action recognition in egocentric video, and demonstrate that by using positionally-encoded trajectories, the network can better recognise observed interactions. We evaluate our proposal on EPIC-KITCHENS and Something-Else datasets, with an ablation study.

1. Introduction

Different from general actions (e.g. jumping), object interactions involve actors influencing objects (e.g. playing an instrument or kicking a ball). Of particular interest to this work is hand-object interactions (HOI) which feature regularly in the activities of daily living. HOIs include one-handed (e.g. “open drawer”) as well as two-handed interactions (e.g. “open bottle”), and many interactions include tools that extend our hands’ abilities (e.g. cutting a vegetable requires a knife). However, most video understanding methods aim to recognise both actions and interactions alike as general video datasets involve a mix of classes [4, 18]. Recently, a handful of large-scale datasets [6, 7, 17] that focus on HOIs have fueled works that specifically reason about interactions.

Recent progresses in interaction reasoning have been driven by the success of object detectors, e.g. [31]. Due to the datasets used, previous works [1, 15, 29, 35] detect the person in the middle frame of the video or all the objects that appear in the video. This means that trajectories of interactions are not explicitly emphasised.

In this work, we focus on the interaction between hands and objects, through their trajectories that encode motion and positions, thus discriminating between different inter-

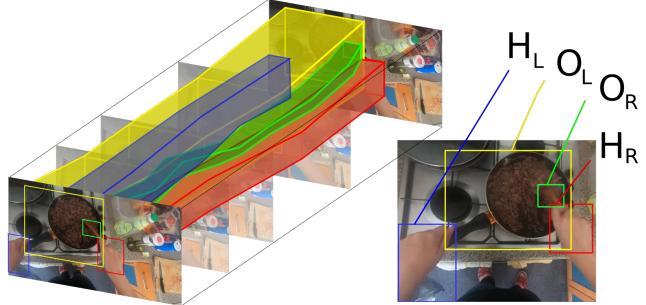


Figure 1: Illustration of Spatio-Temporal HOI reasoning for the action of stirring food in the pan. Both hands play critical roles. The left hand (H_L , in blue) steadies the pan, which we refer to as the left object (O_L , in yellow) given its direct interaction with the left hand. The right hand (H_R , in red) holds the wooden spoon (O_R , in green). While H_L , O_L are steady over time, the red and green trajectories (H_R , O_R) demonstrate the stirring motion relative to the pan.

actions (Fig. 1). We propose an **Interaction Reasoning Network** (IRN) that jointly reasons about interactions between both hands and active objects.

We propose encoders that learn hand-object and hand-hand interactions and decoders that enrich this learned interaction with action representation knowledge. We automatically detect hands and active objects, i.e. those with which the hand interacts, using the approach from [32], and link these detections over time to form trajectories with pooled features from a spatio-temporal backbone. We then reason about pairwise interactions, distinguishing left from right hand interactions. The proposed framework is trained end to end.

Our contributions: (i) this paper proposes to separately reason about interactions between the left and right hands with their corresponding objects in HOI through an encoder-decoder transformer module; (ii) to the best of the authors’ knowledge, the proposed method is the first work that uses the trajectories of hands and objects to enrich the relational representation of interactions; (iii) we showcase the importance of this proposed reasoning on the large-scale

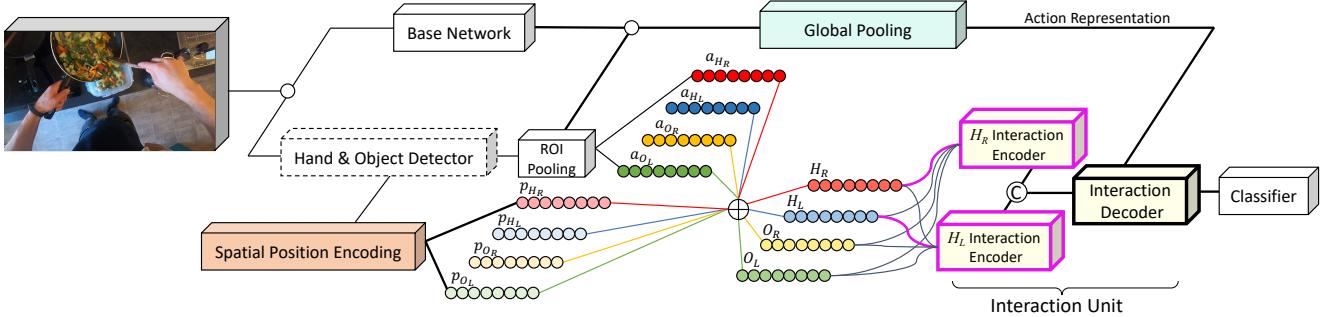


Figure 2: Proposed Architecture for Interaction Reasoning Network (IRN). A base 3D ConvNet extracts temporal-spatial features. These are ROI pooled based on hand-object detections and combined with positional encoding to form trajectories. The trajectories are fed into interaction unit to reason about hand-object relations. A decoder combines the action representation and two-hand encoders for classification (Fig. 4 for details) . \circ is for multiple outputs, \oplus is for summation and C for concatenation. Dashed box for ‘Hand & Object Detector’ highlights frozen weights. Thicker lines highlight backpropagation pathways.

egocentric HOI dataset EPIC-KITCHENS-100 [6] as well as on the crowd-sourced HOI dataset Something-Else [26].

