# Panoptic Video Scene Graph Generation

Jingkang Yang[*,†], Wenxuan Peng[*,†], Xiangtai Li[†], Zujin Guo[†], Liangyu Chen[†], Bo Li[†]
Zheng Ma[‡], Kaiyang Zhou[†], Wayne Zhang[‡], Chen Change Loy[†], Ziwei Liu[†] ✉

[†]S-Lab, Nanyang Technological University, Singapore
[‡]SenseTime Research, Shenzhen, China

https://jingkang50.github.io/PVSG

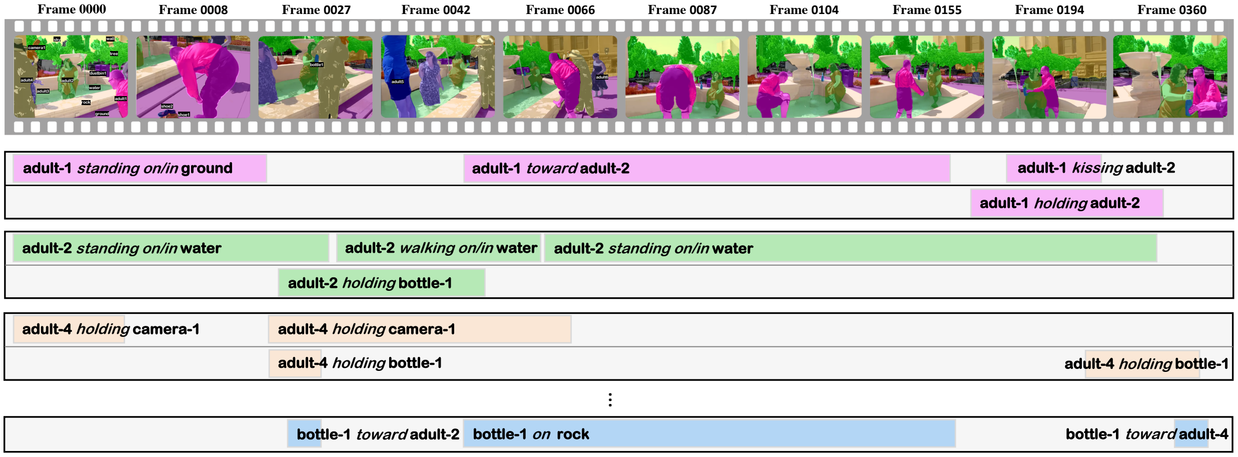


Figure 1. **An example video from our panoptic video scene graph (PVSG) dataset**. The top row shows some keyframes overlaid with the frame-wise panoptic segmentation masks. The timeline tubes underneath the keyframes contain fine, temporal scene graph annotations. The PVSG dataset contains 400 videos (with an average duration of 76.5 seconds), including 289 third-person and 111 egocentric videos.

## Abstract

*Towards building comprehensive real-world visual perception systems, we propose and study a new problem called panoptic scene graph generation (PVSG). PVSG relates to the existing video scene graph generation (VidSGG) problem, which focuses on temporal interactions between humans and objects grounded with bounding boxes in videos. However, the limitation of bounding boxes in detecting non-rigid objects and backgrounds often causes VidSGG to miss key details crucial for comprehensive video understanding. In contrast, PVSG requires nodes in scene graphs to be grounded by more precise, pixel-level segmentation masks, which facilitate holistic scene understanding. To advance research in this new area, we contribute the PVSG dataset, which consists of 400 videos (289 third-person + 111 egocentric videos) with a total of 150K frames labeled with panoptic segmentation masks as well as fine, temporal scene graphs. We also provide a variety of baseline methods and share useful design practices for future work.*

## 1. Introduction

In the last several years, scene graph generation has received increasing attention from the computer vision community [15, 16, 25, 50–53]. Unlike object-centric labels like "person" or "bike", or the precise bounding boxes typical in object detection, scene graphs offer a richer representation of images by capturing both objects and their pairwise relationships and/or interactions, such as "a person riding a bike". A notable trend in this field is the evolution from static, image-based scene graphs to dynamic, video-level scene graphs [1, 43, 51], marking a significant advancement towards more comprehensive visual perception systems.

While videos undoubtedly provide richer information than individual images due to the additional temporal dimension, which greatly aids in understanding temporal events [14], reasoning [61], and identifying causality [10], current video scene graph representations, primarily based on bounding boxes, still fall short of replicating human visual perception. This gap can be attributed to their lack of *granularity*, a limitation that can be overcome by integrating *panoptic segmentation masks*. This is echoed by the evolutionary trajectory in visual perception research, pro-

[*] Main contributors. ✉ Corresponding author.

Table 1. **Comparison between the PVSG dataset and some related datasets**. Specifically, we choose three video scene graph generation (VidSGG) datasets, three video panoptic segmentation (VPS) datasets, and two egocentric video datasets—one for short-term action anticipation (STA) while the other is for video object segmentation (VOS), for comparison. Our PVSG dataset is the first long-video dataset with rich and fine-grained annotations of panoptic segmentation masks and temporal scene graphs.

| Dataset | Task | #Video | Video Hours | Avg. Len. | View | #ObjCls | #RelCls | Annotation | # Seg Frame | Year | Source |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ImageNet-VidVRD [36] | VidSGG | 1,000 | - | - | 3rd | 35 | 132 | Bounding Box | - | 2017 | ILVSRC2016-VID [34] |
| Action Genome [15] | VidSGG | 10,000 | 99 | 35s | 3rd | 80 | 50 | Bounding Box | - | 2019 | YFCC100M [44] |
| VidOR [35] | VidSGG | 10,000 | 82 | 30s | 3rd | 35 | 25 | Bounding Box | - | 2020 | Charades [37] |
| Cityscapes-VPS [17] | VPS | 500 | - | - | vehicle | 19 | - | Panoptic Seg. | 3K | 2020 | - |
| KITTI-STEP [47] | VPS | 50 | - | - | vehicle | 19 | - | Panoptic Seg. | 18K | 2021 | - |
| VIP-Seg [29] | VPS | 3,536 | 5 | 5s | 3rd | 124 | - | Panoptic Seg. | 85K | 2022 | - |
| Ego4D-STA [12] | STA | 1,498 | 111 | 264s | ego | - | - | Bounding Box | - | 2022 | - |
| VISOR [8] | VOS | 179 | 36 | 720s | ego | 257 | 2 | Semantic Seg. | 51K | 2022 | EPIC-KITCHENS [7] |
| **PVSG** | PVSG | 400 | 9 | 77s | 3rd + ego | 126 | 57 | Panoptic Seg. | 150K | 2023 | VidOR + Ego4D + EPIC-KITCHENS |

gressing from image-level labels (i.e., classification) to spatial locations (i.e., object detection), and finally to more fine-grained, pixel-wise masks (i.e., panoptic segmentation [20]).

In this paper, we take scene graphs to the next level by proposing *panoptic video scene graph generation (PVSG)*, a new problem that requires each node in video scene graphs to be grounded by a pixel-level segmentation mask. Panoptic video scene graphs address a critical limitation in bounding box-based video scene graphs: comprehensively covering both "things" and "stuff" classes (i.e., amorphous regions such as water, grass, etc.), with the latter being essential for contextual understanding yet challenging to localize with bounding boxes. For instance, when applying PVSG to the video in Figure 1, relations like "adult-1 standing on the ground" and "adult-2 standing in water" become evident, which are typically overlooked in bounding box-based scene graphs. Furthermore, existing bounding box-based annotations [15] often overlook small yet significant details, for example, "candles on cake".

To help the community progress in this new area, we contribute a carefully annotated PVSG dataset, comprising 400 videos (289 third-person and 111 egocentric) with an average duration of 76.5 seconds each. This dataset encompasses around 150K frames, all annotated with detailed panoptic segmentation and temporal scene graphs, covering 126 object classes and 57 relation classes. A comprehensive comparison of our PVSG dataset with related datasets is shown in Table 1.

Our solution to the PVSG challenge involves a two-stage framework. The first stage generates a set of feature tubes for each mask-based instance tracklet, while the second stage constructs video-level scene graphs based on these tubes. We explore two options for the first stage: 1) combining an image-level panoptic segmentation model with a tracking module, and 2) employing an end-to-end video panoptic segmentation model. For the scene graph generation stage, we present four distinct implementations, encompassing both convolutional and Transformer-based methods.

In summary, we make the following contributions to the scene graph community:

1. **A New Problem**: We identify several issues associated with current research in video scene graph generation and propose a new problem, which combines video scene graph generation with panoptic segmentation for holistic video understanding.

2. **A New Dataset**: A high-quality dataset with fine, temporal scene graph annotations and panoptic segmentation masks is proposed to advance the area of PVSG.

3. **New Methods and Benchmarking**: We propose a two-stage framework to address the PVSG problem and benchmark a variety of design ideas, providing valuable insights for future research in this domain.

We have released two key codebases in this project: PVSGAnnotation[1] for the video panoptic segmentation annotation pipeline, and OpenPVSG[2] to benchmark PVSG methods, both aimed at aiding the community in further research.

