Interaction Region Visual Transformer for Egocentric Action Anticipation

Debaditya Roy\textsuperscript{1}, Ramanathan Rajendiran\textsuperscript{1}, and Basura Fernando\textsuperscript{1,2}

\textsuperscript{1}Institute of High-Performance Computing, Agency for Science, Technology and Research, Singapore  
\textsuperscript{2}Centre for Frontier AI Research, Agency for Science, Technology and Research, Singapore

Abstract

Human-object interaction (HOI) and temporal dynamics along the motion paths are the most important visual cues for egocentric action anticipation. Especially, interaction regions covering objects and the human hand reveal significant visual cues to predict future human actions. However, how to incorporate and capture these important visual cues in modern video Transformer architecture remains a challenge. We leverage the effective MotionFormer that models motion dynamics to incorporate interaction regions using spatial cross-attention and further infuse contextual information using trajectory cross-attention to obtain an interaction-centric video representation for action anticipation. We term our model InAViT which achieves state-of-the-art action anticipation performance on large-scale egocentric datasets EPICKITCHENS\textsuperscript{100} (EK100) and EGTEA Gaze+\textsuperscript{\textsuperscript{\textsuperscript{1}}}. On the EK100 evaluation server, InAViT is on top of the public leader board (at the time of submission) where it outperforms the second-best model by 3.3\% on mean-top5 recall. The code is available\textsuperscript{\textsuperscript{1}}.

1. Introduction

In egocentric action anticipation, the model needs to predict the immediate next human action that is going to happen, usually 1 second into the future \cite{11}. Action anticipation is a challenging task due to various reasons such as the uncertainty in future actions, diversity of execution of actions, and the complexity of human-object interactions presented when executing those actions. A way to reduce the uncertainty in predicting the next action is to develop models that may be able to infer information about probable future objects and interactions that will be used in the future. As the observed and next actions are causally related, it is highly likely that information about observed interactions may possess possible cues about future actions.

\textsuperscript{1}\url{https://github.com/LAHAPrject/InAViT}

Some prior works use hand-object interactions to exploit those cues for action anticipation \cite{48, 32}, yet do not exploit the visual changes in human and object appearance caused by the execution of actions. We hypothesize that modeling of change in the appearance of regions containing objects and hands may reveal vital information about the probable execution of future actions. The work in \cite{32} focuses on predicting manually annotated interaction hotspots without accounting for the specific objects associated with the interaction. However, it is challenging to automatically recognize the most important image regions that reveal what is going to happen next from egocentric videos. We develop an approach that attends to the most informative patches among the entire interaction region in every observed frame and make use of those modeled changes in appearances to predict future actions.

In particular, we propose to model the interaction region containing the hands and the interacted objects. We model the interaction regions as how hand and object appearance change due to the execution of the action and use those changes in appearance. Furthermore, we infuse the interaction regions with contextual cues from the background of the current action to obtain additional visual information about the area in which the interactions are performed. Finally, we propose an effective way to incorporate context-infused hand-object interaction regions to a video Transformer to create a richer interaction-centric video representation. Specifically, we use effective MotionFormer \cite{43} as it aggregates important dynamic information along implicitly determined motion paths. This simple yet effective idea improves the action anticipation performance outperforming many dedicated video Transformer models.

Leveraging on the MotionFormer \cite{43}, we design a spatio-temporal visual transformer denoted as human-object Interaction visual transformer (InAViT) for action anticipation that refines image patches from the interaction regions in every frame which we term as interaction tokens. The interaction tokens are obtained from the refined object and human tokens and the refinement is influenced by ob-
jects, hands, and the anticipated action. We incorporate the visual context of the interaction’s surroundings into the interaction tokens by proposing a trajectory cross-attention mechanism based on trajectory attention [43]. Finally, we infuse interaction tokens into the observed video to build an interaction-centric video representation for effective action anticipation. InAViT provides a way to extract visual changes in interaction regions across observed frames using trajectory cross-attention while in [48], human and object visual features are simply concatenated frame-wise to represent interactions.

