Action Scene Graphs for Long-Form Understanding of Egocentric Videos

Ivan Rodin*[1] Antonino Furnari*[1] Kyle Min*[2] Subarna Tripathi[2] Giovanni Maria Farinella[1]

[1]University of Catania [2]Intel Labs

{ivan.rodin,antonino.furnari,giovanni.farinella}@unict.it {kyle.min,subarna.tripathi}@intel.com

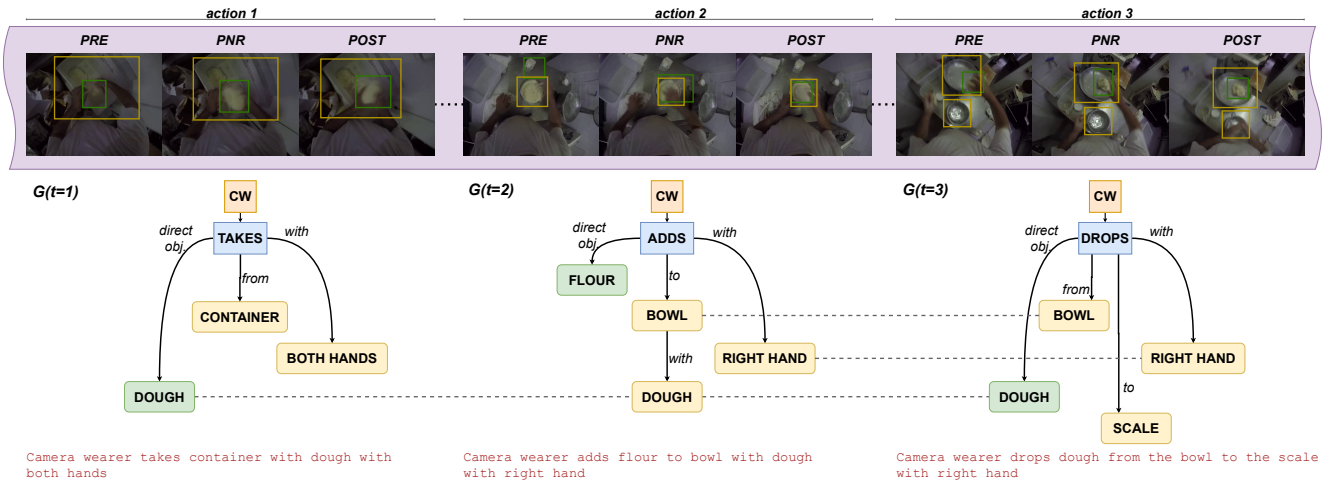

Figure 1. Egocentric Action Scene Graphs are temporal dynamic graphs ($G(t)$) capturing the action verbs (nodes in blue), direct or active objects (nodes in green), and other objects (nodes in yellow) involved in the activity performed by a camera wearer (the orange CW node). Edges between nodes represent relationship between the verb and the objects or between object pairs. The graph evolves through time providing a long-from representation of the egocentric video (dashed lines). Objects of interaction are grounded with bounding boxes.

Abstract

We present Egocentric Action Scene Graphs (EASGs), a new representation for long-form understanding of egocentric videos. EASGs extend standard manually-annotated representations of egocentric videos, such as verb-noun action labels, by providing a temporally evolving graph-based description of the actions performed by the camera wearer, including interacted objects, their relationships, and how actions unfold in time. Through a novel annotation procedure, we extend the Ego4D dataset adding manually labeled Egocentric Action Scene Graphs which offer a rich set of annotations for long-from egocentric video understanding. We hence define the EASG generation task and provide a baseline approach, establishing preliminary benchmarks. Experiments on two downstream tasks, action anticipation and activity summarization, highlight the effectiveness of EASGs for long-form egocentric video understanding. We will release the dataset and code to replicate experiments and annotations[1].

*These authors contributed equally to this work.

[1]The code is available at https://github.com/fpv-iplab/EASG

1. Introduction

Wearable devices allow to capture video of human activities from an egocentric perspective. A proper analysis of such video can enable a detailed understanding of how humans interact with the environment, how they manipulate objects, and, ultimately, what are their goals and intentions. Easily covering sequences of activities performed by the camera wearer in different physical locations, egocentric video is by its own nature *long-form* [48]. Hence, typical applications of egocentric vision systems require algorithms able to represent and process video over temporal spans that last in the order of minutes or hours. Examples of such applications are action anticipation [5, 12, 40], video summarization [8], and episodic memory retrieval [12]. Despite the relevance of such applications in the panorama of egocentric vision [37], progress in this area has been hindered by the lack of a comprehensive and long-form representation of videos that algorithms can rely on, with popular high-level human-gathered representations being in the form of textual narrations [5], verb-noun action labels [9], temporal bounds for action segments [5, 9, 22], object bounding boxes [36], object state changes [12], and hand-object inter-

action states [7, 41], all short-range representations describing temporal spans lasting few seconds.

In this paper, we introduce a novel graph-based representation of actions performed by the camera wearer in an egocentric video, which we term *Egocentric Action Scene Graph (EASG)*. The proposed representation builds on the literature of scene graphs [15, 16, 38] to extend the classic *verb-noun* action representation available in egocentric vision datasets [5, 6, 9, 12, 22] to a structured format in which a sequence of actions performed by the camera wearer is represented with a *temporal dynamic graph* encoding and grounding to the video the objects involved in the action, the action verb, and the main relationships between the considered objects (see Figure 1). EASGs naturally model the temporal evolution of egocentric actions, thus providing a rich representation to be exploited in a variety of tasks requiring long-form video understanding.

We build on the Ego4D dataset [12], which provides egocentric videos of individuals engaged in a range of activities representative of human perception, and augment it with manually gathered egocentric action scene graph labels collected through a novel annotation procedure involving different labeling steps and a validation stage. As customary in the scene graph literature [15, 50], we benchmark the egocentric action scene graph generation task both to provide baseline results and as a means of investigating the feasibility of automatically recovering such rich human-annotated representations, a fundamental ability for downstream applications. We hence show initial results highlighting the effectiveness of the proposed EASG representation in tackling long-form video understanding tasks such as action anticipation and activity summarization.

The contributions of this paper are as follows: 1) We introduce Egocentric Action Scene Graphs, a novel representation for long-form understanding of egocentric videos; 2) We extend Ego4D with manually annotated EASG labels, which are gathered through a novel annotation procedure; 3) We propose a EASG generation baseline and provide initial baseline results; 4) We present experiments that highlight the effectiveness of the EASG representation for long-form egocentric video understanding. We will release the dataset and the code to replicate data annotation and the experiments.