## 2. Related Work

**Scene Graph Generation** Given an image, the scene graph generation (SGG) task expects the model to output a scene graph representation, where nodes represent objects and edges represent relations between objects. To localize object instances, the nodes should be grounded by the bounding boxes [50]. Classic scene graph generation methods have been dominated by the two-stage pipeline that consists of object detection and pairwise predicate estimation [40, 41, 50, 58, 60]. Recent works on one-stage methods [4, 24, 52] provide simpler models that output semantically diverse relation predictions. Though the prevalent SGG benchmark Visual Genome [21] provides rich annotations, it suffers from numerous "noisy" ground-truth predicate labels, e.g., some unannotated negative samples are not absolutely background. NICE [23] reformulates SGG as a noisy label learning problem. They re-assign pseudo labels

[1] https://github.com/LilyDaytoy/PVSGAnnotation

[2] https://github.com/LilyDaytoy/OpenPVSG

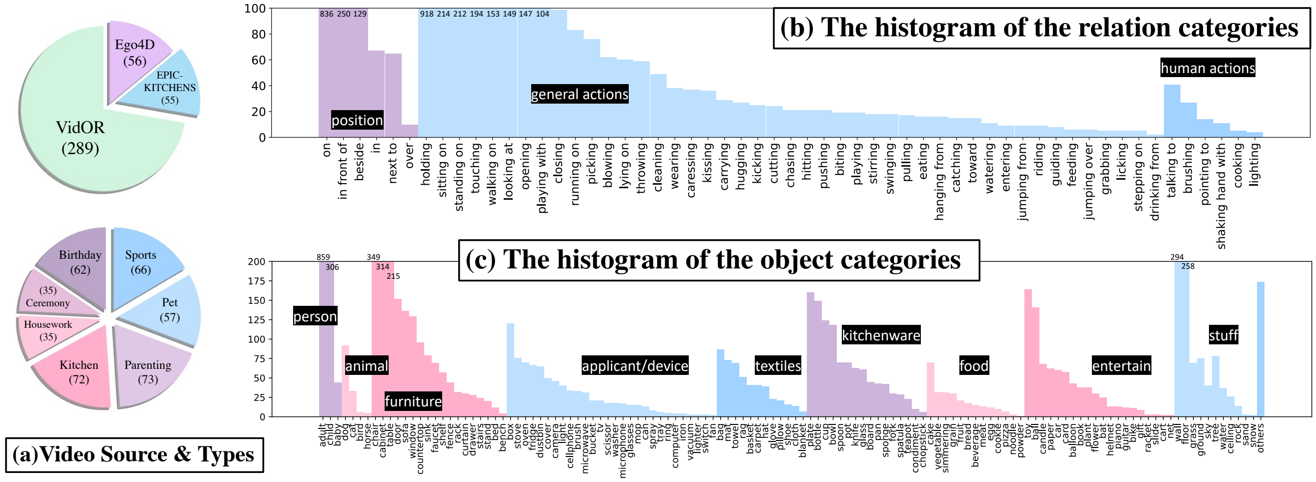


Figure 2. **The PVSG dataset statistics.** The PVSG dataset contains 400 third-person and ego-centric videos from diverse environments, as shown in (a). The statistics of object classes and relation classes are shown in (b) and (c).

to detect noisy negative samples. Instead of exploiting the noisy SGG datasets, recently a new task of panoptic scene graph generation (PSG) [52] has been proposed with a refined PSG dataset, based on panoptic segmentation annotations to identify foreground and background concretely. Our work extends PSG to the video level by predicting spatial-temporal relations.

**Video Scene Graph Generation** Shang *et al.* [36] first proposes Video Scene Graph Generation (VidSGG) and released the ImageNet-VidVRD dataset. They generate object tracklet proposals and short-term relations on overlapping segments. Subsequently, they greedily associate these relation triplets into video level. Several works follow the track-to-detect paradigm with spatio-temporal graph and graph convolutional neural networks [27, 32], or multiple hypothesis association [39]. MVSGG [51] investigates the spatio-temporal conditional bias problem in VidSGG. They perform a meta-training and testing process, constructing the data distribution of each query set w.r.t. the conditional biases. TRACE [43] decouples the context modeling for relation prediction from the complicated low-level entity tracking. [1] raises the issue of domain shift between image and video scene graphs. They exploit external commonsense knowledge to infer the unseen dynamic relationship and employ hierarchical adversarial learning to adapt from image to video data distributions. Embodied Semantic SGG [25] exploits the embodiment of the intelligent agent to autonomously generate an appropriate path by reinforcement learning [9] to explore an environment.

**Video Panoptic Segmentation** Video Panoptic Segmentation (VPS) [18, 29, 48] unifies both Video Semantic Segmentation [5] and Video Instance Segmentation [54] in one framework. It extends panoptic segmentation into video by making instance IDs across frames consistent. VPSNet [18] first extends cityscapes sequences [5] and builds a VPS dataset for driving scene, along with a new metric named Video Panoptic Quality (VPQ). STEP dataset [48] proposes another metric named Segmentation and Tracking Quality (STQ) that decouples the segmentation and tracking error. VIP-Seg [29] proposes a large-scale VPS dataset which contains various scenes. Several works [18, 49, 57] are proposed to solve VPS task respectively. VIP-Deeplab [33] extends the Panoptic-Deeplab [2] with the next frame center map prediction. Video K-Net [26] unifies the VPS pipeline via kernel online tracking and linking. TubeFormer [19] process tube-frames with temporal attention. Compared with previous VPS datasets, our PVSG dataset contains extremely long videos, which bring new challenges for VPS tasks. Moreover, our work goes beyond VPS tasks by also considering relations across a video.

## 3. The PVSG Problem

The goal of the PVSG problem is to describe a given video with a dynamic scene graph, with each node associated with an object and each edge associated with a relation in the temporal space. Formally, the input of the PVSG model is a video clip $\mathbf{V} \in \mathbb{R}^{T\times H\times W\times 3}$, where $T$ denotes the number of frames, and the frame size $H \times W$ should be consistent across the video. The output is a dynamic scene graph $\mathbf{G}$. The PVSG task can be formulated as follows,

$$\Pr\left(\mathbf{G} \mid \mathbf{V}\right) = \Pr\left(\mathbf{M}, \mathbf{O}, \mathbf{R} \mid \mathbf{V}\right). \tag{1}$$

More specifically, $\mathbf{G}$ comprises the binary mask tubes $\mathbf{M} = \{\mathbf{m}_1, \ldots, \mathbf{m}_n\}$ and object labels $\mathbf{O} = \{o_1, \ldots, o_n\}$ that correspond to each of the $n$ objects in the video, and their relations in the set $\mathbf{R} = \{r_1, \ldots, r_l\}$. For object $i$, the mask tube $\mathbf{m}_i \in \{0, 1\}^{T\times H\times W}$ collects all its tracked masks in each frame, and its object category should be $o_i \in \mathbb{C}^O$.

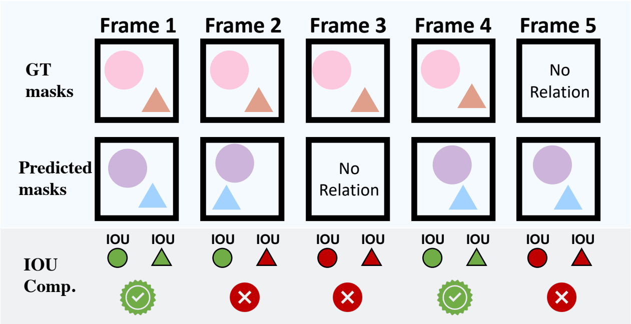


Figure 3. **Illustration of the PVSG Metric.** Assuming the classification of the triplet is correct, to further match the ground truth (GT) frame-wise, the predicted mask pair must have both subject and object masks with a mask IOU above 0.5. In this case, only Frames 1 and 4 satisfy this condition, yielding an intersection count of 2 and a union count of 5. Thus, the volume IOU is calculated as 0.4. As this value falls short of the 0.5 threshold, it is not considered a successful recall.

For all objects in a frame $t$, the masks do not overlap, i.e., $\sum_{i=1}^{n} \mathbf{m}_i^t \leq \mathbf{1}^{H\times W}$. The relation $r_i \in \mathbb{C}^R$ associates a subject and an object with a predicate class and a time period. $\mathbb{C}^O$ and $\mathbb{C}^R$ means the object and predicate classes.

**Metric** In practice, the output of the PVSG task is to predict a set of triplets to describe the input video. Take a triplet as an example, which contains a relation $r_i$ from $t_1$ to $t_2$, associates the subject with its class category $o_s$ and mask tube $\mathbf{m}_s^{(t_1,t_2)}$, and an object with $o_s$ and $\mathbf{m}_o^{(t_1,t_2)}$. $\mathbf{m}^{(t_1,t_2)}$ denotes the mask tube $\mathbf{m}$ span across the period of $t_1$ to $t_2$.

To evaluate the PVSG task, we follow the classic SGG and VidSGG paper and use the metrics of the R@K and mR@K, which calculates the triplet recall and mean recall given the top K triplets from the PVSG model. A successful recall of a ground-truth triplet ($\hat{o}_s$, $\hat{\mathbf{m}}_s^{(\hat{t}_1,\hat{t}_2)}$, $\hat{o}_s$, $\hat{\mathbf{m}}_o^{(\hat{t}_1,\hat{t}_2)}$, $\hat{r}_i^{(\hat{t}_1,\hat{t}_2)}$) should meet the following criteria: 1) the correct category labels of the subject, object, and predicate; 2) the predicted mask tubes ($\mathbf{m}_s^{(t_1,t_2)}$, $\mathbf{m}_o^{(t_1,t_2)}$) and the ground-truth tubes ($\hat{\mathbf{m}}_s^{(\hat{t}_1,\hat{t}_2)}$, $\hat{\mathbf{m}}_o^{(\hat{t}_1,\hat{t}_2)}$) should have volume IOU over 0.5. More specifically, we compute the time IOU between the ground-truth $(t_1, t_2)$ and $(\hat{t}_1, \hat{t}_2)$, and the frame $t$ is considered as intersection only when both $\mathbf{m}_s^t$ and $\mathbf{m}_o^t$ have mask IOU over 0.5 compared to their ground-truth counterpart. Figure 3 shows how volume IOU calculates. Following the scene graph generation conventions, we adopt a 0.5 threshold for time IOU as the standard for considering a triplet recalled. Additionally, in the experiment, we also report results with the threshold of 0.1, a lower standard relaxes the criteria for time span prediction.