2. Related Work

Action Recognition. In recent years, the success of deep learning in computer vision has promoted the rapid development of action recognition models. From the 2D ConvNets [8, 23, 40, 44, 45] and multi-stream networks [4, 12, 33] to 3D ConvNets [10, 11, 21, 37] and transformer-based networks [3, 27], these models have progressively improved the understanding of actions. Of relevance to our model, SlowFast network [11] is a 3D convolutional model that can combine the spatial features with temporal information from two rates of sampling the input video. SlowFast results remain competitive particularly for HOI datasets we analyse. These approaches form the base for interaction reasoning which we review next.

Interaction Recognition. Recently, interest in actor-centric video understanding employed increasingly reliable person detectors, to localise actors in movies [18] or players in sports [34]. Many works [5, 13, 14, 15, 16, 24, 26, 29, 32, 39, 43, 46] utilise an object detector (*e.g.* Faster R-CNN) to locate bounding boxes of human or objects for input into a relational module. However, the detector is likely to yield object proposals that are not relevant to the interaction due to the lack of annotations for training. Instead, [16] learns to predict the location of related objects based on the appearance of actors. Similarly, [5, 14] introduce pairwise streams for interaction patterns to encode the spatial relative locations of human and objects. Based on pairwise streams, human poses are considered for modeling HOIs in [24, 39]. Additionally, Graph Convolutional Network (GCN) is another way to explore image-based object interactions [13, 46].

Different from image-based methods, the motion is also particularly important in video analysis. Given person detections from the middle frame of a video, in [15], detections are pooled as a query to attend to the whole frame’s 3D features, in a transformer encoder block. Longer-term reasoning is proposed in [43] by learning contextual information through short-term person feature banks and long-term feature banks from non-local blocks [41]. Apart from the contextual interaction in the temporal dimension, [1] study relational interactions between objects via training a Gate Recurrent Unit with pairing current and previous frames. Similarly, [42] regard video as a graph of objects, conducting interaction recognition reasoning on a graph network. Moreover, [26] generalise the performance to unseen actions, decomposing each action into a verb, subject, and one or more goals and proposing the Something-Else dataset for exploring hand-object interactions. Detections over time are combined through the Hungarian algorithm to track the detected persons and objects. However, detected objects may not be part of the interaction, which introduces noise in all but the simplest scenes. Besides, in [26], additional expensive-to-collect annotations of objects are required to train the model.

IRN is inspired by [1, 15], but we focus on hands as actors, thus requiring to model two-hand interactions including hand-to-hand interactions as well as hand-to-object interactions. Different from [26], we do not require additional annotations and instead use automatic detection of hands and active objects from [32]. This allows us to reason about interactions in busy scenes efficiently. We detail our method next.

3. Methodology

In this section, we firstly give an overall of the Interaction Reasoning Network (IRN) architecture (visualised in

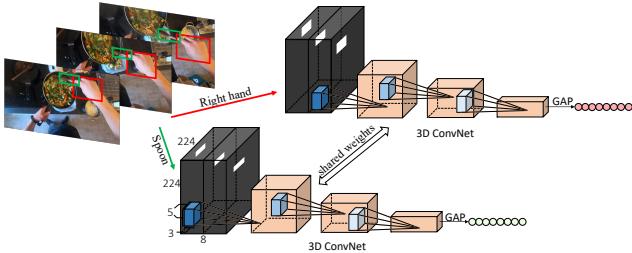


Figure 3: Spatial Position Encoder. Detections, per object, over a sequence are represented by binary maps. A 3-layer 3D ConvNet produces per-frame spatial encoding.

Fig. 2). We then discuss detections and the spatial position encoder followed by details of our proposed interaction unit (IU).

IRN captures the relationships between hands and active (i.e. action-relevant) objects. We model interactions through i) encoders that focus on hand-object and hand-hand interaction; and ii) decoders that enrich the action representation, trained jointly. We formulate the interaction unit as:

$$\begin{aligned} E &= e(H, O, P; \omega_e), \\ I &= d(F, E; \omega_d), \end{aligned} \quad (1)$$

where ω_e and ω_d represent the parameters of encoders $e()$ and decoders $d()$. The encoder aims to model the hand H and interacting object O representations along with their spatial position encoding P . The encoder’s output E and global action representation F are fed into decoders.

3.1. Detections and Positional Encoding

We use a pretrained hand and object detector, which we run per frame, and distinguish four bounding box detections of $\{b_{H_L}, b_{H_R}, b_{O_L}, b_{O_R}\}$ where H/O mean hand and object, L/R denote the side, left or right. We pool a 3D ConvNet to extract frame-level hands and objects features, using RoI average pooling as in [1, 15]. Given a layer in the backbone with C channels, we extract a feature a per detection of size \mathbb{R}^C by RoI pooling, of which we have $\{a_{H_L}, a_{H_R}, a_{O_L}, a_{O_R}\}$.

In addition to the features, the absolute positions of hands and objects offer significant information to distinguish interactions. We thus propose to use spatial positional representation p for each detection, such that $\{p_{H_L}, p_{H_R}, p_{O_L}, p_{O_R}\}$. Inspired by [20], we consider a binary map for each detection b . These show the absolute position and scale of an object in the image. We learn position (and scale) encodings from this binary input and use the consecutive binary maps to capture the positional/scale changes. Note that while we use local 3D convolution to model motion, we have the position encoding per frame to correspond to the bounding box b . We learn a weights-shared 3D ConvNet as shown in Fig. 3.