2. Related works

Our work is related to previous research lines which are revised in the following sections.

Graph-based representations for video understanding The exploration of graph-based representations in image and video analysis has burgeoned over recent years, offering a structured approach to encapsulate complex relationships and interactions among element of the scene inherent in visual data. Seminal works [14, 17, 47, 49, 54]i in this domain have focused on various methodologies to facilitate the transition from raw visual data to structured graph representations. For instance, graph structured data is leveraged for image synthesis in [14, 17], learning video representations in [47], detecting video objects in [54] and person re-identification in [49]. In [45], the Visual Context Tree (VCTree) graph structure is built for the purpose of visual question answering. The work of [25] utilizes a transformer architecture to produce a scene graph of an image, while [4, 11, 21, 31] explores approaches to build dynamic scene graphs to describe the scene based on the video inputs.

Scene graph generation extends beyond being an end goal, as a powerful precursor for downstream applications in computer vision, enabling enhanced performance in complex tasks. For instance, the works of [13] demonstrated how scene graph generation could be leveraged to improve object detection and visual relationship detection, thus offering a more contextual understanding of visual scenes. Similarly, the works of [33, 52, 53] explored the utilization of scene graphs for image captioning, where the generated graphs provided a structured semantic understanding that enriched the descriptive quality of generated captions.

The works of [1, 15, 27, 38, 55] delved into employing scene graphs for video understanding, showcasing that the structured representations facilitated a more nuanced understanding of temporal actions and interactions within videos. These collective efforts accentuate the instrumental role of scene graph generation not just as a standalone objective but as a potent enabler for a spectrum of downstream tasks, amplifying the scope and efficacy of visual understanding.

Building on previous investigations, in this work, we propose a novel graph-based representation for actions in egocentric videos. Our representation is shown to improve long-from video understanding in the considered domain.

Graph-based representations in Egocentric Vision Although there has been substantial work in video scene graph processing, only a limited number of studies have focused on egocentric videos. The unique perspective offered by egocentric videos allows for a different approach to understand human interactions and activities. Scene graphs from the perspective of autonomous vehicles have been more extensively researched by the community [19, 26] than the human-centric view videos. However, there are a few previous works studying the applicability of scene graph representation in egovision. In [28, 29], the authors show how the graph-based representation can be useful for the audio-video diarization of egocentric videos. In [24], the authors solve the problem of scene graph generation by composing exo- and ego-centric view processing to construct scene graphs. In [44], egocentric scene graph representations are used to perform downstream tasks of embodied navigation. Ego-Topo [32] presents graphs derived from egocentric videos, encoding the scene topology, thereby enhancing

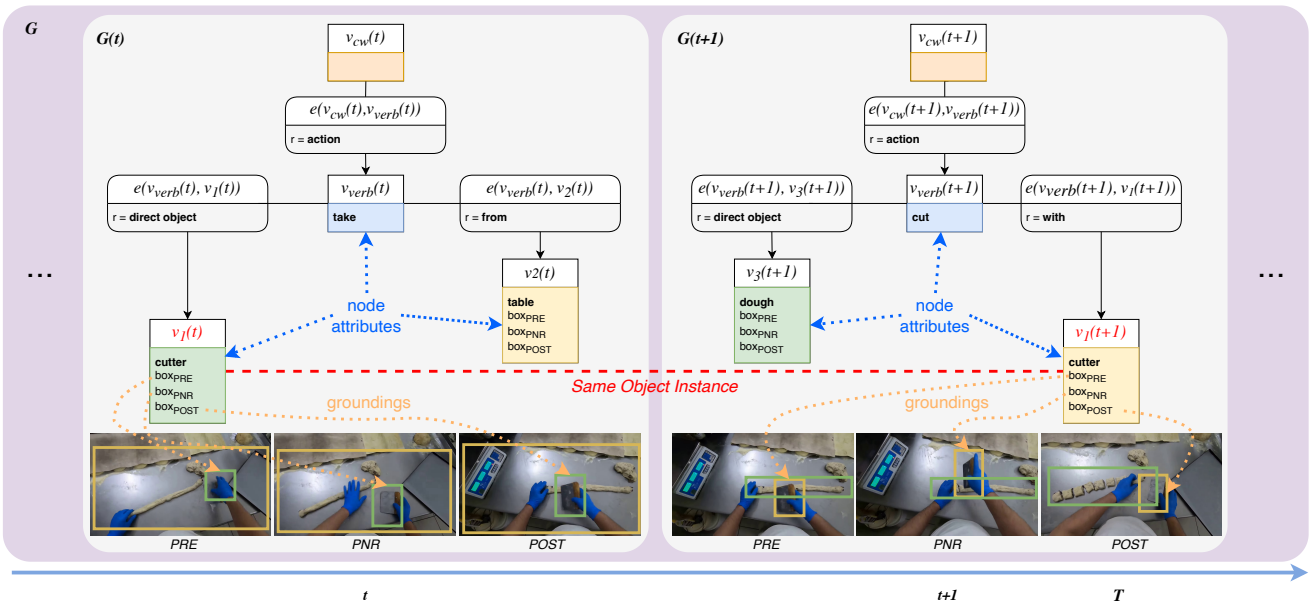

Figure 2. An Egocentric Action Scene Graph (EASG) is a time-varying directed graph $G(t) = (V(t), E(t))$, where nodes $V(t)$ represent either the camera wearer ($v_{cw}(t)$), the action verb ($v_{verb}(t)$), or the involved objects. Edges $E(t)$ represent relationships $e(v_i(t), v_j(t))$ between node pairs. Each node, except for the CW node, can have one or more attributes $att(v_j(t))$ (indicated in blue). Each object has three grounding bounding boxes in the PRE, PNR and $POST$ frames (highlighted in orange). Nodes v_j representing the same object instance maintain the same index across different timesteps (e.g., $v_1(t)$ and $v_1(t+1)$ highlighted in red).

long-term video understanding and egocentric action anticipation. While these previous studies have shown the potential of extending scene graph generation techniques to egocentric videos, in this paper, we propose a general graph-based representation designed to be descriptive of human-object interactions happening in egocentric videos to improve long-form video understanding.