Please notice the nuance of the PVSG metrics compared with VidSGG metrics for VidOR [35]. For a case where a child stops and goes several times in a video, different from VidOR which considers several "child-1 walking on ground" triplets, our PVSG metrics only consider the triplet once, but with a scattered time span. This small change avoids some relations dominating the metrics by repeating.

## 4. The PVSG Dataset

In this section, we first summarize the existing VidSGG datasets and highlight their problems. Then, we introduce the overview and statistics of our PVSG dataset and its annotation pipeline.

### 4.1. Connecting Existing Datasets to PVSG

To select candidate video clips for the PVSG dataset, a go-to option is to borrow the videos from other VidSGG datasets. Table 1 lists three classic VidSGG datasets chronologically. While the limited size of their first VidSGG dataset, ImageNet-VidVRD [36], Shang *et al*. collects 10K videos from the user-uploaded dataset YFCC100M [44] and generate a large-scale VIDOR dataset [35], with dense object and relation annotation. Ji *et al*. also introduces a large-scale dataset Action Genome (AG) based on a diverse, crowd-sourcing Charades dataset [37]. While Charades provides a novel solution to gather large-scale, less-biased video datasets by asking people to act based on the generated script, the curated scripts usually produce random action series, such as a man rushing out of the room and running back for no reason. Also, the performance traces turn out to be heavy in the dataset. These shortcomings limit the potential of the community to explore contextual logic and reasoning in videos.

Alternative video datasets that lean toward logic reasoning and video scene understanding are instruction datasets or movie datasets. However, these datasets are either full of close-up shots (e.g., Something-Something [11], Howto100M [30]) or cut shots (e.g., MOMA [28], HC-STVG [42]). In fact, humans rely on unpolished videos to form an essential understanding of the world. In this sense, we find that the unedited, natural, and diverse VidOR [35] videos are a good candidate for learning the visual essence as well as keeping the potential of contextual logic exploration. While the videos presented above showcase a third-person perspective, egocentric videos have gained popularity due to their practicality in autonomous driving [56], robotic decision-making [59], and in the metaverse [31]. In particular, a subset of the Ego4D dataset [12] is suitable for exploring logical relationships and modeling, as it supports short and long-term action anticipation tasks. Additionally, the Epic-Kitchens [6] dataset is focused on the kitchen scenario and offers rich action data. Its subset, the VISOR dataset, includes video object segmentation (VOS) annotation, which partially matches the PVSG scope, though its relations are not yet annotated.

Another dataset category that is closely related to the PVSG problem is the video panoptic segmentation (VPS) datasets. Popular VPS datasets include Cityscapes-

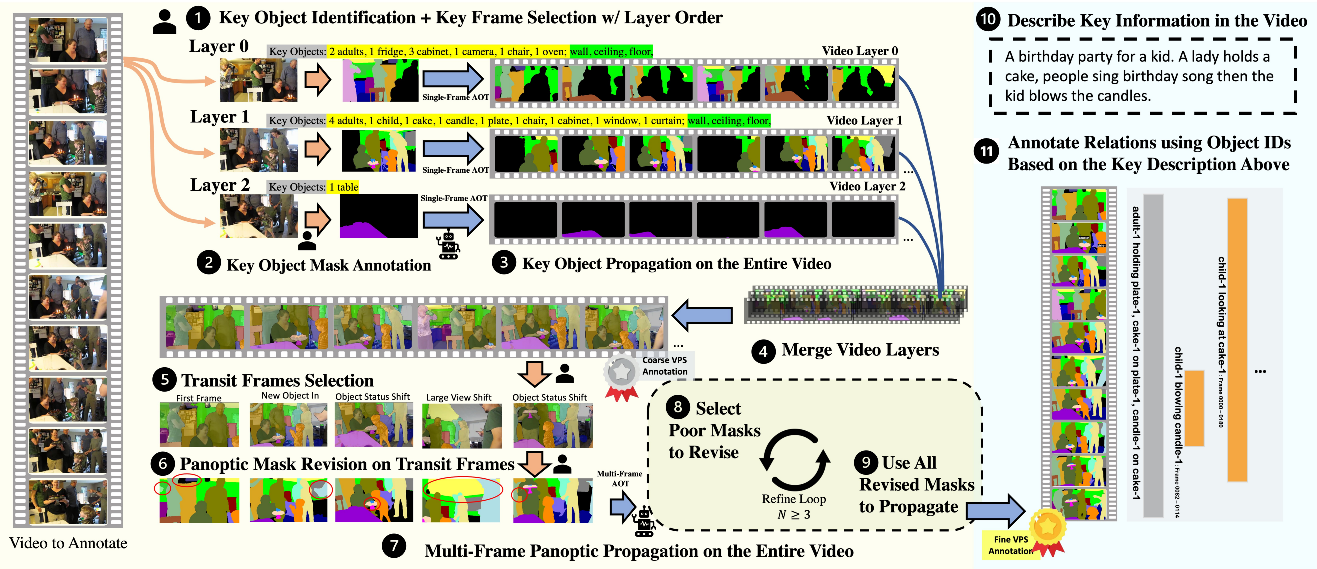

1 Key Object Identification + Key Frame Selection w/ Layer Order
Layer 0
Key Objects: 2 adults, 1 fridge, 3 cabinet, 1 camera, 1 chair, 1 oven; wall, ceiling, floor,
Video Layer 0
Single-Frame AOT
Layer 1
Key Objects: 4 adults, 1 child, 1 cake, 1 candle, 1 plate, 1 chair, 1 cabinet, 1 window, 1 curtain; wall, ceiling, floor,
Video Layer 1
Single-Frame AOT
Layer 2
Key Objects: 1 table
Video Layer 2
Single-Frame AOT
2 Key Object Mask Annotation
3 Key Object Propagation on the Entire Video
4 Merge Video Layers
Coarse VPS Annotation
5 Transit Frames Selection
First Frame
New Object In
Object Status Shift
Large View Shift
Object Status Shift
6 Panoptic Mask Revision on Transit Frames
Multi-Frame AOT
7 Multi-Frame Panoptic Propagation on the Entire Video
8 Select Poor Masks to Revise
Refine Loop
$N \geq 3$
9 Use All Revised Masks to Propagate
Fine VPS Annotation
10 Describe Key Information in the Video
A birthday party for a kid. A lady holds a cake, people sing birthday song then the kid blows the candles.
11 Annotate Relations using Object IDs Based on the Key Description Above
candle1
child1
adult-1 holding plate-1, cake-1 on plate-1, candle-1 on cake-1
child-1 blowing candle-1: Frame 0082 – 0114
child-1 looking at cake-1: Frame 0000 – 0180
...
Video to Annotate

Figure 4. **PVSG Dataset Annotation Pipeline.** The construction of the PVSG dataset can be divided into VPS annotation and relation annotation. For VPS annotation, we select a few key frames and use an off-the-shelf video object segmentation (VOS) model AOT [55] to propagate the annotated objects to the whole video, and then perform frame-level mask fusion using the predefined layer order to obtain a coarse VPS annotation for further revision. The relations are annotated based on the description of the key information in the video.

VPS [17] and KITTI-STEP [47]. However, the relations in the self-driving scenarios are limited, which is not suitable for the PVSG task. Although the recent VIP-Seg [29] provides a more diverse VPS dataset, each video only lasts around 5 seconds, which also lacks temporal relations.

With all the rationale above, we eventually decide to combine three video sources to the PVSG dataset, which are VidOR, Ego4D-STA, and Epic-Kitchens-100 (including some videos from VISOR).

## 4.2. Dataset Statistics

Figure 2 displays the statistics of the PVSG dataset, which consists of 400 videos, including 289 third-person videos from VidOR and 111 egocentric videos from Epic-Kitchens and Ego4D. Among the videos, 62 videos feature birthday celebrations, while 35 videos center around ceremonies, providing rich content for contextual logic and reasoning. Furthermore, the dataset includes numerous videos related to sports and pets, featuring complex and diverse actions and interactions between objects. Figure 2 (c) shows the object count (including stuff) in the PVSG dataset.

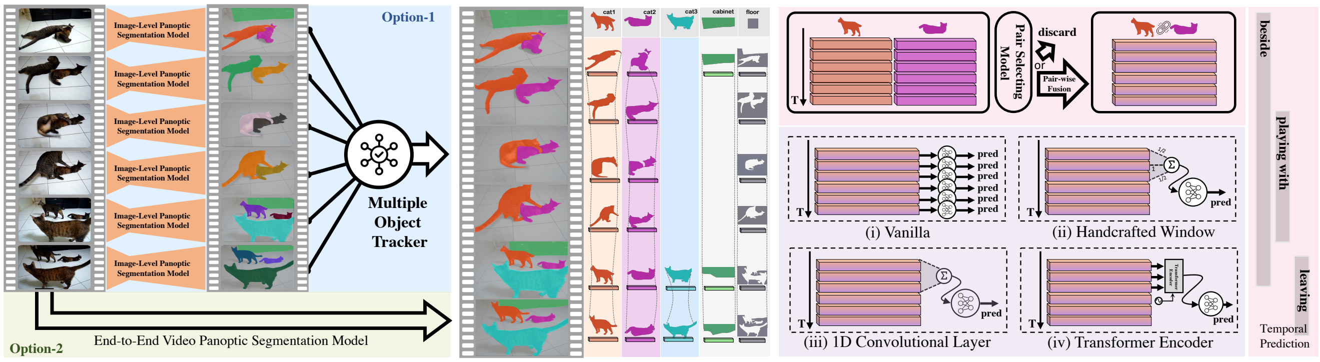
## 4.3. Dataset Construction Pipeline

Creating the PVSG dataset is never a trivial task considering that both video panoptic segmentation and relation annotations are required. This section describes how the PVSG dataset is collected and annotated.