As interactions focus on the temporal evolution of the relationships between hands and objects, we combine features and their positional encodings to form trajectories. We use standard summation but also ablate this against concatenation. We represent each trajectory $H = (a^1 + p^1, \dots, a^T + p^T)$ over T frames, and similarly for objects. We thus have four trajectories $\{H_L, H_R, O_L, O_R\}$, which form the input for our interaction unit. When hands or objects are not present in a frame or are not detected, we set the feature a to 0 and the binary map to blank. We showcase experimentally that our method can recover from missed detections. Additionally, some trajectories might be missing altogether. We consider the presence as well as the absence of hands and objects as informative evidence for interaction reasoning.

We next describe how we can reason about interactions using these trajectories.

3.2. Interaction Unit

In order to reason about HOIs, we consider a pair of trajectories (H, O) where H is the actor - a left/right hand and O is the object with which the hand interacts. Importantly O can be the other hand or an object. As Fig. 4 demonstrates, we have up to 3 interactions per actor/hand. For the left hand, these would be (H_L, O_L) , (H_L, O_R) and (H_L, H_R) . We intuitively describe these using the example from Fig. 1 on stirring food in the pan.

- (H_L, O_L) captures the left hand holding the pan. The hand gesture and relative positions of hand and pan would be captured using this interaction.
- (H_L, O_R) captures the left hand versus the spoon as it stirs through the food in the pan held by the left hand.
- (H_L, H_R) captures the absolute positions and gestures of both hands as one holds the pan and the other stirs the food.

We first describe our encoders. These are concatenated and passed to stacked decoders described after.

Interaction Encoder. We explain the encoder for one interaction pair namely (H_L, O_L) .

$$\begin{aligned} Q_{H_L, O_L} &= q(H_L), \\ K_{H_L, O_L} &= k(O_L), \\ V_{H_L, O_L} &= v(O_L), \end{aligned} \quad (2)$$

where q, k, v all linearly project $\mathbb{R}^{T \times C}$ input to \mathbb{R}^N . Fig. 4 shows the module of left-hand interactions. The encoder is then a residual attention unit:

$$E'_{H_L, O_L} = \sigma \left(\frac{Q_{H_L, O_L} K_{H_L, O_L}}{\sqrt{N}} \right) V_{H_L, O_L} + Q_{H_L, O_L}, \quad (3)$$

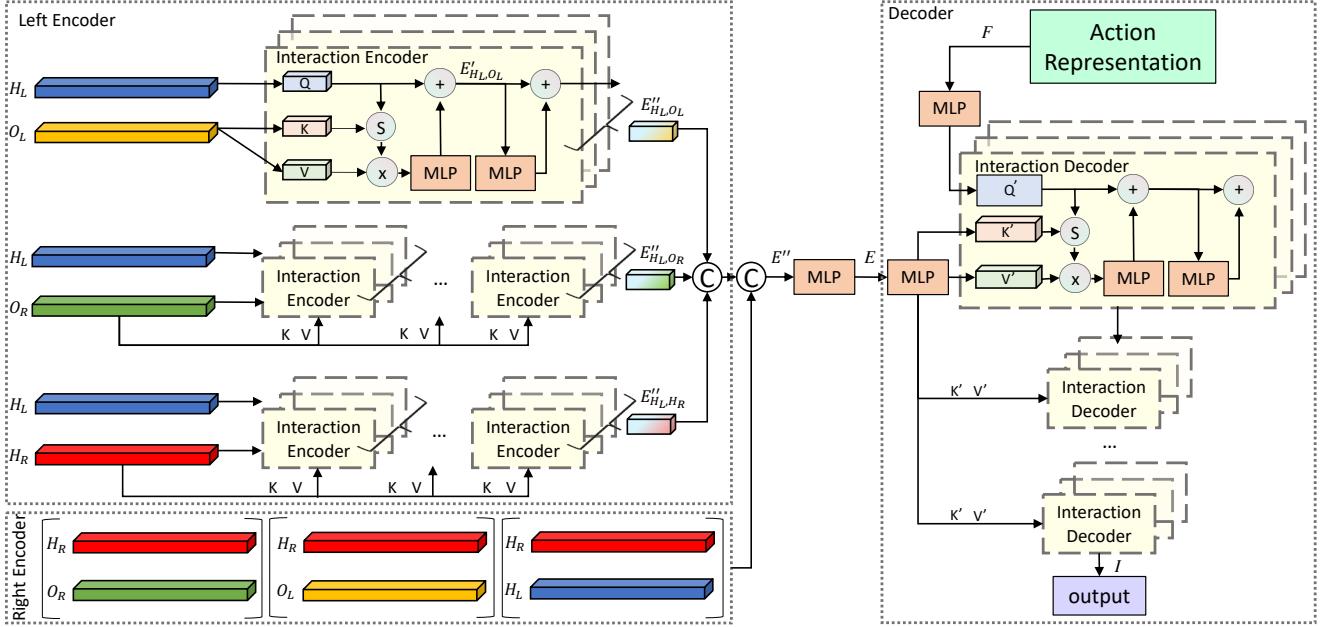


Figure 4: The Architecture of proposed Interaction Unit (IU). This module includes encoders and decoders. Encoders reason about hand-object interactions by multi-head transformers with hands as query and objects as key and value. Encoders outputs from left and right hands are concatenated and fed to decoders. Decoders model the interaction with action representation.