Human-object interactions in actions are expressed as relations in spatio-temporal graphs in [24, 55, 36, 53, 42]. We are inspired by these approaches, especially the progression of human-object relationships for action recognition tasks [42]. We model interaction regions explicitly as spatio-temporal visual changes in hands and objects rather than implicitly as edges between hands and objects [42]. Furthermore, authors in [22, 60] make use of only object dynamics, and do not treat human hands as a separate entity [36, 42]. On the other hand, our interaction-centric video representation recognizes hands as a separate entity from objects that affect visual change on objects and vice-versa. The change in the hand’s visual appearance when interacting with objects also gives a clue about the observed action. Hence, modeling interaction regions by capturing visual changes in both hands and objects is better than modeling the change in objects alone [22]. Therefore, we postulate that interaction-centric representations are better suited for action anticipation than object-centric approaches.

In summary, our contributions are twofold: (1) We propose a novel human-object interaction module that computes appearance changes in objects and hands due to the execution of the action and models these changes in appearances to refine the video representations of Video Transformer models for effective action anticipation. (2) On the EK100 evaluation server, our model InAViT is top of the public leaderboard and outperforms the second-best model by 3.3% on mean-top5 recall. Leveraging on MotionFormer, we also obtain massive improvement in EGTEA Gaze+ dataset.

2. Related Work

Anticipating human actions has gained interest in the research community with large datasets [19, 38, 6] and innovative approaches [11, 51, 17, 56, 9, 20, 57, 52, 47, 1, 44, 58, 26, 13, 12]. In [11], rolling and unrolling LSTM (RU-LSTM) is proposed to predict the next action. The authors in [51] increase RU-LSTM’s temporal context using non-local blocks to combine local and global temporal context. In [17], spatio-temporal transformers called Anticipative Video Transformers (AVT) are proposed for action anticipation. In [56], MemViT extends AVT for long-range sequences by memory caching multiple smaller temporal sequences. RAFTformer [16] proposes a real-time action anticipation transformer using learnable anticipation tokens to capture global context trained using masking-based self-supervision. In [18, 40, 34, 41], present the problem of long-term anticipation (LTA). In [7], motion primitives called therbligs are introduced for decomposing actions which are then used for action anticipation. The authors in [49, 50] use goal representation for next action anticipation while [35] focuses on discovering intentions for LTA. The audio information is used to augment video representation for next-action anticipation in [62] and [39] uses only audio for LTA. Other approaches consider past and future correlation using Jaccard vector similarity [9], self-regulated learning [44], transitional model [37], and counterfactual reasoning [61].

There are a variety of approaches for Human-Object Interactions (HOIs) modeling in images [4, 15, 30, 14, 59, 5, 27, 28, 31] and videos [32, 29, 33, 25, 8]. In [29], human-object interaction regions are detected in videos using verb-object queries describing the action. Observed action labels are not available during testing in action anticipation and hence, we cannot predict the interaction region. In [25], relationships between humans and objects are modeled and verified using relationship labels. Relationship labels are not available for the observed video and so our model attends to all interactions and discovers their importance in predicting the next action.

Another related area is interaction hotspot prediction where future hand-trajectory and interaction spots need to be estimated. Interaction prediction approaches [32, 33] learn future hand motion distribution conditioned on the video representation using an encoder (LSTM [32] or Transformer [33]). Object interaction anticipation [45] requires predicting the next active object bounding box along with the next action. However, these approaches require explicit annotations of hand trajectories in future, object trajectories in the observed frames and location of interaction spots. Our interaction modeling approach obviates the need for hand and object trajectory annotations that are difficult to obtain in videos as shown in [10].