**Step 1: Video Clip Selection** To get rid of the drawbacks of the current datasets (i.e., the unnatural videos in AG [15] without logical script, and the static and short videos from the VPS datasets), we carefully select around 300 long, daily, unedited videos with a logical storyline. In addition, to encourage the VidSGG models to be practical on egocentric videos, we also select around 100 videos from Epic-Kitchens and Ego4D with the same criteria. Videos with too many small and trivial objects are also discarded for VPS annotation purposes. We hope the selected videos could greatly encourage the exploration of video recognition, understanding, and reasoning.

**Step 2: VPS Annotation** Notice that the PVSG videos have more than 300 frames on average and 150K in total, it is impossible to annotate panoptic segmentation for each frame. After iterations and improvements, we finalize a human-machine collaborative VPS annotation pipeline, depicted in Figure 4. In a nutshell, we largely rely on an off-the-shelf VOS model called AOT [55] for the human-machine interactive annotation process.

**Coarse VPS Annotation:** With a few well-annotated object masks in the first frame, the AOT [55] is able to propagate the masks to later frames. With this strong automatic tool, we design a pipeline to obtain coarse VPS annotation. For the example video in Figure 4 (actions 1-3), we first identify several key objects to annotate and also identify key frames where the selected objects have a clear and whole appearance. To identify key objects, our annotators need to select all objects and backgrounds to address "panoptic", except those messy and unrelated ones. After annotating these key objects on their corresponding frames, we use AOT based on the frames to propagate the mask, both forward and backward. Thus, each frame will yield a whole mask video. To merge those mask videos into one, the layer order should be considered beforehand, i.e., the



**(a) Stage-1: For Feature Tube and Mask Tube Output**

**(b) Stage 2: Relation Prediction**

Figure 5. **The two-stage framework to solve the PVSG task.** The goal of the first stage is to obtain the video panoptic segmentation mask for each object, as well as its corresponding video-length feature tube. Two options are provided to achieve the goal. The second stage predicts pairwise relations based on all the feature tubes from the first stage. Four options are provided for a comprehensive comparison.

objects from which layer should be put in front. In fact, the decision of the layer order is made with keyframe selection.

**Fine VPS Annotation:** Based on the coarse VPS annotation, we conduct several rounds (more than 5) of the human-machine interactive revision process to obtain the final annotation. We rely on the multi-frame panoptic segmentation propagation mode of the AOT algorithm [55], which interpolates the entire video mask based on several frames with the entire panoptic segmentation. The quality of interpolation increases with more intermediate frames. To accelerate the revision process, we revise the transit frames first, as shown in action 5 in Figure 4. Typical examples of poor masks include incorrect tracking masks and boundaries that deviate significantly from the object.

**Step 3: Relation Annotation** We annotate temporal relations based on the VPS annotation, with object ID prepared. To guarantee the significance of the relation, we ask annotators to describe the video with several sentences and annotate relations accordingly. The relations they use are strictly within our dictionary, but we also enlarge the dictionary when necessary. Similar to the PSG dataset [52], we ask the annotators to use the most unambiguous predicate possible, i.e., "sitting on" rather than "on".

## 5. Methodology

In this section, we introduce the two-stage pipeline to address the PVSG problem. We provide two options for the first stage and four options for the second stage.

### 5.1. Stage One: Video Panoptic Segmentation

Given a video clip input $\mathbf{V} \in \mathbb{R}^{T \times H \times W \times 3}$, the goal of VPS is to segment and track each pixel in a non-overlap manner. Specifically, the model predicts a set of video clips $\{y_i\}_{i=1}^{N} = \{(\mathbf{m}_i, p_i(c))\}_{i=1}^{N}$, where $\mathbf{m}_i \in \{0,1\}^{T \times H \times W}$ denotes the tracked video mask, and $p_i(c)$ denotes the probability of assigning class $c$ to a clip $\mathbf{m}_i$. $N$ is the number of entities, which includes thing classes and stuff classes.

We present two strong baselines for the first stage of VPS processing. In particular, we adopt the state-of-the-art image segmentation baseline [3] with an extra tracker and the improved video panoptic segmentation method [26]. For the former, it processes the video frames individually. For the latter, it processes the video frames across the temporal dimension, with a nearby frame as the reference frame.

**IPS+T: Image Panoptic Segmentation With Tracker** We adopt strong Mask2Former [3] as our baseline method since it is a mask-based transformer architecture. It contains a transformer encoder-decoder architecture with a set of object queries, where the object queries interact with encoder features via masked cross-attention. Given an image **I**, during the inference, the Mask2Former directly outputs a set of object queries $\{q_i\}, i = 1, \ldots, N$, where each object query $q_i$ represent one entity. Then, two different multiple-layer perceptrons (MLPs) project the queries into two embeddings for mask classification and mask prediction, respectively. During training, each object query is matched to ground truth masks via masked-based bipartite matching.

We first fine-tune the Mask2Former on our dataset. Then, we test the model with an extra tracker [46]. In particular, we first obtain panoptic segmentation results of each frame. Then we link each frame via using UniTrack [46] for tracking to obtain the final $N$ tracked video cubes for each clip. Therefore, a query tube is obtained. For the object $i$ at the $t$-th frame, the query is noted as $q_i^t$. We use $\mathbb{Q}_i^{(t_1,t_2)}$ to denote the set of queries $\{q_i^t\}_{t=t_1}^{t_2}$, and $\mathbb{Q}_i$ denotes the query tube in the entire video.

**VPS: Video Panoptic Segmentation Baseline** For video baselines, we modify the previous state-of-the-art method Video K-Net [26] into Mask2Former framework. We first replace the backbone and neck in Video K-Net [26] with the Mask2Former feature extractor. Then we use the temporal contrastive loss to perform directly on the output queries

Table 2. **Comparison between all two-stage PVSG baselines.** We provide two options for the first stage and four options for the second stage, as described in Section 3. The results show that using the basic image-based method in the first stage with the transformer encoder in the second stage can achieve optimal recall.

| Method | | thre $= 0.5$ | | | thre $= 0.1$ | | |
|---|---|---|---|---|---|---|---|
| Stage-1 | Stage-2 | R/mR@20 | R/mR@50 | R/mR@100 | R/mR@20 | R/mR@50 | R/mR@100 |
| IPS+T [3,46] | Vanilla | 3.04 / 1.35 | 4.61 / 2.94 | 5.56 / 3.33 | 8.28 / 5.68 | 14.47 / 9.92 | 18.24 / 11.84 |
| | Handcrafted Window | 2.52 / 1.72 | 3.77 / 2.36 | 4.72 / 2.79 | 8.07 / 5.61 | 13.42 / 8.27 | 16.46 / 10.11 |
| | 1D Convolution | **3.88** / 2.55 | 5.24 / 3.29 | **6.71 / 5.36** | **10.06 / 8.98** | **14.99 / 12.21** | **18.13 / 15.47** |
| | Transformer Encoder | **3.88 / 2.81** | **5.66 / 4.12** | 6.18 / 4.44 | 9.01 / 6.69 | 14.88 / 11.28 | 17.51 / 13.20 |
| VPS [3,26] | Vanilla | 0.21 / 0.10 | 0.21 / 0.10 | 0.31 / 0.18 | 6.29 / 3.04 | 9.64 / 6.74 | 12.89 / 9.60 |
| | Handcrafted Window | **0.42** / 0.13 | 0.52 / 0.50 | 0.94 / **0.92** | 5.24 / 2.84 | 7.65 / 7.14 | 9.64 / 8.22 |
| | 1D Convolution | **0.42** / 0.25 | 0.63 / 0.67 | 0.63 / 0.67 | **8.07 / 7.84** | **11.01 / 9.78** | **12.89 / 10.77** |
| | Transformer Encoder | **0.42 / 0.61** | **0.73 / 0.76** | **1.05 / 0.92** | 6.50 / 5.75 | 9.64 / 8.25 | 12.26 / 9.51 |

from the last layer of the decoder. In particular, given two frames, we first obtained the object queries from both frames and then we sent them into an embedding layer (a shared MLP) to obtain association embeddings. We adopt the same tracking loss used in [26] to supervise the association embeddings. The embeddings are close if they are the same object, otherwise, they are pulled away.

During the training, the two nearby frames are sent to the model to learn the association embedding. During the inference, the learned association embeddings are used to perform instance-wised tracking cues to match each thing masks frame by frame in an online manner. Compared with the image baseline, our video baseline considers the temporal learned embedding. After this step, we obtain $N$ tracked video cubes for each clip. For both baselines, we also dump the corresponding object queries for further processing.

## 5.2. Stage Two: Relation Classification

The object query (feature) tubes $\{Q_i\}_{i=1}^{N}$ serve as a link between the first and second stages. As shown in Figure 5 (b), these tubes are initially formed into query pairs. For efficient training of the relation model, these pairs are then matched with their corresponding ground-truth pairs based on mask IOU values, with non-matching pairs being discarded. This selective process assigns relation labels to certain predicted query pairs during specific time spans.

**Pair Selection** It is important to note the difference in pairing selection between the training and inference phases. During training, pairs are easily selected based on their match with the ground truth. However, at inference, pairing all possible combinations would yield $N \times (N-1)$ pairs, an impractically large number. To address this challenge, we have developed a compact, trainable pairing component. This component leverages a transformer encoder to cross-attend to all other object features within each frame, thus gathering global information. Subsequently, it uses max pooling to condense the query tube $\{Q_i^t\}_{t=t_0}^{T}$ into a single token for each object. This process allows for the calculation of pair-wise similarities and the construction of a sparse pairing matrix, which is optimized towards the ground truth pairing matrix using a multi-label loss [38].