Graph-based Image and Video Datasets Prominent datasets such as Action Genome [15] and Home Action Genome [38], in particular, have contributed by providing rich graph structures that encode actions and interactions within videos. The Panoptic Scene Graph (PSG) dataset [50] introduced enhanced annotations by replacing bounding boxes with fine-grained object segmentation masks. The Visual Genome dataset [20] has significantly contributed to the elucidation of relationships between objects and attributes through graph representations extracted from images. We extend Ego4D [12] with the proposed graph-based egocentric action annotations, to enhance long-form video understanding and enable further investigations on graph-based representations in egocentric vision.

3. Egocentric Action Scene Graphs

Egocentric Action Scene Graphs (EASGs) provide annotations for a video clip in the form of a dynamic graph. We formalize an EASG as a time-varying directed graph $G(t) = (V(t), E(t))$, where $V(t)$ is the set of nodes at time t and $E(t)$ is the set of edges between such nodes (Figure 2). Each temporal realization of the graph $G(t)$ corresponds to an egocentric action spanning over a set of three frames defined as in [12]: the *precondition* (PRE), the *point of no return* (PNR) and the *postcondition* (POST) frames. The graph $G(t)$ is hence effectively associated to three frames: $\mathcal{F}(t) = \{PRE_t, PNR_t, POST_t\}$. $G(t)$ has two fixed nodes: the camera wearer node $v_{cw}(t)$ representing the camera wearer, and the verb node $v_{verb}(t)$, describing the action performed by the camera wearer at time t. Each graph $G(t)$ also contains a set of object nodes $V_{obj}(t)$ encoding the objects involved in the actions. In this formulation, the camera wearer's hands will appear as object nodes. In sum, we have: $V(t) = \{v_{cw}(t), v_{verb}(t)\} \cup V_{obj}(t)$. Apart for the camera wear node, each other node is associated to one or more attributes through a function att. Hence, for the camera wearer node, we define $att(v_{cw}(t)) = \varnothing$. The verb node is associated to a *verb class attribute*: $att(v_{verb}(t)) = verb$. Noun nodes $v_i(t)$ are associated to a *noun class attribute* $noun$ and to three bounding box attributes grounding the noun to the $PRE(t)$, $PNR(t)$ and $POST(t)$ frames associated to the action taking place at time t: $att(v_i(t)) = (noun, box_{PRE}, box_{PNR}, box_{POST})$. Note that two nodes indexed by the same subscript i are related to the same physical object instance regardless of time t. For instance $v_i(t)$ and $v_i(t')$ represent the same

physical objects even when $t \neq t'$, but their associated bounding box attributes may not correspond as they are related to different frames.

The edges in the graph describe the relationships between nodes. Let $v_i(t)$ and $v_j(t)$ be two nodes in the graph. Then, we can define an edge $(v_i(t), v_j(t)) \in E(t)$ if there is a relationship between the nodes $v_i(t)$ and $v_j(t)$ at time t. We represent the existence of an edge between nodes $v_i(t)$ and $v_j(t)$ using the function e_t, such that $e_t(v_i(t), v_j(t)) = r$, if there is a relationship r between nodes $v_i(t)$ and $v_j(t)$; and $e_t(v_i(t), v_j(t)) = \varnothing$ otherwise. We require $r \in R$, where R is the set of possible relationships between nodes. Relations between verb and object nodes can be of a *direct object* kind (e.g., puts – *dobj* – package), or a preposition (i.e., puts – *in* – fridge), while relationships between object nodes are characterized by the prepositions only (i.e., package –*with* – carrot). Objects $v_i(t)$ which are in a *direct object* relation with the verb node $v_{verb}(t)$ are also referred to as "direct objects", while all other objects are referred to as "indirect objects". There is always an *action* relationship between $v_{cw}(t)$ and $v_{verb}(t)$, i.e., $(v_{cw}, v_{verb}) \in E(t) \wedge r(v_{cw}, v_{verb}) = action$.

Since our representation is centered on the action currently executed by the camera wearer, we add only the objects that are either direct objects (e.g., objects manipulated by $v_{cw}(t)$), or objects that have a direct relationship with either the verb node or any direct object nodes. For example, if the *camera wearer* takes an *apple* from the *table* on which many other objects are located (e.g., a pear), only the apple and table will appear as nodes of the EASG, whereas *pear* will not.

4. Ego4D-EASG Dataset

We build our EASG dataset, *Ego4D-EASG*, by annotating a subset of 221 Ego4D [12] clips sampled over 181 distinct videos containing labels for the State Change Object Detection benchmark (SCOD). These labels, together with the narrations available in Ego4D are used to seed the collection of EASG annotations. Let $\mathcal{C} = \{C_1, \ldots, C_N\}$ be the set of selected clips. Each clip C_i consists in a sequence of object state change annotations from the SCOD benchmark $C_i = \{a^i_{t_1}, a^i_{t_2}, \ldots, a^i_{t_{m_i}}\}$, where the generic annotation $a^i_t = (a^{i,PRE}_t, a^{i,PNR}_t, a^{i,POST}_t)$ contains annotations for three salient frames related to an object-state change at time t of the clip C_i: the precondition (PRE), the point of no return (PNR), and the postcondition (POST). Each annotation is defined as $a^{i,x}_t = (f, n, b_o, b_{lh}, b_{rh}, r)$, where x is either PRE, PNR or POST, f is the frame number, n and b_o are the noun class and bounding box of the object of change (the manipulated object), b_{lh} is the bounding box of the left hand, b_{rh} is the bounding box of the right hand, and r is a corresponding free-form narration which is matched to the current annotation. If the right or left hands are not visible in the scene, then either $b_{lh} = \varnothing$ or $b_{rh} = \varnothing$. We labeled an independent EASG $G_i(t)$ for each clip C_i. Each temporal realization of the graph, $G_i(t)$ is seeded from the annotation tuple $a^i_t = (a^{i,PRE}_t, a^{i,PNR}_t, a^{i,POST}_t)$. The data annotation is performed in two stages: 1) the graph annotation stage, and 2) the graph validation stage. These two stages are detailed in the following sections. We used Amazon Mechanical Turk for both stages. After data graphs annotation and validation, a temporal recollection stage allows to turn individual graphs into temporal dynamic graphs. The annotation process is discussed in the following sections. We will release the code to collect annotations following the proposed procedure.

4.1. Egocentric Action Scene Graph Annotation

This stage aims to obtain initial EASG $G_i(t)$ from annotations $a^i_t \in C_i$. This is done through an initialization and a refinement procedures.