where E'_{H_L,O_L} is interaction representation between actor H_L and object O_L and σ is the softmax operator. Following [38], we add a linear feedforward network $FFN()$:

$$E''_{H_L,O_L} = \delta(FFN(E'_{H_L,O_L})) + E'_{H_L,O_L}, \quad (4)$$

where δ is the dropout operation and $E''_{H_L,O_L} \in \mathbb{R}^N$ is the encoded representation for (H_L, O_L) .

Similarly, E''_{H_L,O_R} and E''_{H_R,O_L} are computed for the left hand (see Fig. 4 Left Encoder) as well as three encoders for E''_{H_R} . We concatenate all outputs to form overall encoding $\mathbf{E}'' \in \mathbb{R}^{6N}$. Subsequently, the dimension of \mathbf{E} is reduced from $6N$ to M . We set M to the size of action representation F by a linear projection.

Interaction Decoders. Having reasoned about all pairwise interactions using the encoder, we use \mathbf{E} to enrich the action representation, as in [15]. The pipeline of the decoder is similar to the encoder. Specifically, the features \mathbf{E} from the encoder is projected to key $K' \in \mathbb{R}^M$ and value $V' \in \mathbb{R}^M$. Different from the initial transformer [38], we remove the self attention from the decoder due to action representation features $F \in \mathbb{R}^M$ pooled on temporal dimension at backbone features. Instead, we directly map F linearly to $Q' \in \mathbb{R}^M$. Similarly, we adopt dropout, a feedforward network as well as residual connections, like in Eq. 4, to learn decoder's output $I \in \mathbb{R}^M$.

At last, we stack the multi-head interaction encoders and decoders. The output I is fed to a fully connected layer, and

trained to classify actions using standard cross entropy loss. During tracking, we backpropagate through all components of the IRN including IU and convolutional positional encoding, as well as the base network. The weights of the detector remain frozen.

4. Experiments and Results

In this section, we experimentally evaluate the model on two datasets featuring hand-object interactions from egocentric and crowd sourced videos respectively.

4.1. Datasets, Evaluation and Implementation Details

EPIC-KITCHENS-100 [6] is the largest video dataset in egocentric vision. We report on the full dataset but the conduct ablation study on a subset of the validation set. The subset was collected by the first participant (P01) and contains 5,509 / 885 action segments for the training and validation set, respectively. We use a fixed random seed and a single run during ablations to ensure results are directly comparable.

Something-Else [26] is based on [17] proposing a new split for novel verb-noun combinations in the test set. Active objects have been densely annotated for training by [26] – however hands are not annotated for side L/R and accordingly the active objects are not associated with a hand. We use this proposed split without the spatial annotations, instead using automatic, potentially noisy, detections.

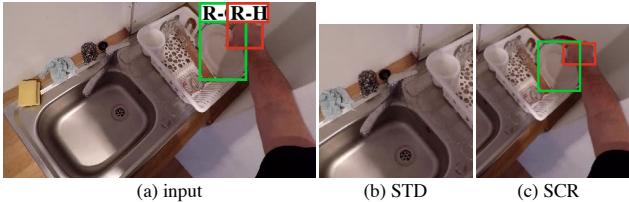


Figure 5: Visualisation of STD and SCR for data augmentation. The action is “take plate”.

Evaluation We use evaluation metrics proposed for both dataset - Top-1/5 accuracy for verb, noun and action classes. For Something-Else, we use a single output classifier.

Implementation Details. The hand-object detector [32] aims to find links between hands and interacting objects by optimising an offset vector. It was trained on 100 days dataset that contains both first-person and third-person videos. We use the publicly available trained weights, and detect frame-level hands and objects with the confidence threshold of 0.5.

We use SlowFast [11] R-50 8x8 due to its performance on both datasets. The input videos are 32-frame clips, where we sample $T = 8$ frames with a temporal stride $\tau = 8$ for the slow pathway, and 32 frames for the fast pathway. We pool hands and objects features $C = 640$ based on the third block of SlowFast as when the layer goes deeper, the features get sparser. We set $N = 5120$ used in the encoder and $M = 2304$ so that it matches the the globally pooled feature size of the last block of SlowFast. For the interaction unit, 16 heads are used in the multi-head attention and 3-layer encoders and decoders are implemented.

For EPIC-KITCHENS-100, the backbone is pre-trained on the training set and $e()$ and $d()$ are trained from random initialisation. We train for 24 epochs with learning rate 0.001 using SGD with 0.0001 weight decay and 0.9 momentum, the learning rate is decayed by the factor of 10 at epochs 10 and 20. For Something-Else the backbone is pretrained on Kinetics-400 [22]. Similarly, the network was optimised by SGD with initial learning rate 0.01 dropped at epochs 12 and 18, 0.0001 weight decay, 0.9 momentum for 20 epochs.