Next, we introduce four operation options to predict the relations of each feature pair.

**Vanilla: Fully-Connected Layers** Begin with the most basic version, the pairwise feature fusion is followed by three straightforward fully-connected layers on the fused features. Since some objects may have several interactions occurring simultaneously, we define the issue as a multi-label classification job with binary cross-entropy loss.

**Handcrafted Filter** To further consider the temporal information, we design a simple kernel to gather the information from the context in nearby frames. By default, the handcrafted filter is a simple vector of $[\frac{1}{4}, \frac{1}{2}, 1, \frac{1}{2}, \frac{1}{4}]$ with a window size of 5. The filter is also required in inference.

**1D-Convolutional Layer** To improve the handcrafted filter, we also utilize a learnable 1D-Convolutional layer to capture temporal information. The kernel sizes are set to 5.

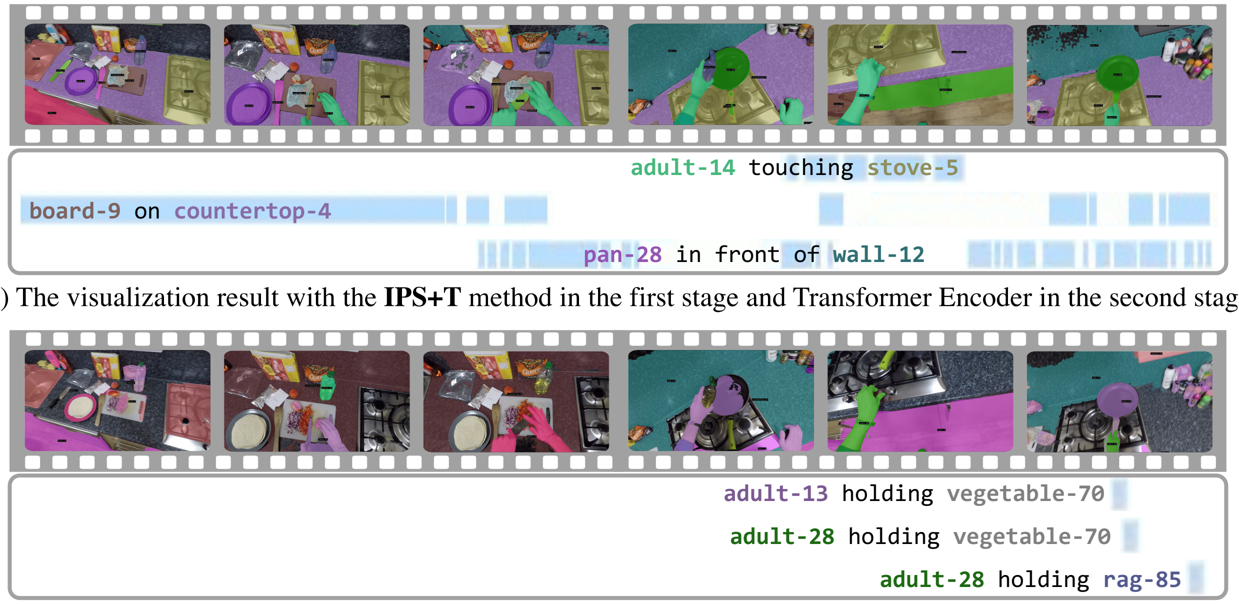
**Transformer Encoder** A transformer encoder [45] is naturally suitable for encoding the temporal data. We utilize a transformer block with positional embeddings in the entire fused query feature to capture temporal information via cross-attention between frames.

# 6. Experiments

In this section, we show the experimental results for the PVSG dataset. We split the dataset with 338 videos for training and 62 videos for testing[3]. For both IPS+T and VPS, we adopt Mask2Former [3] upon the ResNet-50 [13] backbone with 8 training epochs, both take about 48 hours on 4 V-100 GPUs. The training epoch of the second stage is set to 100, which takes about 8 hours on one V-100.

**Better Temporal Modeling Boosts Relations Prediction.** We first take a look at the second stage. The transformer encoder obtains the optimal results regardless of the first-stage options, underscoring its proficiency in synthesizing temporal information. Moreover, the 1D convolutional approach outperforms the handcrafted window method, suggesting that incorporating learnable parameters in the sec-

[3] Check the annotated videos in each split here.


adult-14 touching stove-5
board-9 on countertop-4
pan-28 in front of wall-12
adult-13 holding vegetable-70
adult-28 holding vegetable-70
adult-28 holding rag-85

(a) The visualization result with the **IPS+T** method in the first stage and Transformer Encoder in the second stage.

(b) The visualization result with the **VPS** method in the first stage and Transformer Encoder in the second stage.

Figure 6. **The visualization of the top 3 triplets generated by PVSG models.** The result shows that the IPS+T method is able to predict a better-quality video panoptic mask. The VPS baseline is shown unable to perform well on tracking (e.g., the tracking of the adult switched in the later frames), which leads to its low performance in the PVSG task. Check project page for more visual results.

ond stage can be beneficial. Notably, even the most elementary vanilla method registers some recall considering the harsh recall criteria described in Section 3. This indicates that with a decent model in the first stage, the PVSG task is indeed approachable.

**VPS Models Lag Behind IPS+T.** Moving on to the impact of the first stage, Table 2 reveals that the end-to-end VPS model appears to lag behind the IPS+T baselines. While VPS models have demonstrated their effectiveness on established datasets like Cityscape-VPS and KITTI-STEP, the PVSG dataset, characterized by its longer and more dynamic videos with frequent and significant shifts in camera view, presents novel obstacles for VPS research. This is evident in Figure 6, where the VPS models' subpar tracking ability significantly hampers their PVSG task performance. Table 2 also reflects this, particularly at a 0.1 threshold, where a minimal overlap in masks is sufficient for a recall. Here, the VPS results are nearly on par with IPS+T at R/mR@20, indicating that when the criteria for mask tube overlap are less strict, VPS can almost reach IPS+T levels, though not quite.

**Understanding Numbers.** When examining Table 2, it is crucial to prioritize the R/mR@20 as it represents our most significant indicator. The highest value for R@20 currently stands at 3.88, meaning that roughly for every 25 ground-truth triplets, one meets the criteria for a successful recall, indicating a relatively low efficiency. However, when setting the threshold to 0.1, the score improves to around 10, meaning the model can predict one in every 10 triplets with a looser requirement of recall. This suggests that while the model has some capability in recognizing key video content, there is substantial room for improvement in its accuracy and effectiveness.

## 7. Conclusion, Challenges, and Outlook

In this paper, we introduce a new PVSG task, a new PVSG dataset with several baselines to address the new task, in the hope of encouraging comprehensive video understanding and triggering more interesting downstream tasks such as visual reasoning. Here we discuss the challenges and future work.

**Challenges** Real-world data often exhibit long-tailed distributions across objects and relations, as shown in Figure 2. The PVSG models are expected to predict informative and diverse relations, rather than being obsessed with statistically common relations. Yet another challenge the PVSG models face is the uncertainty in relation descriptions. For example, "playing with" can be overlapping with "chasing" when it describes two kids chasing each other. Another important challenge is that the PVSG models seem to largely rely on video panoptic segmentation. With the video with a large view shift, the VPS models are expected to have a better performance on tracking and segmentation. Additionally, the time span prediction is a critical aspect of the PVSG model. A more sophisticated time-series technique could benefit the model development.

**Outlook on Video Perception and Reasoning** We foresee the potential of PVSG in bridging video scene perception and reasoning. While current video question-answering datasets lack pixel-level segmentation masks that refine (sometimes determine) the relations between object pairs, the inclusion of such dense annotations could be critical to video reasoning tasks. In fact, the PVSG dataset also provides dense captioning and question-answering annotation for each video, which could benefit the topic of reasoning and conversational chatbots. PVSG is related to social intelligence, with rich event annotations in human behaviors and dynamics. In this spirit, the PVSG models might be critical to embodied agent tasks or virtual reality techniques, as the egocentric data is especially highlighted in the dataset.

**Potential Negative Societal Impacts** This work releases a dataset containing human behaviors, posing possible gender and social biases inherently from data. Potential users are encouraged to consider the risks of overlooking ethical issues in imbalanced data, especially in underrepresented minority classes.

**Acknowledgement** This study is supported by the Ministry of Education, Singapore, under its MOE AcRF Tier 2 (MOE-T2EP20221-0012), NTU NAP, and under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s). We are also grateful to SuperAnnotate[4] for providing an outstanding annotation platform and excellent customer service. Special thanks to Binzhu Xie and Zitang Zhou from the Beijing University of Posts and Telecommunications for their dedicated leadership of the annotation team.

**Author Contributions** This paper represents a collaborative effort led by two principal contributors, **JY** and **WP**. **JY**, as the project leader, played a pivotal role in the initiative, monitoring and guiding each aspect, and managing both project and annotation teams, plus relation modeling development. **WP** significantly contributed by spearheading preliminary research, developing the PVSGAnnotation codebase, and crafting the video panoptic segmentation stage, which is fundamental to the OpenPVSG framework. The team was further supported by an experienced technical consultant, **XL**, on video segmentation details, and the academic consultants, **KZ** and **ZL** on project direction. Regular discussions and writing assistance were provided by **ZG**, **LC**, **BL**, and **MZ**. The project was further enriched by the invaluable resources and guidance from three mentors, **WZ**, **CCL**, and **ZL**, whose contributions were crucial to the project's success.

