IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2023.0322000

Anticipating Next Active Objects for Egocentric Videos

SANKET THAKUR[1,3], CIGDEM BEYAN[2], PIETRO MORERIO[1], VITTORIO MURINO[2,4,1], and ALESSIO DEL BUE[1]

[1]Pattern Analysis and Computer Vision (PAVIS) Research Line, Istituto Italiano di Tecnologia (IIT), Genoa, Italy
[2]Department of Computer Science, University of Verona, Italy
[3]Department of Electrical, Electronics and Telecommunication Engineering and Naval Architecture (DITEN), University of Genoa, Italy
[4]Department of Informatics, Bioengineering, Robotics and Systems Engineering (DIBRIS), University of Genoa, Italy

Corresponding author: Cigdem Beyan (e-mail: cigdem.beyan@univr.it).

ABSTRACT Active objects are those in contact with the first person in an egocentric video. This paper addresses the challenge of anticipating the future location of the next active object in relation to a person within a given egocentric video clip, which is challenging since the contact is poised to happen after the last observed frame by the model, even before any action takes place. As we aim to estimate the position of objects, this problem is particularly hard in a scenario where the observed clip and the action segment are separated by the so-called time-to-contact segment. We term this task Anticipating the Next ACTive Object (ANACTO) and introduce a transformer-based self-attention framework to tackle it. We compare our model with the existing anticipation-based methods to establish relevant baseline methods, where our approach outperforms all of them on three major egocentric datasets: EpicKitchens-100, EGTEA+, and Ego4D. We also conduct an ablation study to better present the effectiveness of the proposed and baseline methods on varying conditions. The code as well as the ANACTO task annotations for the aforementioned first two datasets will be made available upon the acceptance of this paper.

INDEX TERMS Egocentric vision, anticipation, next active object, active object, scene understanding

I. INTRODUCTION

THE widespread use of wearable cameras prompted the design of egocentric (first-person) systems that can readily support and help humans in their daily activities, by augmenting their abilities [1]–[3]. In order to assist users, a fundamental problem is to anticipate what the person will do in the next few seconds. Among the various potential tasks, a highly relevant one is to discern, from an egocentric video stream, which object a user will interact with or manipulate in the near future. Pirsiavash and Ramanan [4] define *active objects* as those currently in contact with a person. In the context of action anticipation, objects that will come in contact in the future for future action are termed as *next-active-objects (NAO)*. However, our work extends beyond mere localization of the NAO; it involves modeling the motion and Field-of-View (FoV) drift until the actual contact with the object occurs. This nuanced task provides invaluable insights into understanding not only the future actions of the person but also the interaction dynamics with objects. Anticipating interactable objects presents a significant challenge, as human interactions with the environment depend on their end goals and responses from the surroundings. This task provides insights into whether interacting with an object requires further movement or can be achieved based on the reference from the last observed frame, incorporating the past motions of both the object and the person.

In this paper, we refer to the aforementioned task as *"Anticipating Next ACTive Object" (ANACTO)*, which aims to localize the next-active-object's position for anytime in the future, where a new action starts using that object. It is crucial to emphasize that ANACTO is distinct from Short-Term Anticipation (STA) defined in [5]. STA requires localizing the position of the Next-Active-Object (NAO) in the **last observed frame**, without considering how and where the contact with the object actually occurs. Contrary to this, our ANACTO task extends this definition by potentially incorporating the motion and Field-of-View (FoV) drift of the interactant to anticipate the NAO at **its contact point** in that starting frame(s) of future action (see Fig. 1). In this regard, our task definition aligns with the action anticipation study [4], which characterizes active objects as those currently in contact with the first person. In detail, ANACTO emphasizes the *localization* of the NAO at its contact point *prior to* the commencement of any interactions. One can utilize past

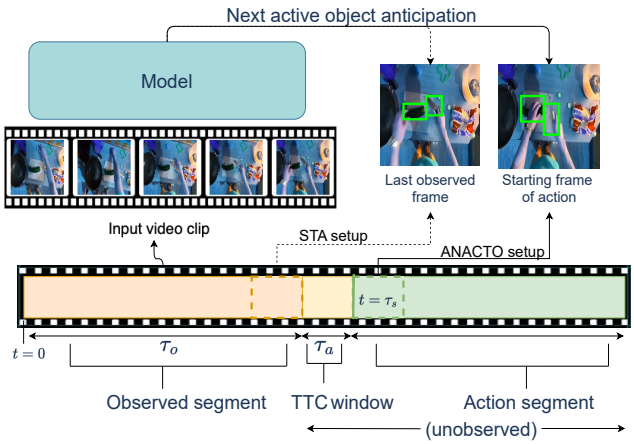

FIGURE 1. The goal of ANACTO is to anticipate the next-active-object, i.e. to localize the object that the person will interact with in the first frame of an action segment, based on the evidence of video clip of length τ_o, located τ_a seconds (anticipation time) before the beginning of an action segment at time-step $t = \tau_s$. Importantly, ANACTO differs from the Short-Term Anticipation (STA) task [5] in that while STA concentrates on forecasting the NAO's position in the final observed frame, ANACTO goes a step beyond by broadening the objective to precisely identify the NAO's location at the outset of a forthcoming action, particularly advantageous for situations where contact occurs at the initiation of the action.

evidence from the observed video clip segment of length τ_o, which precedes the actual action by a *time-to-contact window* τ_a. In other words, it involves predicting the bounding box of the NAO participating in the action in its initial frame(s) at its contact point ($t = 0$). Hence, the ANACTO task encompasses not only the detection/localization of the NAO in the last observed frame (as in the case of Ego4D [5] STA) but also involves anticipating the eventual location of the NAO where the contact/interaction actually occurs, extending well into subsequent frames. In contrast, STA in [5] does not seek to recognize the ultimate interaction with the object. This is because it assumes the static nature of objects, given that only the last observed frame is taken into account.

We aim to tackle the ANACTO task by integrating object-centered cues and scene features, making use of the self-attention mechanism offered by Vision Transformers (VIT) [6]. In detail, the proposed method, termed T-ANACTO, analyzes RGB frames to discern the position and motion of hands without relying explicitly on hand-specific information, such as hand-bounding boxes. Simultaneously, it utilizes an object detector to incorporate the spatial positioning of objects in the observed clip. Given that ego-actions primarily involve interactions between the user's hands and objects in the scene, we posit that the self-attention mechanism of Vision Transformers (VIT) is well-suited for capturing these relationships, both at the frame level and across frames. The validity of this assertion is supported by quantitative analysis, which includes comparisons with various relevant methods, as well as qualitative analysis.

The primary contributions of this work can be summarized as follows.

- We introduce a novel task in the field of egocentric video analysis, named Anticipating the Next ACTive Object (ANACTO).
- We unveil T-ANACTO, a method based on vision transformers that captures interactions between the first-person and objects while considering the time-to-contact window to address the ANACTO task.
- To further present baselines for the ANACTO task, we extended the existing action anticipation methods accordingly.
- T-ANACTO and the aforementioned baselines are evaluated on EpicKitchens-100 [7] (EK-100), EGTEA+ [8], and Ego4D [5] datasets. Performance comparisons across all methods demonstrate the effectiveness of T-ANACTO in all cases.
- For the EK-100 [7] and EGTEA+ [8] datasets, we supply annotations specifically designed for the ANACTO task, aiming to facilitate further research in this area.

The remainder of this paper is structured as follows. Sec. II provides a comprehensive review of related works. In Sec. III, we delineate the ANACTO task. Subsequently, Sec. IV outlines the proposed method to address the ANACTO task. Sec. V covers the datasets used, adaptation of action anticipation state-of-the-art methods for ANACTO, implementation details of our proposed method, and the definition of evaluation metrics. The results, both qualitative and quantitative, are presented in Sec. VI. Finally, Sec. VII concludes the paper, highlighting key findings and proposing potential future directions.