It is important to note that we changed the random cropping typically used for data augmentation. This is because randomly cropping the image in EPIC-KITCHENS-100 results in hands and objects being frequently cropped out of the main frame. As we show in Fig. 5, the standard STD crops the right hand and plate out. This significantly harms the interaction unit. In egocentric footage, hands can be in the bottom half of the image or towards a corner. To avoid this while maintaining the power of augmentation, we first randomly scale (**S**) frames to $H' \times W'$, then crop (**C**) frames to $H' \times H'$ and finally resize (**R**) frames to the target resolution

Method	Ens.	Obj.	Top-1	Top-5
STIN [26]	✗	✓	37.2	62.4
I3D+STIN [26]	✗	✓	48.2	72.6
CAF [30]	✗	✓	52.3	78.9
STRG [42]	✗	✗	52.3	78.3
IRN (Ours)	✗	✗	52.9	80.8
I3D-STIN [26]	✓	✓	51.5	77.1
STRG-STIN [26]	✓	✓	56.2	81.3
CACNF [30]	✓	✓	56.9	82.5

Table 1: Results on Something-Else Datasets. +: jointly trained. -: trained separately. Ens.: Ensemble. Obj.: use manual object labels.

I3D	SlowFast	IRN	Top-1	Top-5
✓			46.8	72.2
✓		✓	47.5	73.8
	✓		52.2	80.3
✓	✓		52.9	80.8

Table 2: Results of different backbones on Something-Else.

224 × 224. We refer to this augmentation approach as SCR, which we use in all our experiments on EPIC-KITCHENS-100. The two data augmentations perform comparably on Something-Else dataset as the action is typically at the centre of the image. We thus use the standard random cropping data augmentation for this dataset.

4.2. Something-Else Dataset Results

Tab. 1 shows the performance of our method outperforms SotA reasoning approaches STIN, STRG and CAF. Specifically, we outperform STIN Combined with I3D, when trained jointly or separately. Notably, our class-agnostic IRN is superior to methods using object labels among non-ensemble models. We also report ensemble methods. These metods are not directly comparable but our model remains competitive.

Importantly, we showcase that our model particularly benefits from the usage of a SlowFast backbone. We evaluate our interaction units with different backbones in the Tab. 2. We get the best performance when combining SlowFast as a base network with our interaction unit, but also report improvement on the I3D backbone. In the Something-Else dataset, we find that only 39.63% of clips have both left and right hands, while 50.11% and 10.26% of clips have only one hand or no hands respectively. When no actor/hand is detected, our interaction unit is likely to struggle. Despite this, our proposed method can enrich the representation for interaction reasoning.

4.3. EPIC-KITCHENS-100 Dataset Results

As this dataset is egocentric, i.e. participants are using a wearable camera, more actions involve both hands making it more suitable to assess our proposal. To manage the size,

(H_L, O_L)	(H_L, O_R)	(H_L, H_R)	(H_R, O_R)	(H_R, O_L)	(H_R, H_L)	Top-1
✗	✗	✗	✓	✓	✗	42.37
✗	✗	✗	✓	✓	✓	43.28
✗	✓	✓	✓	✓	✓	42.60
✓	✗	✓	✓	✓	✓	43.28
✓	✓	✗	✓	✓	✓	42.94
✓	✓	✓	✗	✗	✗	42.82
✓	✓	✓	✗	✓	✓	41.47
✓	✓	✓	✓	✗	✓	44.07
✓	✓	✓	✓	✓	✗	41.69
✓	✓	✓	✓	✓	✓	44.52

Table 3: Ablation study for Interaction Components.

Det.	H_R	Act. Rep.	H_L	Objects	Top-1
middle [15]	✓	✓			43.05
duplicate [29]	✓	✓			44.07
trajectory	✓	✓			42.37
trajectory	✓	✓		✓	43.28
trajectory	✓	✓	✓	✓	44.52

Table 4: Comparison to prior works without trajectory representations as well as active object trajectories.

we conduct an ablation on a selected subset¹.

Interaction Components. To evaluate the contribution of the various encoding interactions, we ablate the results by removing one at a time, as well as left/right hand encoders in Tab. 3. We first note that removing right hand interaction encoders (row 6) results in a larger drop than left hand (row 2). This is anticipated with most participants being right-handed. Similarly, the largest drop is associated with removing the encoder of H_R, O_R , which is critical for one-handed interactions (row 7) followed by H_R, H_L (row 9) which is critical for hand-only interactions (e.g. wash hands). The encoder with the least contribution is that for H_R, O_L (row 8), as it is probably compensated by the other pairwise encoders.

Trajectory. We evaluate the importance of using trajectories. Previous work [15] only uses one actor (person) of the middle frame (*middle*) along with video action representation and a recent work [29] detects persons in the middle frame and then duplicate these detections across time (*duplicate*). We compare these options to our proposed approach that encodes object trajectories as well as two-handed interactions. The best performance is achieved when using the complete trajectory of detections, including both hands and objects. For more details, please refer to the supplementary.

Spatial Positional Encoding. We adopt two ways to fuse positional information with visual features of hands or objects. As we expected in Tab. 5, spatial position encoding *SPE* improves the performance. Compared to the network without *SPE* (*none* in Tab. 5) or concatenations of *SPE*. This is due to the sparsity of positional features. Concatenating lots of zeros to the visual features may introduce

SPE	Top-1	Act. Rep.	Top-1	Det. Rep	Top-1
none	43.39	none	31.98	MLP	43.95
concat	42.71	concat	43.73	CNN	44.52
sum	44.52	decoder	44.52		

Table 5: Ablation study for Spatial Position Encoding.