Graph Initialization We add by default the camera wearer node $v_{cw}(t)$, the verb node $v_{verb}(t)$, and set the default *action* edge $e_t(v_{cw}(t), v_{verb}(t)) = action$. The *verb* attribute of $v_{verb}(t)$ is set by extracting the verb belonging to the narration r associated to the current annotation a^i_t. We then initialize a new object node $n_k(t)$ to represent the manipulated object. We set the *noun* and *box* attributes of $n_k(t)$ as the noun n and bounding box b_o annotations included in a^i_t $(n, b_o \in a^i_t)$. We add a *direct object* edge between $v_{verb}(t)$ and $n_k(t)$: $e_t(v_{verb}(t), n_k(t)) = direct\ object$.

Graph Refinement We ask three independent AMT annotators to provide manual annotations in order to refine the initial graph $G_i(t)$. The annotation pipeline for this stage is shown in the Figure 3. We first ask annotators to inspect the provided verb-noun pair, the associated PRE, PNR and $POST$ frames and a video clip of 5 seconds sampled around the PNR frame. Initial verb-noun pairs are obtained extracting the verb from the narration and the noun from the SCOD annotation. Since the verbs are inherited from narrations, it may occur that similar verbs have different meanings (e.g. "pick tomato" (selecting) versus "pick up hammer" (lifting)), this allows the graph to keep the expressivity of natural language. At the same time, Ego4D taxonomies can be used to map verbs to "structured" categories, to reduce the number of classes and aggregate the verbs with similar meanings (e.g., "take" may include "pick" and "pick up"). Annotators then check if the verb-noun pair corresponds to the observed clip; if it does not, then the annotators provide a correct $(verb, noun)$ pair and the current annotation is ended and marked for later review. We observe that narrations do not match in $< 5\%$ of the cases. These examples have been later manually checked and re-labeled following the same procedure. We then ask the annotators to specify and ground any additional objects

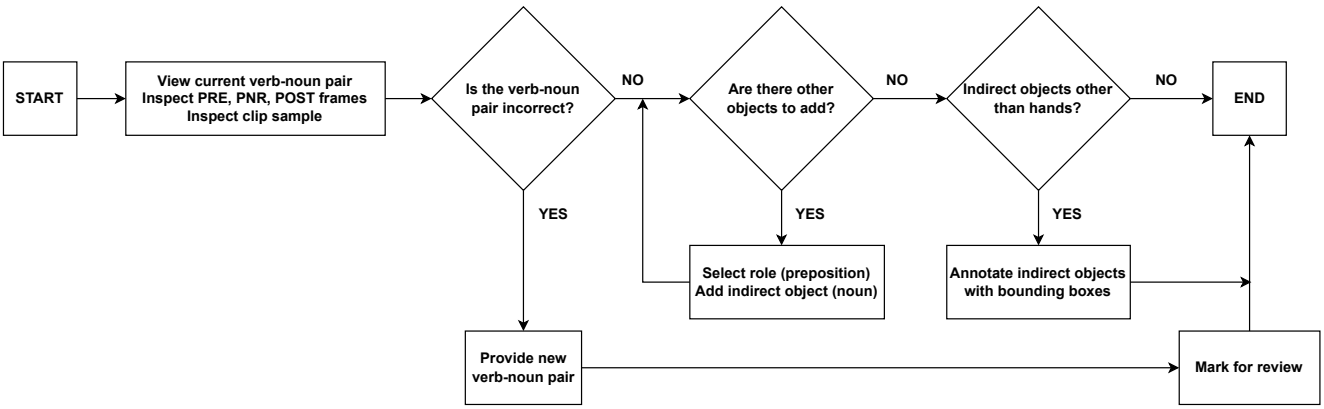

Figure 3. The Ego4D-EASG annotation pipeline. The annotators first review the provided verb-noun pair, the PRE, PNR, $POST$ frames and a clip sampled around PNR. They then check the existing narration and add indirect objects and related groundings, if necessary.

Validation procedure	**Example Questions (answers in red)**
1. Filtering verb-noun pair	Does CW *take bowl* or *press dough*? take bowl
2. Selecting proper preposition in case of multiple edges between two nodes	Select the preposition which is more appropriate: • CW takes bowl **with** left hand ✓ • CW takes bowl **on** left hand
3. Selecting hand(s) if there are different hands with the same preposition	Does CW take bowl with right hand, with left hand or with both hands? left hand
4. Identifying spatial relations	Is the following statement correct: • The bowl is with flour [Y/N] Y • The bowl is from scale [Y/N] N

Example Frame

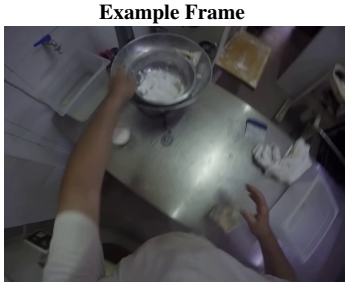

Table 1. Examples of questions (with correct answers in red) asked to the annotators in the validation stage to resolve ambiguities between the labels provided in the annotation stage.

which may be linked to the verb node v_{verb} or any existing object nodes. Note that only indirect objects can be added in this stage. For each newly added object node v_k, annotators are also asked to specify the preposition linking this object to the current graph (e.g., "the camera wearer takes bowl **with** right hand", where "right hand" is the new object and "with" is the specified preposition). We prompt the annotators with some likely objects which may appear in the frame and related prepositions, extracted from the narration r through part of speech tagging, but the annotators were free to add any new objects from the taxonomy of 1610 objects mentioned in Ego4D narrations they may find relevant in the observed video clip. For each of the added *indirect* objects, annotators are also asked to ground them to the PRE, PNR, and $POST$ frames through bounding boxes. If the added objects correspond to the hands, the groundings are set to b_{lh} and b_{rh} as specified in the annotation a_j. At the end of this process, we obtain three graphs $G_i^1(t)$, $G_i^2(t)$, $G_i^3(t)$ as labeled by the three independent annotators.