II. RELATED WORK

Initially, we review research on egocentric action anticipation, aligning with our task goal as an anticipation problem even though our emphasis lies in regressing the location of the NAO. Subsequently, we encapsulate insights from the literature on *active objects*, a domain closely associated with NAO.

A. ACTION ANTICIPATION IN EGOCENTRIC VIDEOS

Action anticipation involves predicting future actions before they occur, and this problem has been extensively explored in various actions within *third-person videos* [9], [9]–[13]. The application of action anticipation in *first-person videos*, formalized in [14], has gained recent attention [15]–[18], possibly due to its relevance in wearable computing platforms [19]. Below, we delve into works related to egocentric action anticipation, as we focus on first-person scenes, sharing evaluation protocols and datasets with these studies.

Liu et al. [15] formulate the egocentric action anticipation problem as human-object interaction forecasting. They leverage hand movement as a feature representation to predict interaction hotspots and anticipate future actions. Dessalene et al. [18] perform hand-object contact and activity modeling to anticipate partially observed and/or near-future action. For hand-object contact modeling, the short-term dynamics are learned with 3D-Convolutions. The localization of boundaries between the hands and objects in contact is performed

by applying segmentation through a U-Net [20]. The activity modeling stage embeds the output of contact modeling through Graph Convolutional Network (GCN) layers [21] and then fed to an LSTM, which is followed by a fully connected layer to make action predictions. On the other side, there exist methods relying on the aggregation of the information from the past frames in an observed video clip [16], [17]. For example, [16] propose RU-LSTM, a method composed of a ''rolling'' LSTM (R-LSTM) encoding the past observations, and the ''unrolling'' LSTM (U-LSTM) taking over the current hidden and cell states of the R-LSTM and producing hypotheses of future actions. RU-LSTM [16] processes RGB frames, optical flow and object-based features within an attention mechanism, which estimates optimal fusion weights across these three types of inputs. Differently, the model in [17] uses a predictive model (a CNN) and a transitional model (a CNN pre-trained on action recognition). The predictive model directly anticipates future action, while the transitional model is constrained to the output of the currently happening action that is later used to anticipate future actions. Recently, [19] presented an architecture based on transformers to encode the data performed by the backbone and predict the future actions performed by the head network. That model [19] achieved superior results compared to [16] and showed the better performance of the transformer backbone with respect to using many other backbones such as TSN [22] and Faster R-CNN [23]. Different from [19], our transformer-based architecture aims to exploit the object-centric features with spatial and temporal attention along with two losses introduced to model past observation and learn about the active object(s) to anticipate NAO at its contact point using an autoregressive decoder.

Since our ANACTO task is novel, to obtain relevant baselines to compare with, we have modified several action anticipation SOTA tested on egocentric videos [15], [16], [19] and tested on third-person videos [22]. We include [22] due to its promising results demonstrated in [19] for egocentric settings. For the baselines [15], [22], we append our decoder (see Section IV-A for its definition) to aggregate the frame-level information gathered from their backbone in order to perform the ANACTO task. In terms of encoder design, as we propose a transformer-based architecture, our method differs from [15], [16], [22] which are based on I3D-Res50 [24], LSTMs [25], and Temporal Segment Networks respectively.

B. ACTIVE OBJECTS

For the first time, [4] defined *active* and *passive* objects in an egocentric setup. Their methodology is based on the appearance differences among the objects (e.g., an opened fridge is an active object which looks different from a closed fridge called a passive object), and the location of the active object (i.e., active objects tend to appear close to the center of an egocentric image). By definition, active objects are those, that are currently involved in an interaction, e.g., being touched by humans, whilst, the passive objects are the background objects that the human agent is not in interaction with, e.g., not manipulating them [4], [26]. Dessalene et al. [18] adapted these definitions to describe NAO, which stands for the object that will be contacted with a hand. Their method [18] requires the detection of objects by an object detector to be able to localize NAO and the existence of the hands in the current frames. It was also only tested when some specific action classes (take, move, cut, and open) were considered. Instead, our proposed method processes the frames independent of the hand(s) visibility or presence in the frames. Importantly, we do not specifically restrict the possible (inter)actions between the human and the objects, i.e., we use all the verb classes supplied by the benchmark datasets. Jingjing et al. [27] also explored NAO prediction using cues from visual attention and hand position, but by only using a single frame for the prediction. That approach [27] is not able to differentiate between the past or future active object, since it does not account for the temporal information acquired by the videos. Furnari et al. [28] explored the NAO problem by taking into account the active/passive objects definition of [4]. Their method [28] uses an object tracker to extract the object trajectories for a small video clip till the last frame precedes an action. This trajectory is later used to classify whether a given object is going to be active or passive in the next frame. Such methodology [28] is restricted to predicting the *immediate NAO* instead of predicting the location of the active objects in several future frames, as our proposed method can perform. Moreover, it requires an observation time which is till the penultimate frame of an action segment, which is unpredictable in real-life implementations. Very recently, Liu et al. [29], proposed to forecast *hand trajectories* to detect the interaction hotspots on NAO. But that method is confined to human hands interactions. Instead, our setup is more generic, e.g., can generalize to the interactions with a tool instead of a hand, given that we do not explicitly code the hand features, their trajectories, and/or their bounding box information.

III. ANACTO IN EGOCENTRIC VIDEOS

Given a video clip V, we divide it into three sequential parts: the observed segment of length τ_o, the time-to-contact (TTC) window of length τ_a, and a given action segment that starts at timestep $t = \tau_s$. The objective is to localize the Next Active Object (NAO) at the beginning of the action segment, where the contact occurs. The observation of the video segment spans a duration preceding the action start time τ_s by observation duration τ_o, and it ends τ_a seconds before τ_s, with τ_a representing the time-to-contact window.

ANACTO can be succinctly defined as predicting the location of the NAO in a given observed segment at some future time, precisely at the commencement of interaction when contact occurs. It is important to note that this definition assumes that, for every action, the camera-wearer interacts with an object, activating it at the action's starting point. The problem scope is not limited to "hand"-object interactions; interactions involving tools are also within the ANACTO task's scope. Consequently, our proposed method (see Section IV) does not involve/demand the detection of hands or any explicit hand-related information.

IV. PROPOSED METHOD: T-ANACTO

We propose a method to address the ANACTO task, which analyzes past video frames and incorporates object detections for the input frames. Object detections include the object bounding box parameters (x_c, y_c, w, h) and a confidence score (c_s) produced by the detector. The proposed method is illustrated in Fig. 2.

The proposed method, called T-ANACTO stands for Transformer-based Anticipating Next ACTive Object, leverages the self-attention mechanism to construct an encoder network that operates on individual frames or short clips, followed by a transformer decoder. The reliance on transformers is motivated by their efficient attention mechanisms, which have shown promising results in predictive video modeling, as well as in tasks related to anticipation and object detection [19], [30]–[33]. Our T-ANACTO encoder consists of a VIT [6] and an object detector [23] which are used to extract the feature embeddings from each video frame. Our decoder draws inspiration from [19], leveraging its *causal* structure to address a predictive task focused on past observations, making it autoregressive and well-suited for an egocentric setting. The decoder consolidates information gathered across the temporal dimension to comprehensively interpret the first-person's movements, ultimately aiming to predict the location of the NAO. Significantly, we introduce two losses to guide the model in attending to past active objects, facilitating the prediction of NAO in future frames based on prior observations.

Given a video clip $V = \{X_1, X_2, \ldots X_T\}$ with T frames, where X_t is the RGB image at time step t and an action segment, recall that we have an observed segment length of τ_o, a TTC window (τ_a) which is before the beginning of the action segment at $t = \tau_s$. Frames from the observed segment are then sampled at a frame rate that is equal to τ_a to maintain consistency between frame intervals as described in Fig. 3. Each frame extracted from the observed segment is an input of an individual T-ANACTO encoder. Our object detection head H_o follows a Faster R-CNN [23] architecture and consists of a region proposal network and a regression head. It takes as input each RGB frame X_t and generate bounding boxes $b_{i,t} \in \mathbb{R}^4$ with corresponding confidence score $cs_{i,t} \in (0, 1)$ such that:

$$b_{i,t}, cs_{i,t} = H_o(X_t), \quad i \in \{1, \ldots N\}, \tag{1}$$

where N is the total categories of objects for a dataset. We empirically set a threshold of 0.5 on the confidence score of each detection to discard noisy detections. Therefore, for each category of object(s) detected, only those predictions with confidence scores more than the threshold value are used.