[4] https://www.superannotate.com/

## References

[1] Jin Chen, Xiaofeng Ji, and Xinxiao Wu. Adaptive image-to-video scene graph generation via knowledge reasoning and adversarial learning. 2022. 1, 3

[2] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *CVPR*, 2020. 3

[3] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. 2022. 6, 7

[4] Yuren Cong, Michael Ying Yang, and Bodo Rosenhahn. Reltr: Relation transformer for scene graph generation. *arXiv preprint arXiv:2201.11460*, 2022. 2

[5] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 3

[6] 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 *ECCV*, 2018. 4

[7] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Epic-kitchens-100. *International Journal of Computer Vision*, 130:33–55, 2022. 2, 15

[8] Ahmad Darkhalil, Dandan Shan, Bin Zhu, Jian Ma, Amlan Kar, Richard Ely Locke Higgins, Sanja Fidler, David Fouhey, and Dima Damen. Epic-kitchens visor benchmark: Video segmentations and object relations. In *NeurIPS*, 2022. 2, 12, 13

[9] Linsen Dong, Guanyu Gao, Xinyi Zhang, Liangyu Chen, and Yonggang Wen. Baconian: A unified open-source framework for model-based reinforcement learning, 2021. 3

[10] Amy Fire and Song-Chun Zhu. Learning perceptual causality from video. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 7(2):1–22, 2015. 1

[11] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017. 4

[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. 2, 4, 12, 13, 15

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 7

[14] Samitha Herath, Mehrtash Harandi, and Fatih Porikli. Going deeper into action recognition: A survey. *Image and vision computing*, 60:4–21, 2017. 1

[15] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *CVPR*, 2020. 1, 2, 5, 12

[16] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *CVPR*, 2015. 1

[17] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9859–9868, 2020. 2, 5

[18] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Video panoptic segmentation. In *CVPR*, 2020. 3

[19] Dahun Kim, Jun Xie, Huiyu Wang, Siyuan Qiao, Qihang Yu, Hong-Seok Kim, Hartwig Adam, In So Kweon, and Liang-Chieh Chen. Tubeformer-deeplab: Video mask transformer. In *CVPR*, 2022. 3

[20] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9404–9413, 2019. 2

[21] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2017. 2

[22] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L Berg. Tvqa: Localized, compositional video question answering. *arXiv preprint arXiv:1809.01696*, 2018. 12, 13

[23] Lin Li, Long Chen, Yifeng Huang, Zhimeng Zhang, Songyang Zhang, and Jun Xiao. The devil is in the labels: Noisy label correction for robust scene graph generation. In *CVPR*, 2022. 2

[24] 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

[25] Xinghang Li, Di Guo, Huaping Liu, and Fuchun Sun. Embodied semantic scene graph generation. In *Conference on Robot Learning*, pages 1585–1594. PMLR, 2022. 1, 3

[26] Xiangtai Li, Wenwei Zhang, Jiangmiao Pang, Kai Chen, Guangliang Cheng, Yunhai Tong, and Chen Change Loy. Video k-net: A simple, strong, and unified baseline for video segmentation. In *CVPR*, 2022. 3, 6, 7

[27] Chenchen Liu, Yang Jin, Kehan Xu, Guoqiang Gong, and Yadong Mu. Beyond short-term snippet: Video relation detection with spatio-temporal global context. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10840–10849, 2020. 3

[28] Zelun Luo, Wanze Xie, Siddharth Kapoor, Yiyun Liang, Michael Cooper, Juan Carlos Niebles, Ehsan Adeli, and Fei-Fei Li. Moma: Multi-object multi-actor activity parsing. *NeurIPS*, 2021. 4

[29] Jiaxu Miao, Xiaohan Wang, Yu Wu, Wei Li, Xu Zhang, Yunchao Wei, and Yi Yang. Large-scale video panoptic segmentation in the wild: A benchmark. In *CVPR*, 2022. 2, 3, 5, 12, 13

[30] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. In *ICCV*, 2019. 4, 12, 13

[31] Beng Chin Ooi, Kian-Lee Tan, Anthony Tung, Gang Chen, Mike Zheng Shou, Xiaokui Xiao, and Meihui Zhang. Sense the physical, walkthrough the virtual, manage the metaverse: A data-centric perspective. *arXiv preprint arXiv:2206.10326*, 2022. 4

[32] Xufeng Qian, Yueting Zhuang, Yimeng Li, Shaoning Xiao, Shiliang Pu, and Jun Xiao. Video relation detection with spatio-temporal graph. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 84–93, 2019. 3

[33] Siyuan Qiao, Yukun Zhu, H. Adam, A. Yuille, and Liang-Chieh Chen. Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. *CVPR*, 2021. 3

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 2

[35] Xindi Shang, Donglin Di, Junbin Xiao, Yu Cao, Xun Yang, and Tat-Seng Chua. Annotating objects and relations in user-generated videos. In *ICMR*, 2019. 2, 4, 12, 13

[36] Xindi Shang, Tongwei Ren, Jingfan Guo, Hanwang Zhang, and Tat-Seng Chua. Video visual relation detection. In *ACM MM*, 2017. 2, 3, 4, 12

[37] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *ECCV*, 2016. 2, 4

[38] Jianlin Su, Mingren Zhu, Ahmed Murtadha, Shengfeng Pan, Bo Wen, and Yunfeng Liu. Zlpr: A novel loss for multi-label classification. *arXiv preprint arXiv:2208.02955*, 2022. 7

[39] Zixuan Su, Xindi Shang, Jingjing Chen, Yu-Gang Jiang, Zhiyong Qiu, and Tat-Seng Chua. Video relation detection via multiple hypothesis association. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 3127–3135, 2020. 3

[40] Mohammed Suhail, Abhay Mittal, Behjat Siddiquie, Chris Broaddus, Jayan Eledath, Gerard Medioni, and Leonid Sigal. Energy-based learning for scene graph generation. In *CVPR*, 2021. 2

[41] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *CVPR*, 2019. 2

[42] Zongheng Tang, Yue Liao, Si Liu, Guanbin Li, Xiaojie Jin, Hongxu Jiang, Qian Yu, and Dong Xu. Human-centric spatio-temporal video grounding with visual transformers. *IEEE TCSVT*, 2021. 4

[43] Yao Teng, Limin Wang, Zhifeng Li, and Gangshan Wu. Target adaptive context aggregation for video scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13688–13697, 2021. 1, 3

[44] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 2016. 2, 4

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 7

[46] Zhongdao Wang, Hengshuang Zhao, Ya-Li Li, Shengjin Wang, Philip HS Torr, and Luca Bertinetto. Do different tracking tasks require different appearance models? *NeurIPS*, 2021. 6, 7

[47] Mark Weber, Jun Xie, Maxwell Collins, Yukun Zhu, Paul Voigtlaender, Hartwig Adam, Bradley Green, Andreas Geiger, Bastian Leibe, Daniel Cremers, et al. Step: Segmenting and tracking every pixel. *arXiv preprint arXiv:2102.11859*, 2021. 2, 5

[48] M. Weber, J. Xie, M. Collins, Yukun Zhu, P. Voigtlaender, H. Adam, B. Green, A. Geiger, B. Leibe, D. Cremers, Aljosa Osep, L. Leal-Taixé, and Liang-Chieh Chen. Step: Segmenting and tracking every pixel. *NIPS*, 2021. 3

[49] Sanghyun Woo, Dahun Kim, Joon-Young Lee, and In So Kweon. Learning to associate every segment for video panoptic segmentation. In *CVPR*, 2021. 3

[50] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *CVPR*, 2017. 1, 2

[51] Li Xu, Haoxuan Qu, Jason Kuen, Jiuxiang Gu, and Jun Liu. Meta spatio-temporal debiasing for video scene graph generation. In *European Conference on Computer Vision*, pages 374–390. Springer, 2022. 1, 3

[52] 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. 1, 2, 3, 6

[53] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *ECCV*, 2018. 1

[54] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *ICCV*, 2019. 3

[55] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation. In *NeurIPS*, 2021. 5, 6

[56] Yu Yao, Mingze Xu, Chiho Choi, David J Crandall, Ella M Atkins, and Behzad Dariush. Egocentric vision-based future vehicle localization for intelligent driving assistance systems. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9711–9717. IEEE, 2019. 4

[57] Haobo Yuan, Xiangtai Li, Yibo Yang, Guangliang Cheng, Jing Zhang, Yunhai Tong, Lefei Zhang, and Dacheng Tao. Polyphonicformer: Unified query learning for depth-aware video panoptic segmentation. In *ECCV*, 2022. 3

[58] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, 2018. 2

[59] Jingzhe Zhang, Lishuo Zhuang, Yang Wang, Yameng Zhou, Yan Meng, and Gang Hua. Video demo: An egocentric vision based assistive co-robot. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 48–49, 2013. 4

[60] Yiwu Zhong, Jing Shi, Jianwei Yang, Chenliang Xu, and Yin Li. Learning to generate scene graph from natural language supervision. In *ICCV*, 2021. 2

[61] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *Proceedings of the European conference on computer vision (ECCV)*, pages 803–818, 2018. 1

# A. Implementation Details

All experiments are performed in a single unified codebase called OpenPVSG, using the `MMDetection` framework to facilitate reproducibility.

## A.1. IPS+T

**Fine-tuning the Image Panoptic Segmentation Model** We first fine-tune the Mask2Former model (ResNet50 backbone) on the panoptic segmentation annotations from our PVSG dataset. Here we treat the PVSG dataset as an image dataset and process all video frames individually. The model is initialized from the best performing COCO-pretrained weights provided by `MMDetection` and then trained using a batch size of 32. The AdamW optimizer is used with a learning rate of 0.0001, weight decay of 0.05, and gradient clipping with a max L2 norm of 0.01. The learning rate is multiplied by 0.1 for the backbone, and the weight decay is set to 0.0 for embedding layers. Training runs for 8 epochs.