3. Preliminaries

3.1. Basic video representation

We extract a set of 3-D cuboids or video ”tokens” from a video as existing video transformers [2, 43, 3]. The set of all the video tokens from a single fixed length video-clip is denoted by $X \in \mathbb{R}^{THW \times d}$ wherein each of the $THW$ cuboids is linearly projected to a $d$-dimensional vector. Here, $T$ is the number of frames in the fixed-length video clip and $H, W$ is the number of vertical and horizontal patches respectively. Let $x_{st} \in \mathbb{R}^d$ denote a video token
from the set $X$ at spatial location $s \in \{1, \cdots, H \times W\}$ and temporal location $t \in \{1, \cdots, T\}$. Similar to [43], we add separate learnable positional encoding for spatial and temporal dimension for each video token denoted as $e_s^t \in \mathbb{R}^d$ and $e_t^t \in \mathbb{R}^d$, respectively. The resultant video token after spatial and temporal embedding is given as $x_{st} = x_{st} + e_s^t + e_t^t$. A classification token $x_{cls}$ is appended for anticipating the next action from $X$ resulting in $THW+1$ tokens in $\mathbb{R}^d$. We exclude the classification token hereafter for clarity.

3.2. Obtaining hand and object tokens

We obtain hand and object representations from video tokens $X$. We obtain object and hand bounding-boxes using Faster R-CNN [46]. In every frame, we use one bounding box for the hand and $N$ bounding boxes for objects closest to the hand. SORT algorithm is used over the detections to obtain sequences of detections where each sequence represents a hand or an object [42]. Now, given the video tokens corresponding to a frame $X_t$ and the bounding box of the hand $B_{h,t}$, we obtain a hand token $h_t \in \mathbb{R}^d$. We make use of RoIAlign [21] layer on $X_t$ to obtain hand region crops similar to [22]. We then use MLP and max-pooling to obtain the final hand representation or the hand token. We apply this to every frame to obtain $T$ hand tokens denoted by $H \in \mathbb{R}^{T \times d}$ where $H = [h_1, \cdots, h_T]$. Similarly, for each of object $i$, we obtain a $T$ object tokens $o_{i,1}, \cdots, o_{i,T}$ and we denote it as $O_i \in \mathbb{R}^{T \times d}$. In total, for the $N$ objects, we end up with $O = \mathbb{R}^{T \times N \times d}$ where $O = [O_1, \cdots, O_N]$.

4. Our method

4.1. Overview of our method

We hypothesize that in egocentric action anticipation, hands and objects play a key role in anticipating actions. Hands and objects change the appearance of other objects causing visible state changes, such as human cutting tomato with knife or human emptying a pan using spatula. Change in the state of the objects reveals cues about the possible next action. We capture these changes using newly designed interaction tokens by refining the original hand and object representations (tokens) with respect to each other.

As the objects affect the appearance of hand regions, we refine the hand tokens using all object tokens in the frame using Eq. (1). Similarly, as hand and objects affect the appearance of other objects when executing the action, we model this by refining every object token using hand tokens and other object tokens in the frame using Eq. (2).

$$\tilde{H} = \phi_H(H|O)$$

(1)

$$\tilde{O}_i = \phi_O(O_i|H, O_j) \forall j \neq i, i \in 1, \cdots, N,$$

(2)

Here $\phi_H$ and $\phi_O$ are attention-based functions that will be discussed in detail in Secs. 4.2.1 to 4.2.3. 

Refined object tokens for all objects are denoted as $\tilde{O} = [\tilde{O}_1, \cdots, \tilde{O}_N]$. Together, the refined hand and object tokens constitute the interaction tokens $I = [\tilde{H}, \tilde{O}]$.

We also hypothesize that the context or the background of the current action may provide useful information when predicting the next action. For example, picking a tomato next to the cutting board (context) informs that the next probable action is cut tomato. Therefore, we enrich the interaction tokens ($I$) using the information coming from the context and vice-versa. We use Trajectory Cross Attention function ($\phi_I$) inspired by [43] to obtain the context-infused interaction tokens $\tilde{I}$

$$\tilde{I} = \phi_I(I|X).$$

(3)

The final video representation is interaction-centric where we first concatenate context-infused interaction tokens to the video tokens. Then, we assimilate the interaction regions into the video using self-attention ($\phi_X$). We choose the refined tokens corresponding to the original video tokens $X$ as the interaction-centric video representation $X_I$

$$X_I = \phi_X([I, X]).$$

(4)

We predict the next action using the interaction-centric video representation with multiple layers of Trajectory Attention as in MotionFormer [43]

$$a_{next} = \phi(X_I).$$

(5)

Next action anticipation is defined as observing $1, \cdots, T_a$ frames and predicting the action that happens after a gap of $T_a$ seconds. It is important to note that a new action starts after $T_a$ seconds that is not seen in the observed frames. Our overall approach is shown in Fig. 1.