The object detections are performed for the original size of the image frames, X_t (e.g. 1920×1080) and then the bounding boxes are scaled to match the resized image size, X_t^r of 224×224 to match with the input size of VIT [6]. The detections are then reshaped $(BS, N, 5) \rightarrow (BS, -1)$ to be passed through an MLP, f_{MLP} to convert them to the same dimensions as the T-ANACTO encoder's output.

For our video backbone V_b, we adopt ViT-B/16 using 224×224 images, where X_t^r is an image at time t. We split each input frame into 16×16 non-overlapping patches, which are later flattened into a 256-dimensional vector. The vector representation is then projected to a 768-dimensional vector to be used as the input for our transformer encoder. The feature dimensions are kept constant throughout the encoder. We also append a learnable [cls] token into the patch features, which can later be used to identify the class of the active object(s) in the current frame, if any. All the other patches are also allocated a spatial positional embedding with their patch embedding. The resulting patch embeddings are then passed through a standard VIT encoder with pre-norm. Finally, the feature representations learned for each frame from the visual backbone are concatenated with the detections obtained from the object detection head as follows:

$$z_t = V_b(X_t^r) + f_{MLP}(H_o(X_t)). \tag{2}$$

In the end, we add a temporal position encoding to the extracted features from the T-ANACTO encoder for each frame, which are further given to the decoder network.

A. T-ANACTO DECODER

We argue that the past observations can provide a lot of context to produce a hypothesis regarding the NAO in the future. Therefore, for the decoder network, we take inspiration from [19], and extend it to make it autoregressive at each step, to aggregate the features of the past frames and exploit the last predicted active object location which allows us to perform ANACTO.

The decoder network D is designed to produce attentive features corresponding to the future frames: $\hat{z_1}, \ldots, \hat{z_t}$ to anticipate the location of the NAO for each input frame as: $\hat{z_t} = D(z_0, \ldots, z_t; \hat{h_0}, \ldots, \hat{h_{t-1}})$ (see also Fig. 2). Here $\hat{z_t}$ is the predicted features of the *future frame* at t+1 obtained after attending to all other encoded features belonging to the frames before t+1 (i.e., $z_0, z_1..z_t$). At each frame, the decoder takes the previously predicted active object location $\hat{h_t}$ in previous frames along with RGB features to estimate NAO position, $\hat{y_t}$ in future frames. Both these features are concatenated together and are then fed to the next step. This helps in aggregating features of the past frames and understanding the intention and final goal of the first-person, which is defined by the action segment ground-truth label. These features are passed through multiple decoder layers, each consisting of masked multi-head attention, LayerNorm (LN), and a multi-layer perceptron (MLP). The final output is then passed through another LN to obtain the final embeddings. For each decoder output $\hat{z_t}$, it is used to regress the NAO in the corresponding frame at t+1. The predicted features are then fed to a linear layer θ, to regress the bounding box coordinates $\hat{y_t} \in \mathbb{R}$, *i.e.* $\hat{y_t} = \theta(\hat{z_t})$. The final prediction y_t represents the model's output at each frame.

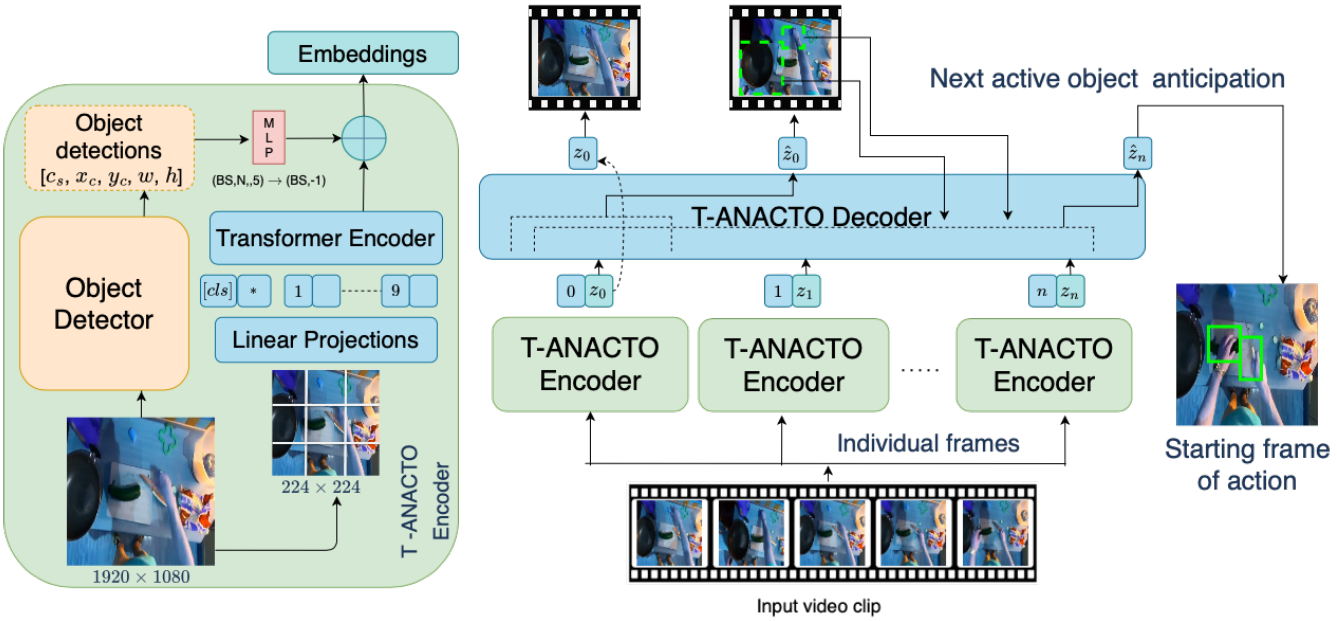

FIGURE 2. **Our T-ANACTO model is an encoder-decoder architecture. Its encoder is composed of an *object detector* and a *Vision Transformer* [6]. The object detector [23] takes an input frame (e.g., size of 1920×1080) and predicts the location of objects in terms of bounding boxes (*x*, *y*, *w*, *h*) and detection confidence scores (*c*). The inputs of VIT are the frame(s), first resized to, 224×224 and then divided into the patches (16×16). The object detections (*x*, *y*, *w*, *h*) are also converted to match the scaled size of the frame (i.e., 224×224), reshaped, and are then passed through an MLP to convert it into the same dimension as the embeddings from the transformer encoder, which are later concatenated together to be given to the decoder. There exists a linear layer between the decoder and the T-ANACTO encoder, which adjusts the feature dimensions to be fed to the transformer decoder. The Transformer decoder uses temporal aggregation to predict the next active object. For each frame, the decoder aggregates the features from the encoder for current and past frames along with the embeddings of the last predicted active objects and then predicts the next active object for the future frames.**

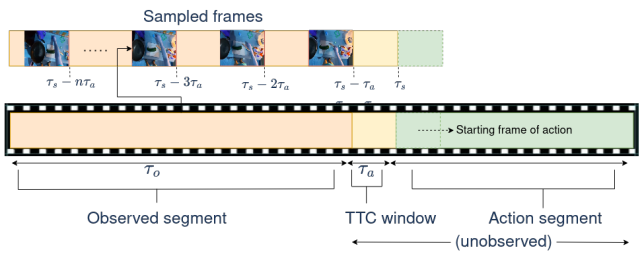

FIGURE 3. **The observed video segment of length τ_o is sampled at a frame rate equal to the TTC time (shown as τ_a) to maintain consistency in (1) the frame interval of sampled frames and (2) between the last observed frame and the starting frame of the action segment, which starts at $t = \tau_s$.**