4.2. Egocentric Action Scene Graph Validation

The validation stage aggregates the data received from the three annotators and ensures the quality of the final annotations. In this stage, for each ($G_i^1(t)$, $G_i^2(t)$, $G_i^3(t)$) graph tuple, we show the annotators the PRE, PNR, and $POST$ frames, the video clip sampled around the PNR and ask a set of questions aiming to sort out inconsistencies across the three graphs. We formulate up to four questions: 1) a question aimed to select the correct verb-noun pair if there is disagreement in the three graphs; 2) a question aimed to disambiguate relations between pairs of nodes, if the three graphs have disagreeing edges between the same node pairs; 3) a question aimed to identify the correct hand used to manipulate objects; 4) a question aimed to disambiguate spatial relationships. The answers provided by the annotators to each of these questions allow to resolve ambiguities and obtain a single graph $G_i(t)$ for each clip C_i and each timestamp t. In our procedure, during the first stage, three annotators agreed 84% of time, while the remaining $G(t)$ were sent to validation stage and were validated by one annotator. Table 1 reports example questions and correct answers for an example annotation.

4.3. Temporal Recollection

The graphs $G_i(t)$ obtained through the annotation and validation stages are *static graphs*, meaning that node indices at different timestamps do not necessarily indicate the same object. For instance, the object "plate" may be identified by

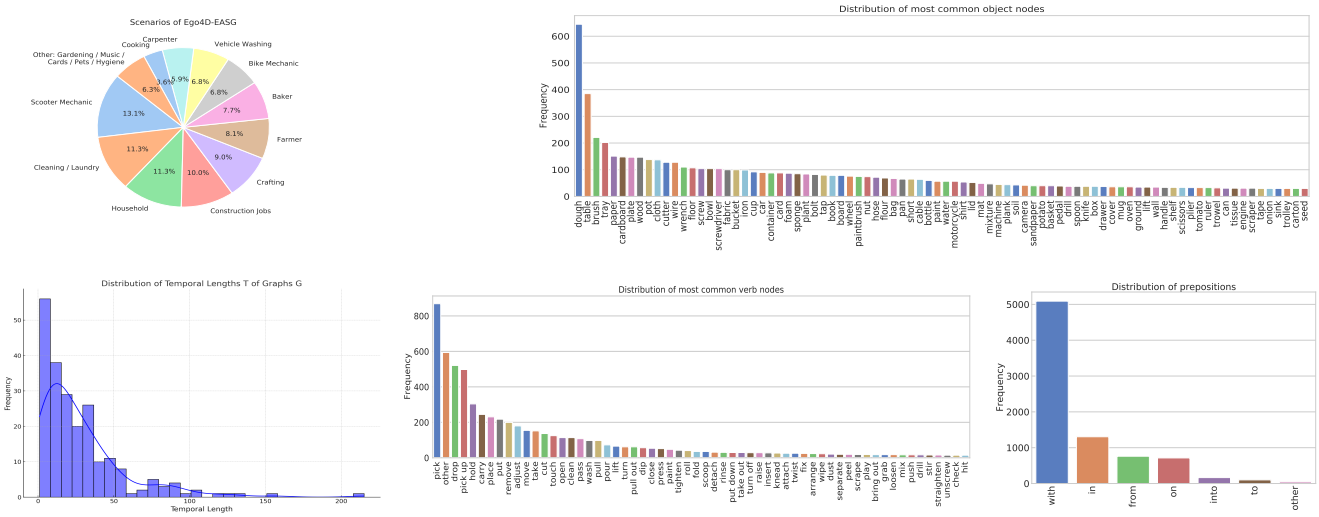

Figure 4. Left-to-right, top-to-bottom: Distributions of clips across scenarios, object nodes, temporal lengths T of graphs G, verb nodes, and relation categories (excluding *action* and *direct object* relations). Data is distributed across different scenarios related to egocentric perception, long-tailed object, verb distributions, and prepositions. The distribution of temporal length of graphs shows the long-form nature of our annotations, with most graphs having a length of up to 50 timesteps.

$n_i(t)$ and $n_j(t')$ with $t \neq t'$. In this stage, we reason globally on the dynamic graph $G_i(t), t = 1 \ldots, T$ and re-assign node indices to make sure that object nodes representing the same object instance are assigned the same index. At the end of this process, a "plate" object will be indexed with the same subscript across timestamps: $n_i(t)$ and $n_i(t')$ with $t \neq t'$. This makes sure that $G_i(t)$ can be interpreted as a dynamic graph across all timestamps.

4.4. Dataset Statistics and Comparison with Other Scene Graph Datasets

Table 2 reports statistics on the proposed Ego4D-EASG dataset and compares it with existing video scene graph datasets. The proposed dataset is the only one designed for long-form egocentric video understanding and it features 221 egocentric video sequences, 11.4 hours of video, comprising an average labeled sequence length of 3.1 minutes, $T = 28.3$ graphs per video in average, 407 object classes, 219 verb classes, and 16 relation classes. As compared to previous datasets, ours is the only including verb nodes explicitly encoding actions. As a result, the number of relations, which in previous datasets also encoded actions (e.g., "looking at") is lower than in other datasets.

Out of the all clips, 129 belong to the SCOD-train split and 92 to SCOD-val split. The dataset contains 30,478 and 19,342 bounding boxes as object groundings in train and validation splits respectively. For an exhaustive enumeration of the sets of all verbs V_{verb}, objects V_{obj}, and relations R, please refer to the supplementary material. Figure 4 reports statistics on the distribution of scenarios, nouns, verbs, relations, and temporal graph lengths.

We measure the length of each sequence from the timestamp of the $G(1) : PRE$ frame to the timestamp of the $G(T) : POST$ frame.

5. Egocentric Action Scene Graphs Generation

Task Definition Unlike standard scene graph generation, EASG generation aims to predict the action verbs as well as objects and their relationships. We define three EASG generation tasks as follows: (1) Edge classification (*Edge Cls*) is to predict verb-object and object-object relationships given visual features, the ground-truth action verb and object classes, (2) Scene Graph Classification (*SG Cls*) is to predict both the object classes and the edge relationships given visual features and the ground-truth action verb, and (3) Egocentric Action Scene Graph Classification (*EASG Cls*) is to predict all these three components, which encompass action verbs, objects, and edge relationships. We follow [15] and report results for predicate (*Edge Cls*) and scene graph (*SG Cls*) classification, and extend it with *EASG Cls* to evaluate time-evolving graphs.