4.2. Human-Object interaction region modeling

Now we discuss how we implement Eq. (1) and Eq. (2). We model spatiotemporal interaction regions between hands and objects in three ways encapsulating different types of interaction information to obtain interaction tokens $I = [\tilde{H}, \tilde{O}]$. These three types of interaction modeling
Figure 2: Modeling interaction region tokens using Spatial Cross Attention. In every frame, hand tokens act as query and object tokens as key and value to compute refined hand tokens. Refined object tokens are computed with object token as query, and hand and other object tokens as key and values (not shown here to avoid clutter). Interaction tokens consist of refined hand and object tokens.

will be evaluated in the experiments. Reader may refer to the supplementary material section 1 for the common definition of cross-attention and self-attention. Next, we present three ways to obtain interaction tokens.

4.2.1 SCA: Spatial cross-attention

We implement Eq. (1) and Eq. (2) using spatial cross-attention. In every observed frame, there is a hand token and multiple object tokens. We model the change in hands by cross-attention [54] as shown in Fig. 2. We use the hand token $h_t$ as the query and compute the attention with respect to every object token in the same frame $o_{i,t}, \cdots, o_{N,t}$. The query, key, and value are denoted as $q_{h,t} = h_t W_q, k_{i,t} = o_{i,t} W_k, v_{i,t} = o_{i,t} W_v$, respectively. We use cross-attention to get the refined hand tokens, $\tilde{h}_t$. Cross-attention implicitly seeks the object token that has the most impact on the hand token by pooling all the object tokens in the frame and weighing each by its probability.

Similarly, we use cross-attention to refine each object token using the hand token and other object tokens in the frame. Every object token $o_{i,t}$ acts as a query, and human and other object tokens act as keys and values. Let $z_{i,t}$ represent either hand or other object tokens in the frame $t$. There are $N$ such tokens in each frame for each object query $o_{i,t}$. The query, key, and values are obtained as $q_{i,t} = o_{i,t} W_q, k_{j,t} = z_{j,t} W_k, v_{j,t} = z_{j,t} W_v$, respectively. The refined object token $\tilde{o}_{i,t}$ are obtained using cross-attention. We call the refined hand and object tokens as interaction tokens $I_t = [\tilde{h}_t, \tilde{o}_{1,t}, \cdots, \tilde{o}_{N,t}]$. We perform spatial cross-attention (SCA) over every frame to obtain all the interaction tokens $I \in \mathbb{R}^{T \times (N+1) \times d}$.

4.2.2 SOT: Self-attention of hand/object over time

We model interaction tokens as the change in hands or objects individually over time as shown in Fig. 3 using self-attention. Hand token $h_t$ is refined using only other hand tokens from all frames. The query, key, and value are obtained from all hand tokens across all frames $q_{h,t} = h_t W_q, k_{h,t} = h_t W_k, v_{h,t} = h_t W_v$ to obtain refined hand token $\tilde{h}_t$ using self-attention. We refine object tokens $o_{i,t}$ of every object $i$ separately over frames using self-attention. The query, key, and value for object $i$ are computed from its own tokens over all the frames $q_{o,t} = o_{i,t} W_q, k_{i,t} = o_{i,t} W_k, v_{i} = o_{i,t} W_v$ to obtain refined object token $\tilde{o}_{i,t}$ using self-attention. We call this method Self-attention Over time SOT and interaction tokens $I_t = [\tilde{h}_t, \tilde{o}_{1,t}, \cdots, \tilde{o}_{N,t}]$ consist of refined hand and object tokens.