B. LOSS CALCULATION

To train T-ANACTO, we sample a clip preceding each labeled action segment in a given dataset, ending τ_a seconds before the start of the action. The clip is then sampled with the same frame rate as τ_a seconds to remain consistent with frame intervals as described in Fig. 3. The sampled frames are then passed through our T-ANACTO model and train the network in a supervised manner with three loss functions, described as follows. L_{feat} defined in Eq. 3 aims at leveraging the predictive structure of the model by supervising the future frame features predicted by the decoder to match the true future frame features that are extracted as embeddings from the encoder.

$$\mathcal{L}_{feat} = \sum_{t=0}^{N} ||\hat{z}_t - z_{t+1}||_2^2, \tag{3}$$

where N is the number of frames in training. It is to be noted that our model does not need the presence of a hand or any active object to be present in the observed segment. However, any active object found in the observed segment provides additional supervision using L_{cao}, which stands for current active object loss, is a Mean-Squared Error (MSE) Loss used for the prediction of active objects *in the observed segment of the video clip*. In addition, $\mathcal{L}_{nao}$, stands for the next-active-object loss, forces the model to identify the location of the *NAO at the start of an action*. It is supported by, $\mathcal{L}_{cao}$ which helps T-ANACTO to identify and keep track of active object(s) found at the end of the observed video segment.

$$\mathcal{L}_{cao} = \sum_{t=0}^{N-1} ||y_t - \hat{y_t}||^2, \mathcal{L}_{nao} = ||y_n - \hat{y_n}||^2, \tag{4}$$

where $y_t \in \mathbb{R}$ and $\hat{y_t} \in \mathbb{R}$ are the ground-truth and predicted bounding boxes for active objects in *the current frame*, respectively. Whereas $y_n \in \mathbb{R}$ and $\hat{y_n} \in \mathbb{R}$ are the ground-truth and predicted bounding box for *NAO* in *the starting frame* of an action after $\tau_a sec$, respectively. The final loss is a linear combination of the aforementioned three losses:

$$\mathcal{L} = \mathcal{L}_{feat} + \lambda_1 \mathcal{L}_{cao} + \lambda_2 \mathcal{L}_{nao}, \tag{5}$$

where λ_1, λ_2 are fixed weights.

V. EXPERIMENTAL ANALYSIS

The experimental analyses were conducted on three major egocentric video datasets, described in Section V-A. Given that there is no existing method performing the ANACTO task since this is the first time it is introduced and is being benchmarked, we adapted the SOTA action anticipation methods, whose details are given in Section V-B to perform comparisons against our method T-ANACTO. Throughout this paper, we refer to such methods as *baselines*. The implementation details of T-ANACTO are described in Section V-C.

A. DATASETS

EK-100 [7]. Consists of about 100 hours of recordings with over 20M frames comprising daily activities in kitchens, recorded with 37 participants. It includes 90K action segments, labeled with 97 verbs and 300 nouns (i.e. manipulated objects). It supplies the annotations regarding the hand and object interactions, which are used for ANACTO. In detail, the aforementioned annotations are in terms of the prediction results of a hand-object interaction detector [34], which provides the hand location, side, contact state, and a bounding box surrounding the object that the hand is in contact with. Such detector [34] was trained on EK-55 [14], EGTEA [8] and CharadesEgo [35] datasets, and applied on EK-55 [14] dataset to annotate it with respect to the hand-object interactions. We use the following annotations: the locations of both hands (i.e., the bounding boxes $b \in \mathbb{R}$), and the locations of the objects along with the contact state information at each frame of each video and, then curate the final ground-truth data for ANACTO problem. It is important to mention that the videos in this dataset were collected with different frame rates. In order to apply the methods: [15], [16], [22] requiring frame rates fixed to 30 frames per second, we converted each video to this constant frame rate, thus the annotations regarding the hand locations and active objects' locations are also interpolated accordingly.

EGTEA+ [8]. Includes 28 hours of videos containing 106 action categories, which corresponds to 2.4M frames. There exist 10325 action segments associated with 19 verbs and 53 nouns (i.e., objects) that were recorded with 32 participants. It is important to notice that there exists no publicly available source supplying annotations needed to perform ANACTO for EGTEA+ [8]. Therefore, we created the hand-object interaction annotations following the annotation pipeline of the EK-100 dataset [7]. These include the hand locations (bounding boxes $b \in \mathbb{R}$ and the corresponding detection confidence scores) at each frame, the active object locations, and their contact state. First, all the videos are converted to a constant frame rate of 30 fps. Then, each frame is fed to the hand-object interaction detector model from [34]. The hand and object threshold is kept at 0.5 to produce better qualitative results, which is also the same when extracting the annotations for EK-100 dataset [7]. Additionally, we provide annotations for the videos with the original frame rate for its original frame size.

Ego4D [5]. This is the largest first-person dataset recently released. The dataset is split into 5 different categories, each focusing on a different task, combining a total of 3,670 hours of egocentric videos across 74 locations. For this task, we focus on the forecasting split, containing more than 1000 videos for a total of 960 hours, annotated at 30 fps for the short-term interaction anticipation task. The annotations are for the *NAO* in the *last observed frame*.

B. BASELINE METHODS

We compare T-ANACTO with SOTA action anticipation methods, namely AVT [19], RULSTM [16], Liu et al. [15] and TSN [22]. For RULSTM [16], we used pre-extracted RGB, flow, and object features as in their paper, for EK-100 [7] and EGTEA+ dataset [8]. For Ego4D [5], we computed the flow and RGB features by following the same TSN model mentioned in [16], which were then fed as the inputs to the RULSTM model. We also tested individual modalities with TSN [22] (ResNet101) for RGB frames and RULSTM-object centric path for object modality. Moreover, we used object detections as well as their confidence score from the object detector [23] to be used as object features in RULSTM(fusion) and RULSTM(obj). We modified and re-trained all these aforementioned methods in order to perform the ANACTO task. We explored these methods (noticed that they were used for action anticipation in egocentric videos, a.k.a. a *classification task*) because our problem formulation is very much related to action anticipations, and we claim that these methods can provide effective learning for ANACTO *regression task* by modeling past motion. For each model, we replace the last classification layer with a regression layer to predict the bounding boxes $\hat{y_n} \in \mathbb{R}$ regarding the next active object. Since the TSN [22] method processes individual frames and not a video clip, for the corresponding experiments, we appended the whole T-ANACTO decoder layer to the TSN [22]. This allows the aggregation of information from all frames, i.e., tuning the task from frame-level processing to video processing.

C. IMPLEMENTATION DETAILS OF T-ANACTO

T-ANACTO was trained with an SGD optimizer for 50 epochs with a learning rate of $1e-5$. Recall that a linear layer exists after the output of the decoder to regress the bounding box coordinates. We fixed the values of λ_2 as 1.0 and λ_1 as 0.5 (see Eq. 5) respectively, during the training of T-ANACTO. The weight for feature loss was set to 1.0. For training and testing, our model takes 10 sampled frames as input and takes 1 second to process a batch of 4 clips during inference. We kept the required input number of frames for each baseline method as suggested in their original implementation.