Table 6: Ablation study for Action Representation.

Table 7: Ablation study for Detection Representation.

noise. Our proposed approach to *sum* the positional encoding yields the best action performance.

Action Representation. In this ablation study, we evaluate the importance of globally pooled action representation. First, We remove decoders and train the network with only interaction features, i.e. *none*. Tab. 6 shows that results drop significantly if the network does not use the globally-pooled action representation. We also concatenate the hand-object interaction features with action representation, in a late-fusion fashion. What stands out in the Tab. 6 is that the Top-1 action result of the decoder surpasses concatenation.

Detection Representation. In all other experiments, we have used RoI pooling to get feature representations of our hands' and objects' detections. Instead, inspired by popular MLPs [36], we also explore using MLP projection directly from image patches. We crop the input image given bounding boxes b and resize to a fixed size by bilinear interpolation. The fixed size patches are flatten, and projected using an MLP. Hence, representations of any detections can be learned by a parameter-shared MLP. Notably, these features only contain the spatial information, without any motion cues as they are projected per image. We compare performances of MLPs and pooled 3D-CNN as the features extractors. Tab. 7 shows that the accuracy of 3D-CNN pooling is superior to MLP projection, indicating that temporal features are significant for video-based hand-to-object interactions.

Detections Feature Representation. In all other experiments, we have used RoI pooling to get feature representations of our detections. As presented in Sec 3, we also explore using MLP projection. Inspired by popular MLPs [36], we compare performances of MLPs and ConvNets as to get the visual features of our detections. Tab. 7 shows that MLP has better modelling capabilities than ConvNets only for verb recognition. However, ConvNets are more advantageous for noun recognition and their accuracy is far better than MLPs by about 2.7%. Hence, ConvNets are more accurate in action recognition. All previous and subsequent results thus use 3D ConvNet RoI pooling to represent the visual features of hand and object detections.

We next report results on the validation set of EPIC-

¹We'll make the subset available for direct comparisons

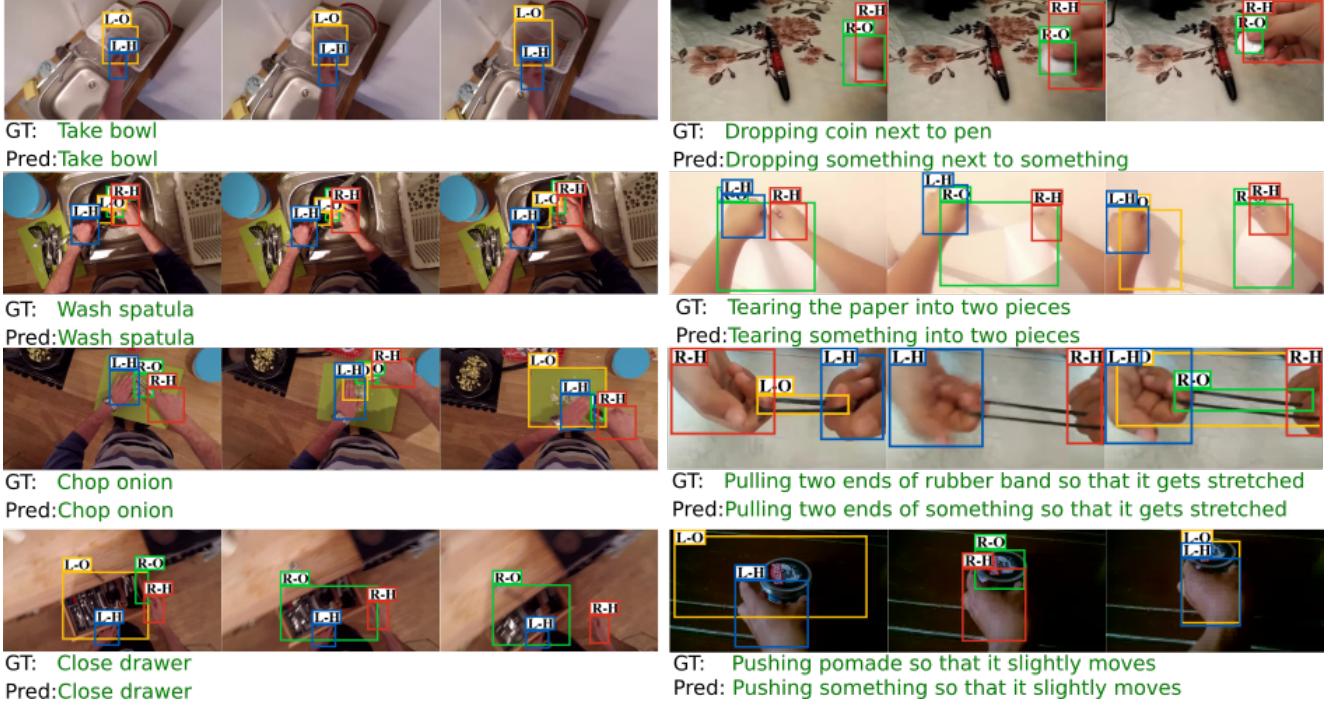


Figure 6: Qualitative results of correctly-recognised interactions from EPIC-KITCHENS-100 (col 1) and Something-Else (col 2). L/R indicate the side and H/O denote hands and objects. GT and Pred are Ground Truth and Prediction.