4.2.3 UB: Union Box of hand and nearest object

We obtain the third type of interaction token using the hand and the nearest object in every frame. We compute the union bounding box from the hand and the nearest object bounding boxes in every frame. Here the hand and object union box is similar to the union of objects and human regions for interaction detection [29]. We then obtain the union tokens $U \in \mathbb{R}^{T \times d}$ using the method described in Sec. 3.2. Unlike the previous two approaches, the union region consists of both human and object features together. We refine the $T$ union tokens in $U \in \mathbb{R}^{T \times d}$ using self-attention to obtain interaction tokens $I_t \in \mathbb{R}^{T \times d}$. We denote this approach as UB for short. Next, we describe how to obtain context-infused interaction tokens in Sec. 4.3. Then, in Sec. 4.4, we describe how context-infused interactions are used to obtain interaction-centric video representation for action anticipation.

4.3 CI: Context-infused interaction tokens

Here we discuss how we implement Eq. (3). The context plays an important role along with interaction in deciding what are the possible next actions. For example, washing a plate in kitchen sink suggests that the next action is probably close the tap. Hence, we infuse context into interaction tokens by proposing Trajectory Cross Attention (TCA) based on trajectory attention [43]. TCA maintains temporal correspondences between interaction tokens and context tokens.
of a frame as shown in Fig. 4.

Our TCA formulation seeks the probabilistic path of an interaction token between frames. The interaction tokens act as the query on the video tokens $X$ that is representative of the context. Let the video patch at spatial location $s$ in frame $t$ be given by $x_{st} \in \mathbb{R}^d$. The key and value are obtained from the video patch. We have $N + 1$ interaction tokens\(^2\) in every frame. For each interaction token $y_t \in \hat{I}_t$, we obtain a set of trajectory tokens $\hat{y}_{tt'} \in \mathbb{R}^d, \forall t' \geq t$ that represents pooled information weighted by the trajectory probability. The pooling operation implicitly looks for the best location $s$ at frame $t' \geq t$ by comparing the interaction query $q_t = y_t W_q$ to the context keys $k_{st'} = x_{st'} W_k$ using $q, k, v$-attention. Attention is applied spatially and independently for all the interaction tokens in every frame. This is complementary to our previously computed cross attention (Eq. (1) and Eq. (2)) where hand/object query tokens are refined with respect to other object/hand and key value token. In TCA, we seek to infer interaction tokens with the visual context of the video which is similar to [23] where cross-attention is used to infer query tokens with information from key and value tokens.

Once trajectories are computed, we pool them across time to reason about connections across the interaction regions in a frame given the environment. For temporal pooling, the trajectory tokens are projected to a new set of queries, keys, and values $\hat{q}_{tt'} = \hat{y}_{tt'} W_q, \hat{k}_{tt'} = \hat{y}_{tt'} W_k, \hat{v}_{tt'} = \hat{y}_{tt'} W_v$, respectively. The new query $\hat{q}_{tt'}$ has information across the entire trajectory that extends across the entire observed video frames. We perform temporal pooling using 1D attention across the new time (trajectory) dimension to obtain refined interaction tokens. We term these refined interaction tokens as context-infused interaction tokens $\hat{I} = [\hat{y}_1, \cdots, \hat{y}_T]$.

### 4.4. ICV: Interaction-centric Video Representation

Here we discuss how we implement Eq. (4). The next action is dependent on interactions and context. Therefore, our goal is to obtain a video representation that incorporates information from the context-infused interaction tokens. We concatenate the video tokens $X$ and the context-infused interaction tokens $\hat{I}$ to form the augmented video representation $X_\alpha$.\(^3\) Then we perform self-attention over the augmented video tokens $x_\alpha \in X_\alpha$ wherein video tokens attend over the interaction regions tokens and vice-versa. After self-attention, we obtain refined augmented video tokens $\hat{x}_\alpha$. We construct the interaction-centric video representation $X_f$ using $\hat{x}_\alpha$ that corresponds to the original video tokens.

\[
X_f = \hat{x}_\alpha^a, \forall t, \text{ if } x_t^a \in X.
\]

We call $X_f$ as interaction-centric video representation in the same vein as object-centric video representation [22].