We used annotations from [34] detector for identifying active objects in the observed segment and to train the model for all datasets with $\mathcal{L}_{cao}$ loss. Specifically, for EK-100 [7] and EGTEA+ [8] datasets, during training, we maintained a lookup window of 10 frames from the starting frame of action

to look for the first identified location of active objects *i.e., bounding boxes* (if visible) to be labeled as ground truth for the ANACTO task. It is also possible that for some clips, the *true contact*, i.e., the actual interaction with an object can start sometime later after our lookup window. For such cases, we do not get bounding box labels for the location of the active object. This means no object was actually active during the start of the action segment. We checked whether this situation leads to any inconsistency, and observed that an active object is present *94% and 92%* of the times in the first 10 frames of the action segment for the EK-100 [7] and EGTEA+ [8] dataset, respectively. However, later (in Section VI-D we demonstrate the effectiveness of our model for those cases as well. It is important to notice that EK-100 [7] and EGTEA+ [8] do not supply object detections. As mentioned before, to obtain this information, we rely on Faster-RCNN [23] provided by [7] pre-trained on EK-55 [14] to detect the location of every object in the scene with a confidence score associated with each prediction $b \in \mathbb{R}^5$. For both datasets, we use the training and test splits provided by [16] for the evaluations of the T-ANACTO and the baseline methods. On the other hand, for Ego4D, we used the forecasting split for training and validation provided by [5]. It is important to notice that the annotations provided for NAO are with respect to only *the last observed frame*. As Ego4D [5] is highly big-scaled, it was not possible to annotate it as we performed for other datasets. Therefore, we utilized only the supplied data as the ground truth. On the other hand, this allowed us to show another utility of the ANACTO task as well as T-ANACTO, i.e., both work for the model(s) to forecast NAO in the last observed frame.

D. EVALUATION METRICS

As the evaluation metrics, Average Precision (*AP*) with various Intersection over Union (IoU) thresholds: 5, 10, 20, and 50 as well as their average shown as AP_{avg} are utilized. This choice of metrics is in line with object detection and localization literature such as [36]–[38].

VI. RESULTS

This section encompasses the outcomes of the ablation study, examining the impact of losses and backbones (Sec. VI-A), as well as the anticipation length (Sec. VI-B). Subsequently, we provide performance comparisons between T-ANACTO and the baseline methods (Sec. VI-C). Finally, we discuss the results of T-ANACTO with a qualitative analysis in Sec. VI-D, and present the failure cases in Sec. VI-E.

A. THE EFFECT OF LOSSES AND THE BACKBONE

We conducted an ablation study to assess the losses outlined in Eq. 5. Additionally, we experimented with a different backbone, namely ResNet101, which is the backbone used in TSN [22]. All other settings of T-ANACTO remained constant. The results of these experiments, performed on the EK-100 [7] dataset with an anticipation length of $\tau_a = 0.25s$, are presented in Table 1.

Ablation	AP5	AP10	AP20	AP50	AP_{avg}
ResNet101	31.2	28.1	17.4	2.3	19.75
T-ANACTO w/ $\mathcal{L}_{nao}$	33.5	29.6	19.3	2.4	21.2
T-ANACTO w/ $\mathcal{L}_{cao}$+$\mathcal{L}_{nao}$ (FULL)	**37.1**	**32.6**	**21.1**	**4.1**	**23.7**

TABLE 1. Ablation study performed on the EK-100 [7] to investigate the effect of losses and the backbones of T-ANACTO.

As seen in Table 1, employing a transformer backbone yields better results compared to ResNet101 in all scenarios: (1) ResNet101 vs. T-ANACTO with $\mathcal{L}_{nao}$ and (2) ResNet101 vs. T-ANACTO with $\mathcal{L}_{cao}$+$\mathcal{L}_{nao}$. Furthermore, the inclusion of $\mathcal{L}_{cao}$ noticeably enhances performance in the ANACTO task, highlighting the significance of utilizing object-centric features.

B. EFFECT OF ANTICIPATION LENGTH.

We assess the performance of T-ANACTO and baseline methods across different anticipation lengths for the ANACTO task in unobserved scenes. These experiments were conducted on the EK-100 [7] dataset, and the results are presented in Table 2. It is noteworthy that, as we maintain a constant number of sampled frames from a given observed clip across all experiments, the variation in anticipation time τ_a also impacts the observed length τ_o of the clip. Consequently, in these experiments, the reduction in anticipation length τ_a corresponds to a decrease in the observed length τ_o of the clip.

The results presented in Table 2 demonstrate that modifying anticipation lengths from higher values to lower values (e.g., from 1 second to 0.5 seconds or from 0.5 seconds to 0.25 seconds) consistently enhances the performance of both T-ANACTO and all baseline methods.

C. COMPARISONS AMONG T-ANACTO AND BASELINE METHODS

Table 2 presents a performance comparison among T-ANACTO and baseline methods on the EK-100 dataset [7]. As seen, our method T-ANACTO surpasses all the other methods in all metrics, for all TTC durations, while the second-best method is chaining for different TTC durations. We also present comparisons on EGTEA+ [8] and Ego4D datasets in Tables 3 and 4, respectively, when the TTC duration τ_a is 0.25 seconds for EGTEA+ [8] and rate of sampling frames is 0.25 seconds for Ego4D [5]. To do so, for EGTEA+ [8], we used training and testing splits-1 (see [16] for details) and for Ego4D [5], the experiments were conducted with the training and validation splits provided for the forecasting task. As mentioned in Sec. V-A, the NAO for Ego4D [5] is identified at the end of the past observed segment. Even for this setup, we notice that the attention-based mechanism elevated by object-centric information performs better, compared to other baselines. The obtained results in the aforementioned tables are in line with the results obtained for the EK-100 dataset [7], showing that T-ANACTO outperforms the other baseline methods, while the performance improvement can be up to 12% in terms of AP_{avg}.

Anticipation time	$\tau_a = 1.0$ s					$\tau_a = 0.5$ s					$\tau_a = 0.25$ s				
Models	AP5	AP10	AP20	AP50	AP_{avg}	AP5	AP10	AP20	AP50	AP_{avg}	AP5	AP10	AP20	AP50	AP_{avg}
AVT [19]	25.2	19.1	13.6	1.5	14.9	30.0	26.4	17.2	3.1	19.2	32.3	27.1	18.4	3.3	20.2
RULSTM [16]	27.6	21.3	14.2	2.1	16.3	29.5	24.2	15.5	3.0	18.0	31.6	25.8	16.6	3.1	19.3
TSN(rgb) [22]	17.2	12.1	7.6	0.7	9.4	20.2	16.4	8.6	1.7	11.7	25.6	19.1	11.8	1.8	14.6
RULSTM(obj) [22]	24.4	19.3	11.1	1.7	14.1	24.4	19.1	11.3	1.8	14.1	27.0	20.2	14.7	1.9	16.0
Liu et al. [15]	13.1	9.8	5.2	0.4	7.1	13.4	10.7	5.6	0.6	7.6	14.7	10.4	5.6	0.7	7.9
T-ANACTO	**34.4**	**28.8**	**18.1**	**3.2**	**21.2**	**35.4**	**29.7**	**20.2**	**3.3**	**22.1**	**37.1**	**32.6**	**21.1**	**4.1**	**23.7**

TABLE 2. **Results of our T-ANACTO model and other baseline methods for different TTC duration, i.e., 1, 0.5, and 0.25 seconds, tested on the EK-100 [7] dataset. The best results are given in bold.**

Models	AP5	AP10	AP20	AP50	AP_{avg}
AVT [19]	19.7	16.5	10.2	2.6	12.2
RULSTM [16]	18.8	13.4	7.7	1.4	10.3
TSN(rgb) [22]	14.8	12.1	7.4	1.4	9.0
RULSTM(obj) [16]	15.1	12.4	6.8	1.3	9.0
Liu et al. [15]	11.8	8.5	5.7	1.0	6.8
T-ANACTO	**26.6**	**21.0**	**14.7**	**2.8**	**16.3**

TABLE 3. **T-ANACTO and the baseline methods' performances when they are tested on EGTEA+ dataset [8] with the TTC duration $\tau_a = 0.25s$. The best results of each column are given in bold.**