Experimental Setting We design a baseline model for the novel EASG generation task consisting of task-specific fully-connected layers working on top of pre-extracted visual features. For *Edge Cls*, we use a single-layer model to predict the edge relation from the clip-level features and ROIAlign features of each object bounding box. For the clip-level features, we take the average of SlowFast [10] features (pre-extracted and provided within the Ego4D dataset [12]) for the whole clip spanning from *PRE* to *POST* frames. We extract the ROIAlign features using the Faster-

Dataset	Dynamic	Egocentric	Sequences	Hours	Avg. Len. (seconds)	Avg. Graphs per Vid.	Obj Cls	Verb Cls	Rel Cls
VidVRD [42]	✗	✗	1,000	3	11	3.9*	35	25**	132
VidOR [43]	✗	✗	10,000	99	35	8.8* action + 29.2* spatial	80	42	50
Action Genome [49]	✓	✗	10,000	82	30	5	35	-	25
PVSG [51]	✗	Partly (28%)	400	9	77	382	126	44	57
HOMAGE [38]	✗	paired ego-exo	1,752	25	3	3.8	86	453	29
Ego4D-EASG (Ours)	✓	✓	221	11.4	186	28.3	407	219	16

Table 2. Comparison with existing video scene graph datasets. Our Ego4D-EASG dataset is the only one explicitly designed for long-form egocentric video understanding, featuring egocentric videos, dynamic graphs, an average sequence length of 3.1 minutes and an average number of 28.3 graphs per sequence. *measured in object-relation-object triplets. **intransitive + transitive verb predicates.

Method	With Constraint									No Constraint								
	Edge Cls			*SG Cls*			*EASG Cls*			*Edge Cls*			*SG Cls*			*EASG Cls*		
	R@10	R@20	R@50	R@10	R@20	R@50	R@10	R@20	R@50	R@10	R@20	R@50	R@10	R@20	R@50	R@10	R@20	R@50
Random Guess	8.0	8.0	8.0	0.2	0.4	1.0	0.0	0.0	0.0	36.5	72.6	99.9	0.3	0.5	1.0	0.0	0.0	0.0
Baseline (Ours)	60.4	60.4	60.4	41.4	44.3	50.6	14.3	16.4	17.9	94.4	99.8	100	51.6	58.2	62.4	14.7	18.3	20.9

Table 3. Baseline results for three EASG generation tasks (i.e. *Edge Cls*, *SG Cls*, and *EASG Cls*) in terms of Recall@K.

RCNN [39] pre-trained for the short-term action anticipation benchmark [12]. For *SG Cls*, we add an additional fully-connected layer to predict the object classes from the ROIAlign features. For *EASG Cls*, we add another additional layer to predict the action verb from the clip-level features. Following the convention in the literature of scene graph generation, we evaluate this baseline under two different setups: *With Constraint* and *No Constraint*. The former restricts each graph to have at most a single verb-object relationship, whereas the latter has no such restriction. The baseline model is trained for 10 epochs using the Adam optimizer [18] with a learning rate of 10^{-3}.

Results We report the baseline results for all different tasks and setups in Table 3 using the standard metrics of Recall@K (R@K, K=[10, 20, 50]). Baseline results are compared with random guess. We can observe that the scores of *EASG Cls* are significantly lower than other results, indicating that action verbs introduce another layer of difficulty to EASG understanding.

6. Downstream long-from video understanding tasks with Egocentric Action Scene Graphs

In this section, we report experiments aimed to show the potential of the EASG representation in the downstream tasks of action anticipation and activity summarization. Both tasks require to perform long-form reasoning of egocentric video, processing long video sequences spanning over different timesteps. Following recent results showing the flexibility of Large Language Models (LLMs) as symbolic reasoning machines [30], we perform these experiments with LLMs accessed via the OpenAI API [34]. The experiments aim to examine the expressive power of the EASG representation and its usefulness for downstream applications. We show that EASG offers an expressive way of modeling long-form activities, in comparison with the gold-standard verb-noun action encoding, extensively adopted in previous work [6, 12]. The exact prompts used in the experiments and additional results are provided in the supplement.

6.1. Action anticipation with EASGs

Experimental Setting For the action anticipation task, we use the GPT3 [3] *text-davinci-003* model. We prompt the model to predict the future action from a sequence of length $T \in \{5, 20\}$. We compare two types of representations - EASG and sequences of verb-noun pairs. The input sequence of graphs can be represented as $s_{EASG} = [G(t_0), G(t_0+1), ..., G(t_0+T-1)]$, with $t_0 + T - 1 \geq 20$. Each graph $G(t)$ is represented as a string of triplets, where each triplet encapsulates the relationship between nodes (e.g., *CW - verb - wash; wash - direct object - car; wash - with - sponge*). As an output, we request to provide the future unobserved scene graph $G(t+T)$ in the same triplet format. From the predicted graph, we extract the action as the pair of verb and direct object node class for evaluation. In the verb-noun baseline, the input sequence is represented as $s_{vn} = [s_{vn}(t_0), s_{vn}(t_0+1), ..., s_{vn}(t_0+T-1)]$, with $t_0 + T - 1 \geq 20$. The generic term of the sequence is a $(verb, noun)$ pair extracted from the EASG annotation, where $noun$ is the noun class of the direct object. The ground truth future action is $s_{vn}(t_0+T)$. Given the uncertainty in forecasting future events, we prompt the LLM to output up to $N = 5$ predictions, a standard practice in anticipation [5, 12]. We evaluate results using top-k accuracy, with $k \in \{1, 5\}$, reported for verb, noun, and actions. The sample size for this experiment is 3030.

Results Table 4 reports the results of these experiments. Best results are always achieved by EASG-based represen-

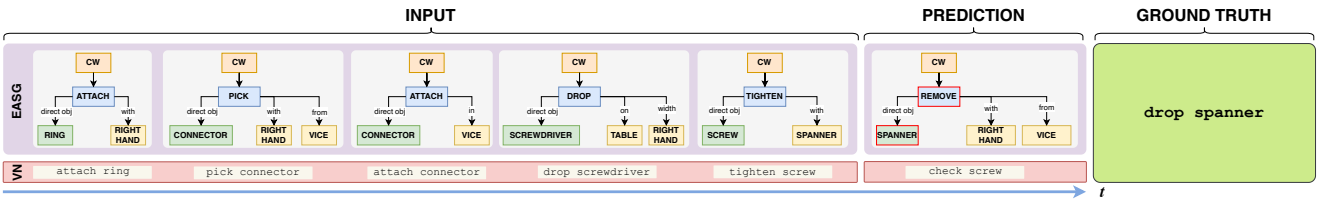

Figure 5. Qualitative example of input sequences and outputs produced using the EASG (top) and verb-noun (bottom) representations for action anticipation, along with the ground truth future action (right). The EASG prediction "remove spanner" is much more semantically aligned to the ground truth "drop spanner" action than "check screw", the prediction based on the verb-noun representation.