Till now, we have only applied a single attention layer for interaction modeling, context infusion for interaction, and interaction-centric video representation. In literature, video transformer approaches [43, 42, 2] have shown excellent performance with multiple layers of attention. We apply 12 layers of Trajectory Attention following MotionFormer [43] on the interaction-centric video representation to obtain the final video representation $X_f$. The reason for choosing MotionFormer is that it performs best empirically. The classification token $x_{cls}$ obtained at the end of multiple trajectory attention layers is used to predict the next action, i.e., $\hat{a}_{next} = \phi(x_{cls})$ where $\phi$ is a linear layer.

**Loss function.** We use cross-entropy loss for the next action label to train our model. We compare the model’s prediction $\hat{a}_{next}$ with the ground truth one-hot label $a_{next}$ for the next action as follows

\[
\mathcal{L}_{ant} = - \sum a_{next} \odot \log(\hat{a}_{next}).
\]

It should be noted that as we train our model with the cross-entropy loss to predict the next action, the interaction tokens are optimized for finding the most influential interactions when predicting the next action (see Fig. 5).

### 5. Experiments and Results

#### 5.1. Datasets and Implementation Details

We evaluate and compare our methods on two large unscripted action anticipation datasets EPIC-
KITCHENS100 [6] (EK100) and EGTEA Gaze+ [38]. For EK100, we report results on the test set evaluation server that uses mean-top5 recall as the metric. For EK100 and EGTEA, anticipation gap ($T_a$) is 1s and 0.5s, respectively.

We follow MotionFormer [43] and use 16-frame long clips of resolution $224 \times 224$ uniformly sampled from an observed video of 64 frames (approximately 2s). Every 3D video token is extracted from a video patch of size $2 \times 16 \times 16$. We extract hand, object, and union tokens following the strategy explained in Sec. 3.2. Then, to implement InAViT(SCA) (Sec. 4.2.1), we use a single cross-attention layer with 12 attention heads. Similarly, InAViT(SOT) and InAViT(UB) are implemented with a self-attention layer with 12 heads. We set the number of objects per frame as 4 for EK100 and 2 for EGTEA based on empirical performance (this also makes sure batch processing is efficient). We report the results of varying the number of objects per frame in Supplementary material section 3. If there are fewer objects (less than 4 or 2 respectively), then we zero pad them using null tokens and mask them, which will not impact the model. The same configuration of objects is used for training other baseline models such as ORViT-MF [22].

EK100 provided hand detections do not contain hand annotations for 20% of the frames. Since our aim is not to localize hands accurately, we reduce the threshold to 0.05 from 0.1 used by EK100 to get hand detections in all frames. Lowering the threshold introduces bounding-boxes that cover the region around the hand that is useful for interaction region modeling and anticipation as shown in Fig. 6. The number of hand regions per-frame is set to 1 as both hands are not visible in most frames. If there are two hands in a frame, we randomly pick one and track it using SORT. Some analysis and statistics about detected objects in frames are shown in Supplementary material section 2.

We use one layer of trajectory cross-attention with 12 attention heads and a temporal resolution of 8 to obtain the context infusion of interaction tokens (Sec. 4.3). We then concatenate the refined interaction and original video tokens to form the augmented video tokens. We use a single self-attention layer with 12 heads on the augmented video tokens to obtain interaction-centric video tokens (Sec. 4.4). Finally, we apply MotionFormer on the interaction-centric video tokens to predict the next action (Sec. 4.4). We use a batch size of 16 video (clips) to train on 4 RTX A5000 GPUs with 24 GB memory each and the learning rate is set to $1e^{-4}$ with AdamW optimizer. We will release our code.