Models	AP5	AP10	AP20	AP50	AP_{avg}
AVT [19]	38.8	28.7	12.9	2.9	20.8
RULSTM [16]	37.6	27.4	10.3	1.7	19.3
TSN(rgb) [22]	35.5	23.2	8.5	1.5	17.1
RULSTM(obj) [16]	34.6	21.3	8.2	1.5	16.4
Liu et al. [15]	15.2	11.1	7.4	1.1	8.7
T-ANACTO	**41.2**	**31.4**	**18.6**	**4.6**	**24.0**

TABLE 4. **T-ANACTO and the baseline methods' performances when they are tested on Ego4D dataset [5] to identify NAO with respect to the last observed frame. Frames are sampled from the observed segment at 0.25s. The best results of each column are given in bold.**

D. QUALITATIVE ANALYSIS

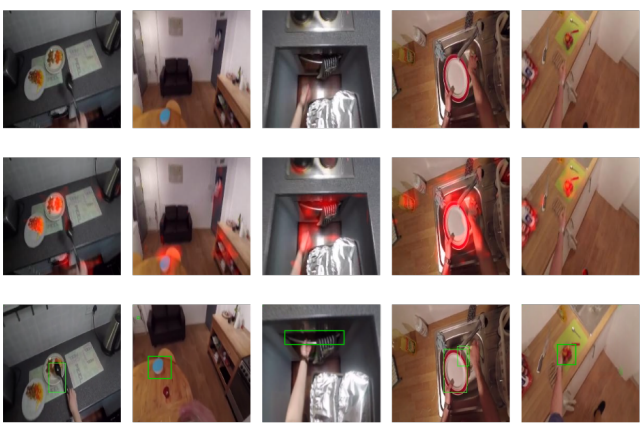

FIGURE 4. **Qualitative results obtained for EK-100 dataset [7]. The top row shows the "last observed frame", the middle row shows "the region of interest of T-ANACTO", and the bottom row shows "the starting frame of an action". The green box(es) in the last row represents the ground-truth bounding box(es) of NAO in the starting frame(s) of action.**

FIGURE 5. **Qualitative results obtained for EK-100 dataset [7]. Our T-ANACTO model also learns to attribute to hand position in an image frame, even though we do not explicitly provide the hand location in training and testing. The spatial attention of our model shown in red identifies the hand positions in addition to possible NAO. The green boxes depict the ground-truth location of NAO.**

We visualize the spatial attention corresponding to our T-ANACTO encoder on the last observed frame in Fig. 4 for EK-100 dataset [7]. In that figure, the red regions demonstrate the regions of interest predicted by our model, which indeed correspond to human-object interaction in the future frames, and are related to anticipating the NAO. The results show that our model learns to focus on objects that are likely to be contacted by the first person based on observation till the last observed frame and the inference can even be performed before the contact happens. In the second column of that figure, one can observe that, even though the object is not active in the starting frame, our model is able to focus on a possible object, which becomes active later on. Additionally, we observe that our model is able to perform equally well for different lighting conditions.

For the majority of the time, our model is also able to identify the hand's positions and interaction hotspots for certain objects, although our model does not explicitly require the hands' position as an input. The results given in Fig. 5 confirm this. Since our method learns to identify the future (in that case) hand-object interaction, it focuses on locating the position of (in that case) hands and respectively locating the NAO in the consequent starting frame of an action segment.

To qualitatively investigate the performance change of the model as the τ_a is reduced from $1.0s$ to 0.25 seconds, we report a comparison in Fig. 6. It is visible that as the model is fed with frames, closer to the beginning of an action segment, *i.e.*

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3395282

FIGURE 6. **Results show the diversity of spatial attention for the last observed frame preceding the beginning of an action segment for different setups of TTC window for $\tau_a = 0.25, 0.5, 1.0$ seconds. The spatial attention regions tend to appear more assertive as the model examines the frames closer to an action segment, *i.e;* as the τ_a is decreased. This also attributes to a higher accuracy of the model for a shorter time to contact window.**

lower τ_a, the model confidence regarding the NAO prediction increases. This is indeed in line with the quantitative results we present in Section 2.

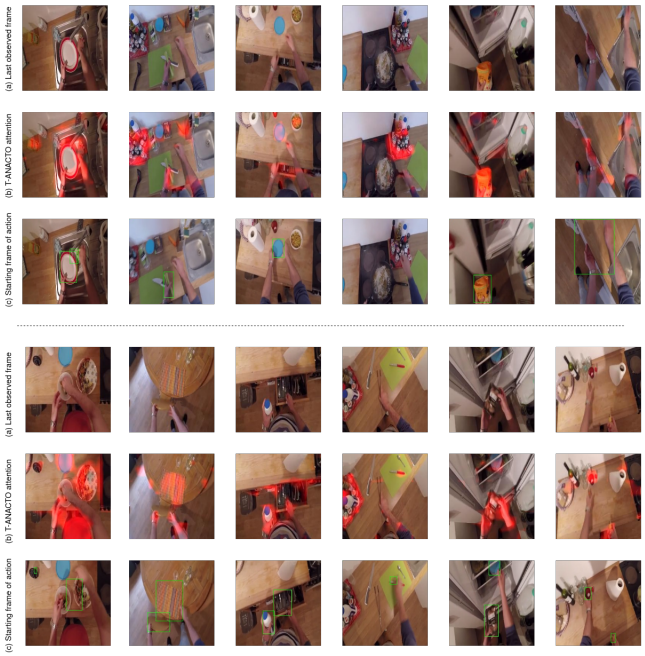

FIGURE 7. Results showing the attention map generated by our T-ANACTO encoder for the last observed frame of video clip with TTC $\tau_a = 0.25$ seconds before the beginning of the action for EK-100 dataset [7]. The red regions depict the region of interest to identify the next active object in the starting frame of the action. The green bounding box for the starting frame of the action (row) shows the localization of the active object for that frame. It is interesting to note that for segments in which there is no active object at the start of the action, our encoder is able to identify the possible area of interest for the next future frames post the starting frame of the action.

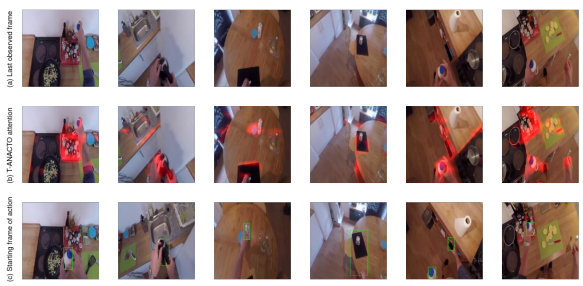

FIGURE 8. Results showing the attention map generated by our T-ANACTO encoder for last observed frame of video clip with TTC $\tau_a = 0.5$ seconds before the beginning of the action or EK-100 dataset [7].

In addition, we show the effectiveness of our model T-ANACTO for anticipating next active object task (ANACTO) for different TTC window $\tau_a = 0.25$ seconds, 0.5 seconds, 1.0 second, as spatial attention of our encoder in additional figures for both EK-100 [7] and EGTEA+ [8] datasets in Fig. 7, 8, 9 and 10. Through these visualizations, one can understand how the confidence of the model differs as it analyzes frames that are temporally distant from the beginning of an action segment for a different TTC window τ_a. In other words, we

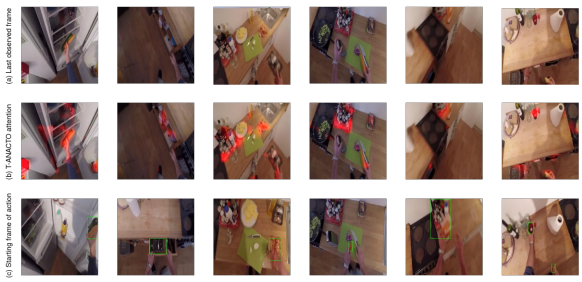

FIGURE 9. Results showing the attention map generated by our T-ANACTO encoder for last observed frame of video clip with TTC $\tau_a = 1.0$ second before the beginning of the action or EK-100 dataset [7].