			Verb		Noun		Action	
	Seq. length T	Avg. duration	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
V-N	5	19s	2.54	5.01	47.68	62.24	1.28	2.60
EASG	5	19s	3.33	9.53	**48.84**	66.03	1.88	5.24
V-N	20	82s	3.43	8.41	46.69	64.85	2.01	4.98
EASG	20	82s	**5.94**	**15.97**	47.36	**67.26**	**3.40**	**9.24**
Improvement			+2.51	+7.56	+0.67	+2.41	+1.39	+4.26

Table 4. Performance Comparison for the Action anticipation task.

tations. As can be noted, even short EASG sequences ($T = 5$) tend to outperform long V-N sequences ($T = 20$), highlighting the higher representation power of EASG, when compared to standard verb-noun representations. EASG representations achieve the best results for long sequences ($T = 20$). For instance, Top-5 verb is equal to 15.97 for $T = 20$, as compared to 9.53 for $T = 5$. These results further confirm the suitability of EASG for long-form understanding of egocentric video. EASGs bring overall significant improvements of up to +7.56 with respect to the best verb-noun based prediction across the different metrics. Figure 5 reports a qualitative example.

6.2. Long-form activity summarization with EASGs

Experimental Setting We select a subset of 147 Ego4D-EASG clips containing human-annotated summaries describing the activities performed in the clip in 1-2 sentences from Ego4D [12]. We construct three types of input sequences: sequences of graphs $s_{EASG} = [G(1), G(2), ..., G(T_{max})]$, sequences of verb-noun pairs $s_{vn} = [s_{vn}(1), s_{vn}(2), ..., s_{vn}(T_{max})]$, and sequences of original Ego4D narrations, matched with the EASG sequence. This last input is reported for reference, as we expect summarization from narrations to bring the best performance, given the natural bias of language models towards this representation. Each $G(t)$ is represented as a sentence (e.g., `CW wash car with sponge`) to decrease the number of tokens in the input and to align with the natural language form of the predicted output. We select clips for which $T_{max} \geq 5$. We use the *GPT-3.5 Turbo* LLM model for these experiments. We evaluate the produced summaries using the CIDEr [46] metric, adopted in the image captioning literature, and standard metrics for NLG (ROUGE [23], BLEU [35], METEOR [2]).

	CIDEr	ROUGE-1	ROUGE-2	ROUGE-L	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
V-N	9.42	31.5	10.3	29.7	35.7	18.6	7.6	3.9	26.09
EASG	13.79	33.3	10.7	31.4	37.3	19.0	7.8	4.2	26.30
Narrations	19.99	37.7	14.0	34.4	42.0	24.0	11.7	6.7	29.43

Table 5. Results of activity summarization with EASGs and verb-noun representations.

Results Results reported in Table 5 indicate strong improvement in CIDEr score over s_{vn} inputs, showing that models which process EASG inputs capturing detailed object-action relationships, will generate more specific, informative sentences that align well with reference descriptions. As expected, inputs based on narrations achieve the best performance. It should be noted that, while EASG are not as expressive as narrations, they provide a much more structured representation which may be beneficial for the development of computer vision systems. All the NLG metrics show improvements of s_{EASG} over s_{vn} representation, which indicates that indeed, Egocentric Action Scene Graphs provide meaningful information that can improve the quality of long-form video summarization.

7. Conclusion

Our paper reports four key contributions: Egocentric Action Scene Graphs (EASG) as a novel representation for understanding long-form egocentric videos; A procedure for the collection of such graphs and extended the Ego4D dataset with manually annotated EASG labels: Initial baseline results for EASG generation; The validation of the effectiveness of the EASG representation in two downstream tasks, aimed at long-form egocentric video understanding. We believe that these contributions mark a step forward in long-form egocentric video understanding.

Acknowledgements. This research is supported by Intel Corporation. Research at the University of Catania is supported in part by the project Future Artificial Intelligence Research (FAIR) – PNRR MUR Cod. PE0000013 - CUP: E63C22001940006.

References

[1] Anurag Arnab, Chen Sun, and Cordelia Schmid. Unified graph structured models for video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8117–8126, 2021. 2

[2] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005. 8

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 7

[4] Yuren Cong, Wentong Liao, Hanno Ackermann, Bodo Rosenhahn, and Michael Ying Yang. Spatial-temporal transformer for dynamic scene graph generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16372–16382, 2021. 2

[5] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018. 1, 2, 7

[6] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Rescaling egocentric vision. *arXiv preprint arXiv:2006.13256*, 2020. 2, 7

[7] Ahmad Darkhalil, Dandan Shan, Bin Zhu, Jian Ma, Amlan Kar, Richard Higgins, Sanja Fidler, David Fouhey, and Dima Damen. Epic-kitchens visor benchmark: Video segmentations and object relations. *Advances in Neural Information Processing Systems*, 35:13745–13758, 2022. 2

[8] Ana Garcia Del Molino, Cheston Tan, Joo-Hwee Lim, and Ah-Hwee Tan. Summarization of egocentric videos: A comprehensive survey. *IEEE Transactions on Human-Machine Systems*, 47(1):65–76, 2016. 1

[9] Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. In *CVPR 2011*, pages 3281–3288. IEEE, 2011. 1, 2

[10] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 6

[11] Shengyu Feng, Hesham Mostafa, Marcel Nassar, Somdeb Majumdar, and Subarna Tripathi. Exploiting long-term dependencies for generating dynamic scene graphs. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5130–5139, 2023. 2

[12] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022. 1, 2, 3, 4, 6, 7, 8

[13] Tao He, Lianli Gao, Jingkuan Song, and Yuan-Fang Li. Exploiting scene graphs for human-object interaction detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15984–15993, 2021. 2

[14] Roei Herzig, Amir Bar, Huijuan Xu, Gal Chechik, Trevor Darrell, and Amir Globerson. Learning canonical representations for scene graph to image generation. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2020. 2

[15] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10236–10247, 2020. 2, 3, 6

[16] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678, 2015. 2

[17] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *CVPR*, 2018. 2

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[19] Pawit Kochakarn, Daniele De Martini, Daniel Omeiza, and Lars Kunze. Explainable action prediction through self-supervision on scene graphs. *arXiv preprint arXiv:2302.03477*, 2023. 2