## 5.2. Ablation on InAViT

**Component-wise validation.** In Tab. 1(a), we show the contribution of each component of InAViT - interaction modeling using SCA (Eqs. (1) and (2)), Context Infusion of interaction tokens (CI) (Eq. (3)), and Interaction-Centric Video representation using trajectory attention (ICV) (Eq. (4)). Context infusion improves overall performance but we see the biggest improvement when we add the interaction-centric video representation (i.e. Eq. (4)). So, we conclude that interactions and the interaction-centric video representations are important for action anticipation.

**Comparing different interaction regions.** In Tab. 1(b), we compare the three models for interaction region modeling - spatial cross attention (SCA+ CI + ICV), spatial attention over time (SOT+ CI + ICV), and union boxes (UB + CI + ICV) described in Sec. 4. In our comparison, SCA performs the best on overall, unseen, and tail classes compared to both SOT and UB. SCA contextualizes the visual change of each hand/object better using other objects compared to SOT which computes visual change individually. UB’s focus is narrower than SCA as it only considers the nearest object and potentially leaves out other objects that can be used in the next action. SCA performs much better in tail classes where few examples are available and the model relies on visual information for anticipation. As SCA uses all the objects to model interactions, it extracts the most visual information from every frame to make better predictions.

**Only Hand/Object as interaction regions.** The best-performing SCA interaction model involves both refined hand and refined object tokens as interaction region tokens. In Tab. 1(c), we compare the contribution of only refined hand (SCA(Hand)+ CI + ICV) and only object tokens (SCA(Obj)+ CI + ICV) by using either of them as interaction region tokens. As in SCA formulation, we refine hand tokens with objects and object tokens with hand and other objects. Refined hand tokens perform better than refined object tokens as the position of the hand is vital in determining what object(s) can be used next. Still, interaction region tokens containing both refined hands and object tokens (SCA+ CI + ICV) perform the best. We conclude that modeling the changes in hand and object tokens provides useful information to improve action anticipation.

**Effect of infusing context.** We evaluate the effect of context infusion on interaction tokens in Tab. 2. For this comparison, we change either the input or mecha-

---

### Table 1: Ablation of InAViT on EK100 evaluation server [Test set]

Verb and Noun results are in Supplementary.

<table>
<thead>
<tr>
<th>Method</th>
<th>Overall Action(%)</th>
<th>Unseen Action(%)</th>
<th>Tail Action(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCA</td>
<td>12.66</td>
<td>15.49</td>
<td>06.03</td>
</tr>
<tr>
<td>SCA + CI</td>
<td>14.21</td>
<td>14.26</td>
<td>09.12</td>
</tr>
<tr>
<td>SCA+ICV</td>
<td>22.21</td>
<td>20.85</td>
<td>17.07</td>
</tr>
<tr>
<td>SCA + CI + ICV</td>
<td>23.75</td>
<td>23.49</td>
<td>18.11</td>
</tr>
<tr>
<td>(a) Component-wise validation of InAViT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UB+CI + ICV</td>
<td>22.75</td>
<td>22.14</td>
<td>17.04</td>
</tr>
<tr>
<td>SOT+CI + ICV</td>
<td>22.48</td>
<td>20.56</td>
<td>17.46</td>
</tr>
<tr>
<td>(b) Comparing interaction modeling methods</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCA(Hand) + CI + ICV</td>
<td>23.27</td>
<td>23.21</td>
<td>17.57</td>
</tr>
<tr>
<td>SCA(Obj) + CI + ICV</td>
<td>22.49</td>
<td>22.23</td>
<td>16.73</td>
</tr>
<tr>
<td>(c) Comparing refined hand vs. object as interaction tokens</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
nism of context infusion (Sec. 4.3). Interaction modeling is done using SCA and interaction-centric video representation is computed using the original video tokens. For CI (Mask FG), we mask the foreground i.e., hands and objects from the context. We crop the objects and hands based on the bounding boxes and apply a Gaussian filter to soften the edges. We call this masked foreground context and use it to refine interaction tokens. The performance of (SCA+CI(Mask FG)+ICV) is quite poor which means that the complete context with foreground objects (SCA+CI+ICV) is better for refinement. We also find that the concatenation of context (video tokens) to interaction tokens (SCA+Concat+ICV) is worse than our proposed (SCA+CI+ICV) approach.