**Mask Association with Tracker** With both panoptic segmentation masks and the corresponding query features obtained from the fine-tuned Mask2Former model above, we then adopt the UniTrack model to associate masks in each frame to get the panoptic mask tubes and query feature tubes for each video clip. We configure the tracker using Unitrack's default config (config/imagenet_resnet18_s3_womotion.yaml)[5] of Multi-Object Tracking and Segmentation (MOTS) setting and load pre-trained weights of their provided image-based SSL model `MoCoV1-ResNet50`, which has the best performance in MOTS task.

## A.2. VPS

**Fine-tuning the Video Panoptic Segmentation Model** We utilize Video K-Net implemented on a Mask2Former backbone as our VPS model, and train it using video panoptic segmentation annotations from our PVSG dataset. Optimal COCO-pretrained weights obtained from `MMDetection` are used to initialize the Mask2Former model.

## A.3. Relation Modeling

**Relation Dataset Formation** After completing the training of IPS+T and VPS models, we extract predicted feature tubes for each entity in the training videos. These tubes are segmented into individual frames for relation analysis. Specifically, within each frame, if both the subject and object predicted masks have a mask IOU greater than 0.5, we establish a relation between the pair. Subsequently, we map the relation annotations from ground truth pairs to these predicted pairs, forming the basis for the secondary stage of training.

**Training** The training process begins with the use of two transformer models, one for the subject and one for an object, designed to encode each entity. This encoding ensures that each entity feature is enriched with contextual information from other entities in the frame. For the pair-selection model, we employ max pooling to distill the object feature tube into a singular object token. We then calculate cosine similarity to form a pairing matrix, indicative of potential relations. This matrix is contrasted with a ground truth matrix for supervision. In relation prediction, a multi-label loss computation is applied exclusively to pairs with established relations. The training is conducted with a batch size of 32, utilizing an Adam Optimizer with a learning rate of 0.001.

# B. Details of PVSG Formation

In Section 4, we discuss several existing video datasets related to the PVSG dataset. In this section, we would like to elaborate on their characteristics and highlight our main considerations when building the PVSG dataset.

## B.1. Video Selection and Focus

We will first discuss some typical video datasets that are closely related to the PVSG task, and explain how we choose video sources to compose the PVSG dataset. We especially pay attention to the videos we think are better suited to explore contextual logic and reasoning. Here is a list of candidate datasets.

**Third-Person-View (TPV) video candidates** include VidSGG datasets (e.g., ImageNet-VidVRD [36], VIDOR [35] and Action Genome [15]), video understanding dataset (e.g., TVQA [22], Howto100M [30]), and video panoptic segmentation dataset from VIP-Seg [29].

**Egocentric video candidates** include Ego4D-STA [12], VISOR [8].

**ImageNet-VidVRD [36]: Short and Static Videos** Although it is the first dataset to study video visual relations, most of VidVRD's video clips last only a few seconds and have almost static scenes. While it is a useful dataset for detecting relations between objects, additional research in logic and reasoning through temporal relation changes in videos may not be possible.

**Action Genome [15]: Lack of Logic between Actions, Heavy Traces of Performance** With dense relation annotations, Action Genome is commonly used in video scene graph generation task in recent years. Videos in Action Genome source from Charades dataset, which was made by 267 different users acting out certain sentences constructed by objects and actions from a fixed vocabulary. While some

[5] https://github.com/Zhongdao/UniTrack/

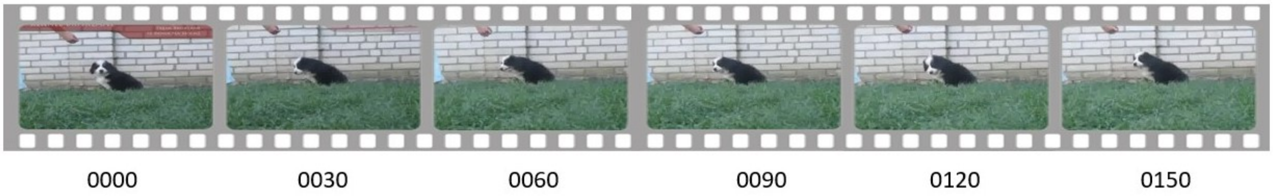


Figure A1. **An ImageNet-VidVRD example.** The dog and the scene in this 7-second video barely change from start to finish.

videos contain multiple actions and dynamic scenes, the transitions between these actions show heavy traces of performance. Due to the nature of the generated scripts, the whole video is more like a splicing of some instructed verbs. Consequently, there is no clear logic in the sequence of actions for observers to comprehend the video. In addition, simple videos with one or two actions made up a certain portion of this dataset. Therefore, we think Action Genome is not suitable for the PVSG task.

**VIDOR [35]: A Good Candidate** VIDOR is a large-scale dataset with all videos collected from user-uploaded videos on Flickr. Most of the videos are unedited records of daily life scenes, which ensures the coherence of the video plot and the natural connection of action changes. While there are useful videos in VIDOR, most of the videos contain too simple relations and some have ambiguous content. Therefore, we carefully select a subset of videos from VIDOR that fulfill our requirements to form the third-person view part of the PVSG dataset. The explanation of the good video in VIDOR is shown in Figure A4.

**Detailed Video Selecting Rules**

- Selected videos should have main characters and contain a sequence of consistent actions (relations).
- Selected videos need to be comprehensible by observers.
- Discard videos with too many trivial and small objects for annotation purposes.

**TVQA [22]: Too many cut shots** TVQA is a frequently used dataset in Video Question Answering tasks. Since VQA also intends to study logical reasoning in videos, datasets in this domain lie in the relevant scope of the PVSG. However, videos from the TV show/movie datasets like TVQA contain lots of cut shots, which make it challenging to associate and relate objects across discontinuous scenes. Moreover, understanding such videos usually relies heavily on contextual information in the show. Hence, datasets like TVQA cannot apply to the PVSG task. Figure **??** shows an example for reference.

**VIP-Seg [29]: VPS but Static Videos** VIP-Seg is a large-scale video panoptic segmentation dataset. Despite the fine annotations, the video scene is generally static and includes only one action due to the average video length of 5 seconds. Figure A6 shows an example for reference.

**Howto100M [30]: Curated videos with many cut shots** HowTo100M is a large-scale dataset of instructional videos where complex tasks are broken into steps for the audience to learn. Clear logic can be found in such a dataset when a series of actions are made to address a specific task. However, one video contains many cut shots and is filmed from different angles (edited by the video creator). Hence, Howto100M does not fit for PVSG task. Figure A7 shows an example for reference.

**Egocentric Videos: Suitable for PVSG** The PVSG dataset also contains egocentric videos, with the scope that the models should tackle the problems for both third-view videos and egocentric videos. Two high-quality egocentric datasets, Ego4D [12] and VISOR [8] are good candidates for selection. In fact, egocentric videos are usually token when the actors are performing a specific task. The task usually contains a chain of actions with inherently clear logic. Therefore, egocentric videos are suitable for the reasoning task, and exploring scene graph generation based on these videos might invite a variety of techniques including perception and reasoning. Figure A8 shows an example for reference.

## B.2. Annotation Quality

In this section, we will focus on our selected datasets discussed in Section B.1, compare their original annotations with the PVSG annotations, and analyze the necessity of panoptic segmentation for video scene graph generation and video reasoning.

**Comparison between VIDOR** Figure A9 shows the comparison between VIDOR and our PVSG dataset. We first observe the drawbacks of the VIDOR. 1) It contains inconsistent bounding box annotations, e.g., the child is not cropped out at the first frame; 2) It misses important details, with no candle annotation, resulting in missing relations that are important to understand the scene. Actually,

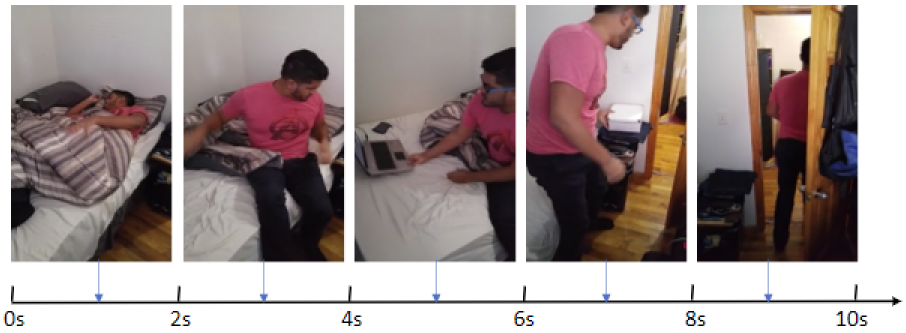


Figure A2. **An Action Genome example.** In 10 seconds, the man quickly makes several actions: gets up from the bed, puts on his glasses, grabs and glances at the computer on the bed in 1 second, gets up and picks up the box, then walks out the door.

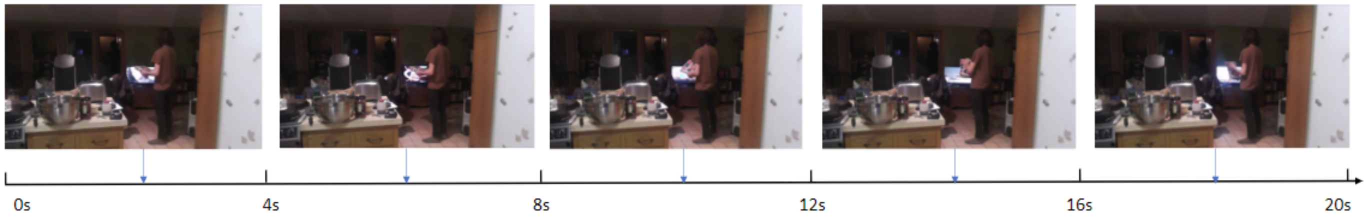


Figure A3. **Another Action Genome example.** In 20 seconds, the man keeps standing in front of the TV and holding a book in his hand.