[20] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017. 3

[21] Rongjie Li, Songyang Zhang, and Xuming He. Sgtr: End-to-end scene graph generation with transformer. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19486–19496, 2022. 2

[22] Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 619–635, 2018. 1, 2

[23] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 8

[24] Yichao Lu, Cheng Chang, Himanshu Rai, Guangwei Yu, and Maksims Volkovs. Multi-view scene graph generation in videos. In *International Challenge on Activity Recognition (ActivityNet) CVPR 2021 Workshop*, page 2, 2021. 2

[25] Yichao Lu, Himanshu Rai, Jason Chang, Boris Knyazev, Guangwei Yu, Shashank Shekhar, Graham W Taylor, and Maksims Volkovs. Context-aware scene graph generation with seq2seq transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15931–15941, 2021. 2

[26] Arnav Vaibhav Malawade, Shih-Yuan Yu, Brandon Hsu, Harsimrat Kaeley, Anurag Karra, and Mohammad Abdullah

Al Faruque. Roadscene2vec: A tool for extracting and embedding road scene-graphs. *Knowledge-Based Systems*, 242: 108245, 2022. 2

[27] Jianguo Mao, Wenbin Jiang, Xiangdong Wang, Zhifan Feng, Yajuan Lyu, Hong Liu, and Yong Zhu. Dynamic multistep reasoning based on video scene graph for video question answering. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3894–3904, 2022. 2

[28] Kyle Min. Intel labs at ego4d challenge 2022: A better baseline for audio-visual diarization. *arXiv preprint arXiv:2210.07764*, 2022. 2

[29] Kyle Min. Sthg: Spatial-temporal heterogeneous graph learning for advanced audio-visual diarization. *arXiv preprint arXiv:2306.10608*, 2023. 2

[30] Suvir Mirchandani, Fei Xia, Pete Florence, Brian Ichter, Danny Driess, Montserrat Gonzalez Arenas, Kanishka Rao, Dorsa Sadigh, and Andy Zeng. Large language models as general pattern machines. *arXiv preprint arXiv:2307.04721*, 2023. 7

[31] Sayak Nag, Kyle Min, Subarna Tripathi, and Amit K Roy-Chowdhury. Unbiased scene graph generation in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22803–22813, 2023. 2

[32] Tushar Nagarajan, Yanghao Li, Christoph Feichtenhofer, and Kristen Grauman. Ego-topo: Environment affordances from egocentric video. *arXiv preprint arXiv:2001.04583*, 2020. 2

[33] Kien Nguyen, Subarna Tripathi, Bang Du, Tanaya Guha, and Truong Q Nguyen. In defense of scene graphs for image captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1407–1416, 2021. 2

[34] R OpenAI. Gpt-4 technical report. arxiv 2303.08774. *View in Article*, 2023. 7

[35] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 8

[36] Hamed Pirsiavash and Deva Ramanan. Detecting activities of daily living in first-person camera views. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2847–2854. IEEE, 2012. 1

[37] Chiara Plizzari, Gabriele Goletto, Antonino Furnari, Siddhant Bansal, Francesco Ragusa, Giovanni Maria Farinella, Dima Damen, and Tatiana Tommasi. An outlook into the future of egocentric vision. *arXiv preprint arXiv:2308.07123*, 2023. 1

[38] Nishant Rai, Haofeng Chen, Jingwei Ji, Rishi Desai, Kazuki Kozuka, Shun Ishizaka, Ehsan Adeli, and Juan Carlos Niebles. Home action genome: Cooperative compositional action understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11184–11193, 2021. 2, 3, 7

[39] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 7

[40] Ivan Rodin, Antonino Furnari, Dimitrios Mavroeidis, and Giovanni Maria Farinella. Predicting the future from first person (egocentric) vision: A survey. *Computer Vision and Image Understanding*, 2021. 1

[41] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F Fouhey. Understanding human hands in contact at internet scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9869–9878, 2020. 2

[42] Xindi Shang, Tongwei Ren, Jingfan Guo, Hanwang Zhang, and Tat-Seng Chua. Video visual relation detection. In *Proceedings of the 25th ACM International Conference on Multimedia*, page 1300–1308, New York, NY, USA, 2017. Association for Computing Machinery. 7

[43] Xindi Shang, Donglin Di, Junbin Xiao, Yu Cao, Xun Yang, and Tat-Seng Chua. Annotating objects and relations in user-generated videos. In *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, pages 279–287. ACM, 2019. 7

[44] Kunal Pratap Singh, Jordi Salvador, Luca Weihs, and Aniruddha Kembhavi. Scene graph contrastive learning for embodied navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10884–10894, 2023. 2

[45] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6619–6628, 2019. 2

[46] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015. 8

[47] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 2

[48] Chao-Yuan Wu and Philipp Krahenbuhl. Towards long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1884–1894, 2021. 1

[49] Yiming Wu, Omar El Farouk Bourahla, Xi* Li, Fei Wu, Qi Tian, and Xue Zhou. Adaptive graph representation learning for video person re-identification. *IEEE Transactions on Image Processing*, 2020. 2, 7

[50] Jingkang Yang, Yi Zhe Ang, Zujin Guo, Kaiyang Zhou, Wayne Zhang, and Ziwei Liu. Panoptic scene graph generation. In *European Conference on Computer Vision*, pages 178–196. Springer, 2022. 2, 3

[51] Jingkang Yang, Wenxuan Peng, Xiangtai Li, Zujin Guo, Liangyu Chen, Bo Li, Zheng Ma, Kaiyang Zhou, Wayne Zhang, Chen Change Loy, and Ziwei Liu. Panoptic video scene graph generation. In *CVPR*, pages 18675–18685, 2023. 7

[52] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10685–10694, 2019. 2

[53] Xuewen Yang, Yingru Liu, and Xin Wang. Reformer: The relational transformer for image captioning. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 5398–5406, 2022. 2

[54] Yuan Yuan, Xiaodan Liang, X. Wang, D. Y. Yeung, and Abhinav Kumar Gupta. Temporal dynamic graph lstm for action-driven video object detection. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1819–1828, 2017. 2

[55] Yu Zhao, Hao Fei, Yixin Cao, Bobo Li, Meishan Zhang, Jianguo Wei, Min Zhang, and Tat-Seng Chua. Constructing holistic spatio-temporal scene graph for video semantic role labeling. In *ACM Multimedia*, 2023. 2