Method	DataAug.	Top-1 Accuracy (%)		
		Verb	Noun	Act.
Chance [6]	STD	10.42	1.70	0.51
IRN (Ours)	STD	60.94	43.97	31.97
TSN [6]	STD	60.18	46.03	33.19
SlowFast [6]	SCR	63.64	48.58	36.76
IRN (Ours)	SCR	63.68	48.94	37.11

Table 8: Quantitative results on Validation set. STD and SCR denote standard and our data augmentation strategies.

KITCHENS-100. In Tab. 8, our method *IRN* improves over published baselines. Adding the interaction unit to SlowFast improves results for verb, noun and action top-1 accuracy by 0.04%, 0.36% and 0.35%, respectively. We explain the data augmentation strategies in supplementary, where STD performs random cropping often cropping the hands out of the image. Comparing row 2 and row 5, we demonstrate that the random cropping harms the performance of IRN.

4.4. Qualitative Results

We demonstrate the results on clips containing only one hand, or both hands in Fig. 6. It illustrates correct action recognition examples from both datasets. It is also important to highlight that our method recovers from partial failures in automatic detections of hands and objects, as in rows 3 and 4. IRN is robust to two types of detec-

tion errors: (a) incorrect active objects detection; (b) incorrect side of hand and objects. Specifically, row 3 illustrates “chop onion” and “pull two ends of something so that it gets stretched”. The onion is not detected in several frames due to occlusion. Similarly, the knife and rubber are missed in several frames along the trajectory. For row 4 (col 1) the drawer switches between being a left-object or a right object, while in row 4 (col 2) the hand sides are swapped in various frames. Both errors (a) and (b) have little impact on our method, due to our usage of trajectories and the attention mechanism that selects the relevant hand and object representations to reason about interactions.

Moreover, we show failure cases in Fig. 7. The main reason for failure is undetected or unobserved hands throughout the videos (row 1). In both examples, the hands are not visible throughout. Row 2 (col 1) shows that our focus on both hands might result in detecting another concurrent action, paying more attention to the object in the other hand (salt). In row 2 (col 2) the method fails to recognise the interaction with small movement. In row 3, the less frequent action of replacing the bin bag is incorrectly mistaken as the frequent throwing action. A clear limitation of our approach is evident in row 3 (col 2) where the person is checking the time on their hand watch. Evidently, the watch is never recognised as the interacting object, as it is always part of the hand detection.

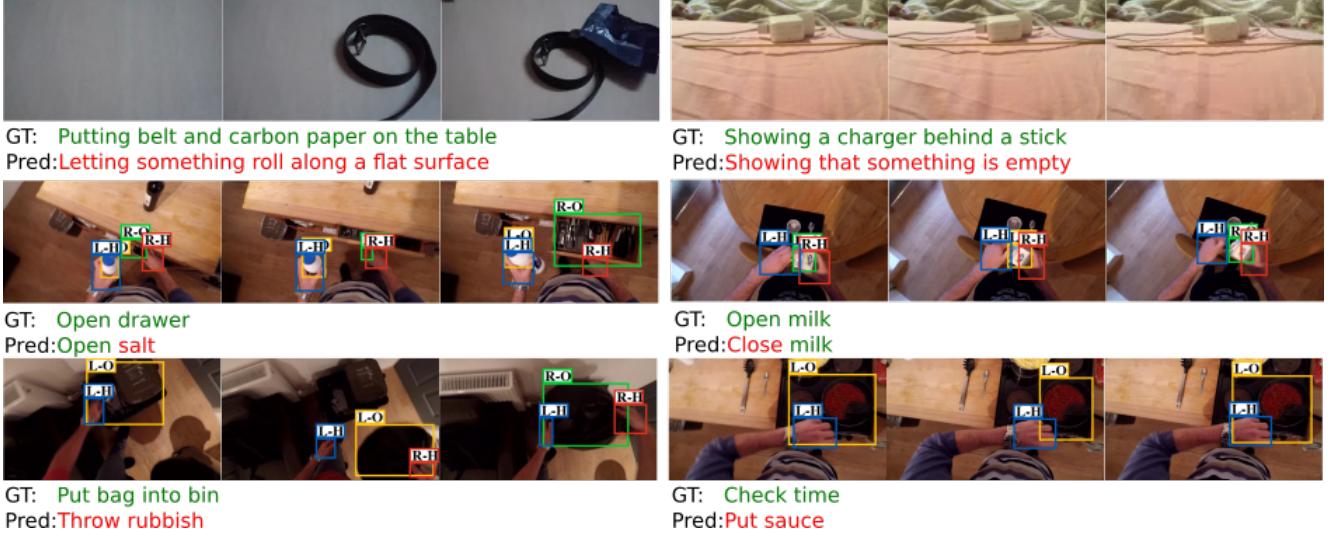


Figure 7: Failure cases of Something-Else (row 1) and EPIC-KITCHENS-100 (rows 2, 3).

5. Conclusion

In this paper, we present a framework for hand-object interaction reasoning that separately attends to actors (hands) and interacting objects, through encoders, as well as action representation which includes contextual information through a decoder. We present results on two hand-object interaction datasets, demonstrating generality and competitive performance, with an ablation study.