### 5.3. Comparison with state-of-the-art

We now compare our best-performing InAViT (SCA+CI+ICV) model against state-of-the-art approaches. On EK100 evaluation server, InAViT significantly outperforms other approaches as seen in Tab. 3. InAViT’s performance is much better than AVT [17], MeMVIT [56], and RAFTformer [16] on EK100 which also use visual transformers for representing the video. We also compare InAViT against the baseline MotionFormer (MF) [43] and object-centric video representation ORViT-MF [22]. As ORViT and MF are not trained for action anticipation, we train them using the official repositories.

We are the first to show the effectiveness of MF and ORViT-MF for action anticipation which alone outperforms prior state-of-the-art methods. Our approach InAViT performs even better than both MF and ORViT-MF and achieves significantly better results than the previous best results, the Abstract Goal [50] on EGTEA (Tab. 4) and AVT [17] on EK100. In fact, InAViT outperforms [50] by 18% in mean accuracy and 20.8% in top-1 accuracy on EGTEA. It also outperforms the human-object interaction method in [32] by 30% and 35% on mean and top-1 accuracy, respectively. Similarly, InAViT outperforms AVT [17] by 22%, 17%, and 7%, in the overall verb, noun, and action anticipation on EK100 which is impressive given the large number of actions (3805), nouns (300), and verbs (97). On EK100, InAViT outperforms AVT on unseen and tail action anticip-
(a) Next action: peel onion

(b) Next action: pour sugar

Figure 5: InAViT attends to the location(s) where the next action will occur in (a) onion and (b) cup and sugar.

map of the $x_{cls}$ token used for action anticipation on all spatial tokens across the frames. This helps us understand where InAViT focuses compared to MotionFormer. Motionformer attention is divided into many areas while InAViT attends to the interaction. While anticipating peel onion (Fig. 5(a)), InAViT pays high attention to the exact location the onion is being peeled. Similarly, when anticipating pour sugar (Fig. 5(b)), InAViT attends to both the cup and the sugar container. The frames for visualization are chosen based on the significant motion of hands and objects during the observed action. In Fig. 6(c)(d), we show that InAViT anticipates the action correctly even if the bounding box covers the region around the hand. We show more qualitative results in Supplementary.

6. Discussions and Conclusion

We present an effective method to improve ego-centric action anticipation by capturing human-object interaction information using a Transformer architecture. We showed that our spatial cross-attention (SCA) based human-object interaction extraction along with the trajectory attention-based context infusion (CI) and the interaction-centric video (ICV) representations are effective in egocentric action anticipation. It is interesting to note that our model obtains a 6.0% improvement over published results on EK100 test dataset and a massive 20.0+% improvement on EGTEA Gaze+ dataset. However, the biggest improvement comes from trajectory attention-based MotionFormer. We improve MotionFormer by 4.0% and 1.3% on EK100 and EGTEA Gaze+ datasets, respectively. The findings of this work advance action anticipation research.

Acknowledgment This research/project is supported in part by the National Research Foundation, Singapore under its AI Singapore Program (AISG Award Number: AISG-RP-2019-010) and by the National Research Foundation Singapore and DSO National Laboratories under the AI Singapore Programme (AISG Award No: AISG2-RP-2020-016). This research is also supported by funding allocation to B.F. by the Agency for Science, Technology and Research (A*STAR) under its SERC Central Research Fund (CRF), as well as its Centre for Frontier AI Research (CFAR).
References


[34] Antoine Miech, Ivan Laptev, Josef Sivic, Heng Wang, Lorenzo Torresani, and Du Tran. Leveraging the present to anticipate the future in videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 0–0, 2019.


[61] Tianyu Zhang, Weiqing Min, Jiahao Yang, Tao Liu, Shuqiang Jiang, and Yong Rui. What if we could not see? counterfactual analysis for egocentric action anticipation. In IJCAI, 2021. 2