FIGURE 10. Results showing the spatial attention map for EGTEA+ dataset [8]. The green bounding box specifies the location of the active object at the start of an action.

are able to compare the diversity of attention for a different TTC window, τ_a *v/s* observed τ_o time of video clips for EK-100 dataset [7].

As described earlier, the change in τ_a for a video clip also affects the observed video segment length τ_o proportionally. In Fig. 10, we provide the attention map for the learning of our model for EGTEA+ dataset [8] when trained on *train split 1* and tested on *test split 1*. In Fig. 11, we provide the attention map for the learning of our model for the Ego4D dataset [5] when trained on a training set of forecasting split to predict the next active object location *wrt.* last observed frame. Both sowing the effectiveness of the proposed method.

E. FAILURE CASES

In this section, we discuss the failure cases for T-ANACTO for each dataset. We were able to identify two major cases for EK-100 [7] and EGTEA+ [8] datasets as follows:

Light-colored objects. We noticed that the model might not be able to confine its attention to those areas in the video

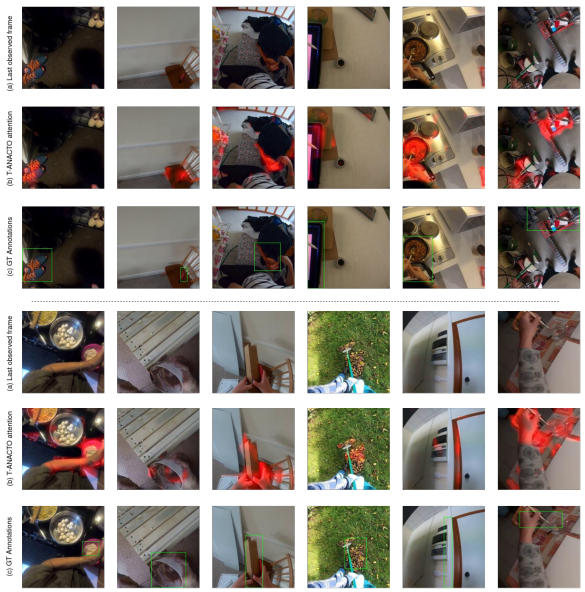

FIGURE 11. **Results shows the spatial attention map for Ego4d [5] dataset when trained to identify next active object *wrt* last observed frame. The green bounding box specifies the location of the object which will become active in the future. The highlighted region specifies the attention stress by the model in the last observed frame.**

FIGURE 12. **Some failure cases of T-ANACTO, collectively for all three datasets. (a) The model fails to attribute attention to objects that are light-colored or easily camouflaged with the background, (b) The scene completely changes at the beginning of action from the past observed segment.**

clips where a light-colored or transparent object is used for human-object interaction (see Fig. 12(a)). It could perhaps be a failure of the object detection model, which is not able to identify items due to the transparent nature of the object and camouflage with the background of frames(s).

Scene transition. As stated earlier, the next active object

FIGURE 13. **Some failure cases of our model for the Ego4D dataset [5]. (a) The model fails to attribute for higher TTC time for a given next active object. (b) For objects that are tiny or transparent or scattered around multiple objects, it is difficult to identify the next active object for larger TTC.**

detection is a challenging task due to the consistent nature of humans to continuously interact with the environment. In the process, a person interacts with the objects based on the activities being performed, which can lead to sudden changes in scenes from one moment to another. Therefore, a current scene at the start of an action segment might be drastically different *wrt.* past observed frames. In such cases, it is extremely difficult for the model to locate "interactable" objects in the scene that have not been observed by the model (see Fig. 12(b)).

For Ego4D dataset [5], we detected two other cases in which the proposed method tends to fail (see in Fig. 13). These are discussed as follow.

Sampling of frames. Since our model takes input frames at a sampled interval, it is trained to output predictions after the sampled interval time after the last observed frame. However, in the Ego4D dataset [5], the TTC for the next active object varies drastically for each clip, which is one of the main reasons our model suffers for those objects whose TTC is much higher than the sampled frame rate for our input frames.

Tiny and clustered objects. We also notice that our model fails for tiny/transparent objects in the scene or where multiple objects are scattered in the frame.

VII. CONCLUSIONS

We have investigated the problem of anticipating the next active object localization. First, we discussed the formulation of the ANACTO task. We then presented a new vision transformer-based model, T-ANACTO which learns to encode first-person-object interactions with the help of an object detector. We proved its effectiveness by comparing it against relevant strong anticipation methods. The experimental evaluation highlights that: (1) the object-centered cues help in elevating the performance to locate the next possible active object; (2) the effectiveness of the model increases when the anticipation time for the prediction before the beginning of an action is kept short. Besides, we also discuss the effect of observation length on the performance of model(s). (3) Our model effectively learns to identify and allocate attention

to possible action objects in the future, as realized from qualitative results. (4) Importantly, T-ANACTO is also able to detect NAO location even in the last observed frame. Finally, we also supply the ANACTO task annotations for EGTEA+ [8] and EK-100 [7] datasets, i.e., hand and active object bounding box annotations along with their contact state as well as providing the object annotations for the entire dataset using an object detector pre-trained on EK-55 [14].

Broad Impact of ANACTO. Addressing this task is beneficial in real-time robotic and industrial applications where challenges involve moving objects, dynamic backgrounds, and the motion of the first person. It is particularly useful for forecasting motion until the point of interaction, providing support in human-robot interactions, such as in factories. In automotive factories, for instance, anticipating interactable objects enables a robotic system to assist in faster maneuvering of industrial parts, reducing assembly time. Given the repetitive nature of these tasks, an AI system can effectively anticipate the required action and object(s). Moreover, such systems contribute to preventing collisions between objects and humans in a warehouse by analyzing past observations and estimating the future point of contact.

Future Work. We plan to extend the ANACTO task to predict dynamic TTC, nouns, and verbs for the NAO. Additionally, we aim to explore the potential of leveraging Vision Language Models to enhance the anticipation capabilities of video models. Our research will also investigate the impact of action recognition on NAO identification and localization.

REFERENCES

[1] D. Damen, T. Leelasawassuk, O. Haines, A. Calway, and W. Mayol-Cuevas, "You-do, i-learn: Discovering task relevant objects and their modes of interaction from multi-user egocentric video," in *Proceedings of BMVC*, 2014.

[2] D. Damen, T. Leelasawassuk, and W. Mayol-Cuevas, "You-do, i-learn: Egocentric unsupervised discovery of objects and their modes of interaction towards video-based guidance," *CVIU*, vol. 149, pp. 98–112, 2016.

[3] T. Kanade and M. Hebert, "First-person vision," *Proceedings of the IEEE*, vol. 100, pp. 2442–2453, 2012.

[4] H. Pirsiavash and D. Ramanan, "Detecting activities of daily living in first-person camera views," in *IEEE CVPR*, 2012, pp. 2847–2854.

[5] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, and A. e. a. Furnari, "Ego4d: Around the world in 3,000 hours of egocentric video," in *Proceedings of the IEEE/CVF CVPR*, June 2022, pp. 18 995–19 012.

[6] A. Dosovitskiy, L. Beyer, and A. K. et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICML*, 2021.

[7] D. Damen, H. Doughty, G. M. Farinella, A. Furnari, J. Ma, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Rescaling egocentric vision," *CoRR*, vol. abs/2006.13256, 2020.

[8] Y. Li, M. Liu, and J. M. Rehg, "In the eye of beholder: Joint learning of gaze and actions in first person video," in *Proceedings of ECCV*, September 2018.

[9] Y. Abu Farha, A. Richard, and J. Gall, "When will you do what? - anticipating temporal occurrences of activities," in *Proceedings of the IEEE CVPR*, June 2018.

[10] P. Felsen, P. Agrawal, and J. Malik, "What will happen next? forecasting player moves in sports videos," in *Proceedings of the IEEE ICCV*, 2017, pp. 3342–3351.

[11] J. Gao, Z. Yang, and R. Nevatia, "Red: Reinforced encoder-decoder networks for action anticipation," *BMVC*, 2017.