Acknowledgements. This work used public datasets and is supported by EPSRC UMPIRE (EP/T004991/1).

References

- [1] Fabien Baradel, Natalia Neverova, Christian Wolf, Julien Mille, and Greg Mori. Object level visual reasoning in videos. In *ECCV*, 2018. [1](#), [2](#), [3](#)
- [2] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.
- [3] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, 2021. [2](#)
- [4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017. [1](#), [2](#)
- [5] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *WACV*, pages 381–389, 2018. [2](#)
- [6] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision. *CoRR*, abs/2006.13256, 2020. [1](#), [2](#), [4](#), [7](#)
- [7] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *ECCV*, 2018. [1](#)
- [8] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, pages 2625–2634, 2015. [2](#)
- [9] Haoqi Fan, Yanghao Li, Bo Xiong, Wan-Yen Lo, and Christoph Feichtenhofer. Pyslowfast. <https://github.com/facebookresearch/slowfast>, 2020.
- [10] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *CVPR*, June 2020. [2](#)
- [11] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, pages 6202–6211, 2019. [2](#), [5](#)
- [12] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *CVPR*, pages 1933–1941, 2016. [2](#)
- [13] Chen Gao, Jiarui Xu, Yuliang Zou, and Jia-Bin Huang. Drg: Dual relation graph for human-object interaction detection. In *ECCV*, pages 696–712, 2020. [2](#)
- [14] Chen Gao, Yuliang Zou, and Jia-Bin Huang. ican: Instance-centric attention network for human-object interaction detection. In *BMVC*, 2018. [2](#)
- [15] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *CVPR*, pages 244–253, 2019. [1](#), [2](#), [3](#), [4](#), [6](#)
- [16] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions. In *CVPR*, pages 8359–8367, 2018. [2](#)
- [17] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The “something something” video database for learning and evaluating visual common sense. In *ICCV*, pages 5842–5850, 2017. [1](#), [4](#)

- [18] Chunhui Gu, Chen Sun, David A. Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, Cordelia Schmid, and Jitendra Malik. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *CVPR*, June 2018. 1, 2
- [19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017.
- [20] Md Amirul Islam, Sen Jia, and Neil DB Bruce. How much position information do convolutional neural networks encode? In *ICLR*, 2020. 3
- [21] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *TPAMI*, 35(1):221–231, 2012. 2
- [22] 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. 5
- [23] Chao Li, Qiaoyong Zhong, Di Xie, and Shiliang Pu. Collaborative spatiotemporal feature learning for video action recognition. In *CVPR*, pages 7872–7881, 2019. 2
- [24] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yanfeng Wang, and Cewu Lu. Transferable interactiveness knowledge for human-object interaction detection. In *CVPR*, pages 3585–3594, 2019. 2
- [25] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *IOC*, pages 7083–7093, 2019.
- [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 *CVPR*, pages 1049–1059, 2020. 2, 4, 5
- [27] Daniel Neimark, Omri Bar, Maya Zohar, and Dotan Asselman. Video transformer network. *arXiv preprint arXiv:2102.00719*, 2021. 2
- [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 *CVPR*, pages 464–474, 2021.
- [29] Junting Pan, Siyu Chen, Zheng Shou, Jing Shao, and Hongsheng Li. Actor-context-actor relation network for spatio-temporal action localization. 2020. 1, 2, 6
- [30] Gorjan Radevski, Marie-Francine Moens, and Tinne Tuytelaars. Revisiting spatio-temporal layouts for compositional action recognition. *arXiv preprint arXiv:2111.01936*, 2021. 5
- [31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, pages 91–99, 2015. 1
- [32] Dandan Shan, Jiaqi Geng, Michelle Shu, and David Fouhey. Understanding human hands in contact at internet scale. In *CVPR*, 2020. 1, 2, 5
- [33] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 2014. 2
- [34] Khurram Soomro and Amir R Zamir. Action recognition in realistic sports videos. In *the Computer Vision in Sports*, pages 181–208. 2014. 2
- [35] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. In *ECCV*, pages 318–334, 2018. 1
- [36] Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, et al. Mlp-mixer: An all-mlp architecture for vision. *arXiv preprint arXiv:2105.01601*, 2021. 6
- [37] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015. 2
- [38] A Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 4
- [39] Bo Wan, Desen Zhou, Yongfei Liu, Rongjie Li, and Xuming He. Pose-aware multi-level feature network for human object interaction detection. In *ICCV*, pages 9469–9478, 2019. 2
- [40] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, pages 20–36, 2016. 2
- [41] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803, 2018. 2
- [42] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *ECCV*, pages 399–417, 2018. 2, 5
- [43] Chao-Yuan Wu, Christoph Feichtenhofer, HaoQi Fan, Kaiming He, Philipp Krähenbühl, and Ross Girshick. Long-Term Feature Banks for Detailed Video Understanding. In *CVPR*, 2019. 2
- [44] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *CVPR*, pages 4694–4702, 2015. 2
- [45] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *ECCV*, pages 803–818, 2018. 2
- [46] Penghao Zhou and Mingmin Chi. Relation parsing neural network for human-object interaction detection. In *ICCV*, pages 843–851, 2019. 2