[12] C. Vondrick, H. Pirsiavash, and A. Torralba, "Anticipating visual representations from unlabeled video," in *Proceedings of the IEEE CVPR*, June 2016.

[13] C. Rodriguez, B. Fernando, and H. Li, "Action anticipation by predicting future dynamic images," in *Proceedings of ECCV Workshops*, September 2018.

[14] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Scaling egocentric vision: The epic-kitchens dataset," in *ECCV*, 2018.

[15] M. Liu, S. Tang, Y. Li, and J. Rehg, "Forecasting human object interaction: Joint prediction of motor attention and actions in first person video," in *ECCV*, 2020.

[16] A. Furnari and G. M. Farinella, "What would you expect? anticipating egocentric actions with rolling-unrolling lstms and modality attention." in *Proceedings of ICCV*, 2019.

[17] A. Miech, I. Laptev, J. Sivic, H. Wang, L. Torresani, and D. Tran, "Leveraging the present to anticipate the future in videos," in *Proceedings of the IEEE/CVF CVPRw*, 2019, pp. 0–0.

[18] E. Dessalene, C. Devaraj, M. Maynord, C. Fermuller, and Y. Aloimonos, "Forecasting action through contact representations from first person video," *IEEE TPAMI*, pp. 1–1, 2021.

[19] R. Girdhar and K. Grauman, "Anticipative Video Transformer," in *ICCV*, 2021.

[20] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*. Springer, 2015, pp. 234–241.

[21] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[22] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Val Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *ECCV*, 2016.

[23] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NeurIPS*, vol. 28, 2015.

[24] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," *IEEE CVPR*, pp. 4724–4733, 2017.

[25] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: Lstm cells and network architectures," *Neural computation*, vol. 31, no. 7, pp. 1235–1270, 2019.

[26] I. González-Díaz, V. Buso, and J. Benois-Pineau, "Perceptual modeling in the problem of active object recognition in visual scenes," *Pattern Recognition*, vol. 56, pp. 129–141, 2016.

[27] J. Jiang, Z. Nan, H. Chen, S. Chen, and N. Zheng, "Predicting short-term next-active-object through visual attention and hand position," *Neurocomputing*, vol. 433, pp. 212–222, 2021.

[28] A. Furnari, S. Battiato, K. Grauman, and G. M. Farinella, "Next-active-object prediction from egocentric videos," *Journal of Visual Communication and Image Representation*, vol. 49, pp. 401–411, 2017.

[29] S. Liu, S. Tripathi, S. Majumdar, and X. Wang, "Joint hand motion and interaction hotspots prediction from egocentric videos," in *Proceedings of the IEEE/CVF CVPR*, June 2022, pp. 3282–3292.

[30] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *ECCV*, ser. Lecture Notes in Computer Science, vol. 12346. Springer, 2020, pp. 213–229.

[31] Y. Fang, B. Liao, X. Wang, J. Fang, J. Qi, R. Wu, J. Niu, and W. Liu, "You only look at one sequence: Rethinking transformer in vision through object detection," *NeurIPS*, vol. 34, 2021.

[32] B. Kim, J. Lee, J. Kang, E.-S. Kim, and H. J. Kim, "Hotr: End-to-end human-object interaction detection with transformers," in *Proceedings of the IEEE/CVF CVPR*, 2021, pp. 74–83.

[33] I. Rodin, A. Furnari, D. Mavroeidis, and G. M. Farinella, "Predicting the future from first person (egocentric) vision: A survey," *CVIU*, vol. 211, p. 103252, 2021.

[34] D. Shan, J. Geng, M. Shu, and D. Fouhey, "Understanding human hands in contact at internet scale," 2020.

[35] G. A. Sigurdsson, A. K. Gupta, C. Schmid, A. Farhadi, and A. Karteek, "Actor and observer: Joint modeling of first and third-person videos," *2018 IEEE/CVF CVPR*, pp. 7396–7404, 2018.

[36] J. Peng, H. Wang, S. Yue, and Z. Zhang, "Context-aware co-supervision for accurate object detection," *Pattern Recognition*, vol. 121, p. 108199, 2022.

[37] K. Shuang, Z. Lyu, J. Loo, and W. Zhang, "Scale-balanced loss for object detection," *Pattern Recognition*, vol. 117, p. 107997, 2021.

[38] Z. Piao, J. Wang, L. Tanga, B. Zhao, and W. Wang, "Accloc: Anchor-free and two-stage detector for accurate object localization," *Pattern Recognition*, vol. 126, p. 108523, 2022.

SANKET KUMAR THAKUR is a third-year PhD student at the Pattern Analyisis and computer VISion (PAVIS) Research Line of the Italian Institute of Technology (IIT), Genoa, Italy. He completed his B.Tech in Computer Science from Cochin University, India. His main interests is in multimodal learning, video understanding, and object detections. He has also been a winner of CVPR23 EGO4D STA challenge.

CIGDEM BEYAN received her Ph.D. degree in Informatics from the University of Edinburgh, U.K. in 2015. She is currently an Associate Professor at the University of Verona in the Department of Computer Science. Among her main research interest, there are human behavior understanding, social artificial intelligence and multimodal data analysis. She is a reviewer of several journals including various IEEE Transactions, and top-tier IEEE/ACM conferences. She is on the Editorial Board of ICES Journal of Marine Science, a Guest Editor in the International Journal of Social Robotics, and has been a Guest Editor in Frontiers in Robotics and AI. She is a member of ELLIS.

PIETRO MORERIO is a Technologist at the Pattern Analyisis and computer VISion (PAVIS) Research Line of the Italian Institute of Technology (IIT), Genoa, Italy. He received his B.Sc. and M.Sc. in Physics from the University of Milan (Italy) in 2007 and 2010 (summa cum laude). He was Research Fellow at the University of Genoa (Italy) from 2011 to 2012, working in Video Analysis for Interactive Cognitive Environments and pursued a PhD in Computational Intelligence at the same institution in 2016. From 2016 to 2021 he was a Postdoctoral Researcher at Italian Institute of Technology (IIT). His research focuses on machine learning, deep learning and computer vision.

VITTORIO MURINO is a full professor with the University of Verona and Genova, Italy. From 2009 to 2019, he was director of the PAVIS (Pattern Analysis and Computer Vision) Department, Istituto Italiano di Tecnologia, Genova, Italy, and from 2019 to 2021, he worked as senior video intelligence expert with the Ireland Research Centre of Huawei Technologies (Ireland) Company, Ltd. in Dublin. His main research interests include computer vision, pattern recognition and machine learning, nowadays focusing on deep learning approaches, specifically, domain adaptation and generalization and multimodal learning for (human) behavior analysis and related applications, such as video surveillance and biomedical imaging. He is co-author of more than 400 papers published in refereed journals and international conferences, member of the technical committees of important conferences (CVPR, ICCV, ECCV, ICPR, ICIP, etc.), and guest co-editor of special issues in relevant scientific journals. He is also member of the editorial board of Computer Vision and Image Understanding and Machine Vision & Applications journals. Finally, he is IAPR and ELLIS Fellow.

ALESSIO DEL BUE is a tenured senior researcher leading the Pattern Analysis and computer VISion research line of the Italian Institute of Technology in Genoa, Italy. Previously, he was a researcher at the Institute for Systems and Robotics at the Instituto Superior Técnico (IST) in Lisbon, Portugal. Before that, he obtained his Ph.D. in the Department of Computer Science at the Queen Mary University of London. His current research interests are related to 3D scene understanding from multimodal input (images, depth, audio) to support the development of assistive AI systems. He is a co-author of more than 100 scientific publications, in refereed journals and international conferences, a member of the technical committees of important computer vision conferences (CVPR, ICCV, ECCV, BMVC, etc.), and serves as an associate editor of Pattern Recognition and Computer Vision and Image Understanding journals. Finally, he is an IEEE and ELLIS member in the Genoa unit.