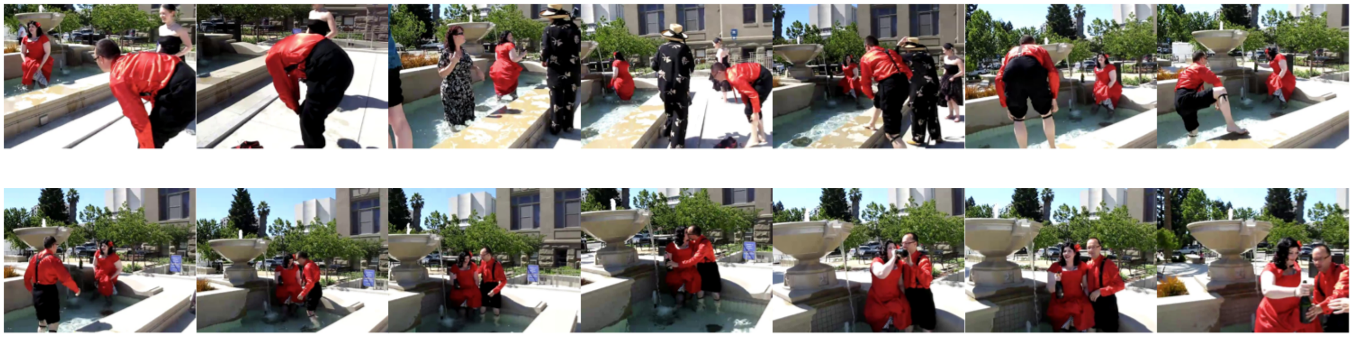


Figure A4. **A VIDOR good video example (demo video).** At the beginning of the video, the man is pulling up his pants, the woman is holding up her skirt and standing in the water while the woman in black is taking pictures. We can guess from their outfits and the surroundings that this is a wedding photo shoot scene. Next, the man walks into the water and poses for photos with the woman. They hug each other, kiss each other, and drink liquor as the crowd cheers.
There's a sequence of actions in this one-minute demo video, with natural logical relationships between these actions. Both rich actions and coherent plots make this video more understandable and predictable. We think videos like this can make dynamic scene graphs better connected and attach more significance to the PVSG task.

bbox is hard to annotate such small details; 3) There are many overlaps among different bounding boxes, one bounding box contains multiple object features; 4) Relation annotations only have simple predicates such as ”watch”, ”hold” and many prepositions in the example frame, which cannot really describe what is happening in this frame; 5) Bounding box cannot annotate stuff such as water and ground. Actually, they also play an important role to understand the scene.

With all the considerations above, we annotate the PVSG dataset with great caution, such as 1) having a consistent annotation for all the objects. Once they are decided to be annotated, they will be annotated throughout the video; 2) we carefully design the object vocabulary beforehand after watching all the videos in the dataset to ensure all important objects are annotated; 3) panoptic segmentation avoids

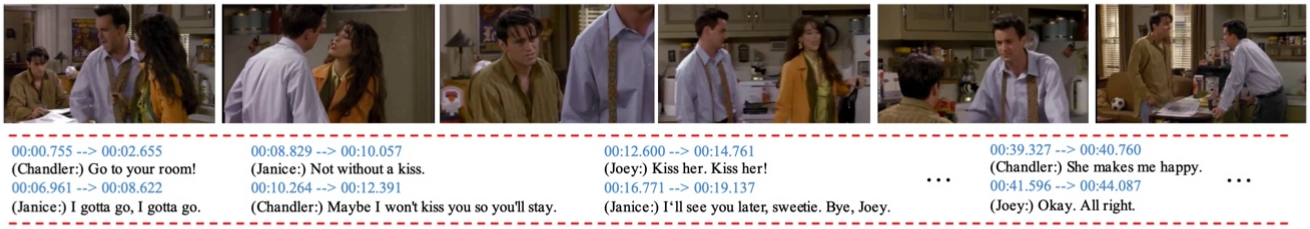

00:00.755 --> 00:02.655
(Chandler:) Go to your room!
00:06.961 --> 00:08.622
(Janice:) I gotta go, I gotta go.
00:08.829 --> 00:10.057
(Janice:) Not without a kiss.
00:10.264 --> 00:12.391
(Chandler:) Maybe I won't kiss you so you'll stay.
00:12.600 --> 00:14.761
(Joey:) Kiss her. Kiss her!
00:16.771 --> 00:19.137
(Janice:) I'll see you later, sweetie. Bye, Joey.
...
00:39.327 --> 00:40.760
(Chandler:) She makes me happy.
00:41.596 --> 00:44.087
(Joey:) Okay. All right.
...

Figure A5. **TVQA example** Cut shots appear every few seconds in TVQA videos.
**??**

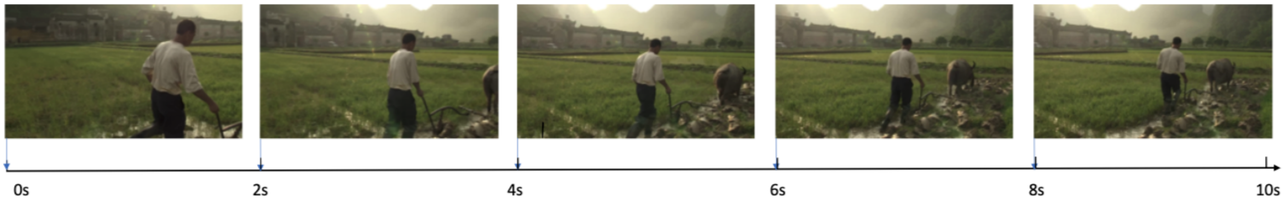
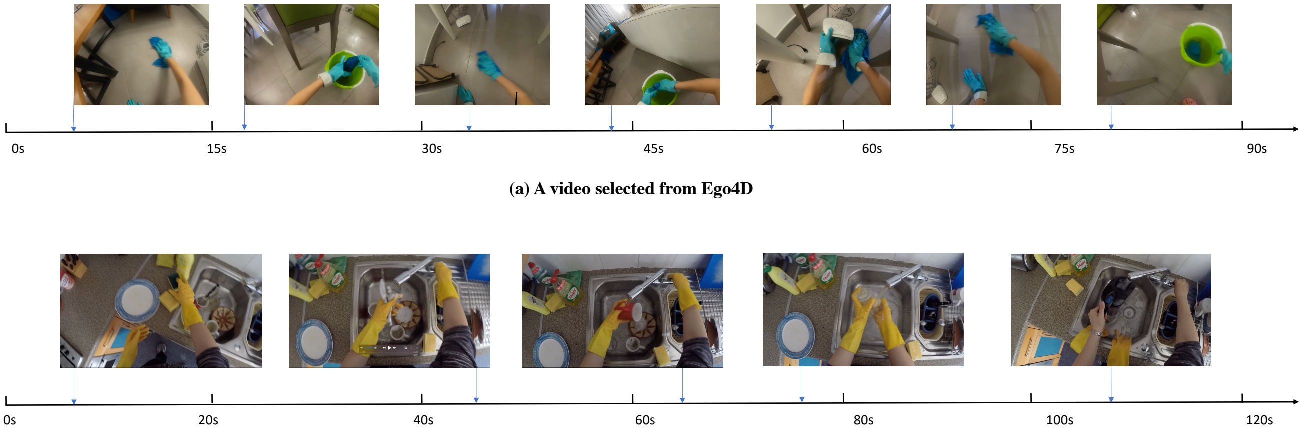


Figure A6. **A VIP-Seg example.** The man keeps on plowing in this 10-second video clip.

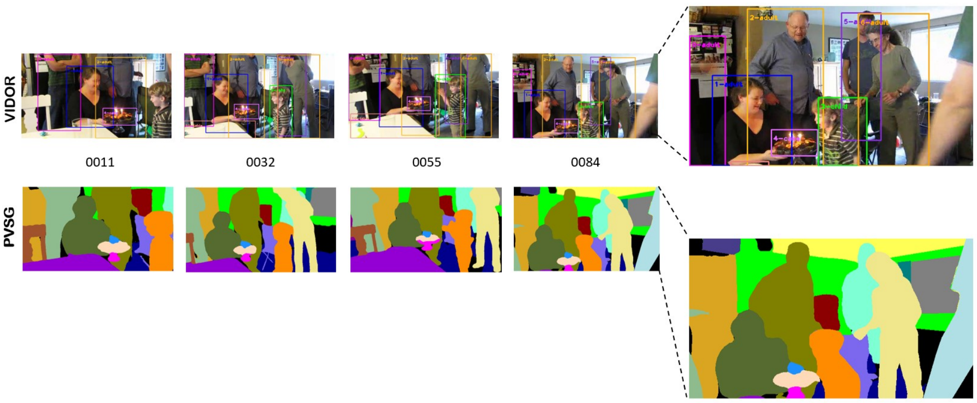
Figure A7. **An Howto100M example.** A girl is teaching people how to care for American Girl Doll's hair.



**(a) A video selected from Ego4D**

**(b) A video selected from VISOR**

Figure A8. **Examples from Ego4D [12] and VISOR [7].** Both of these egocentric datasets are long videos (over 1 min), and the actors are doing specific tasks such as cooking and cleaning. Therefore, these videos inherently contain abundant contextual logic.



adult-1 hold cake-4; adult-2 speak to child-3; adult-2 watch cake-4;
child-3 watch cake-4; adult-5 watch child-3; adult-6 watch cake-4

adult-0 behind adult-1; adult-0 next to adult-2; adult-0 in front of child-3
adult-0 in front of adult-5; adult-0 in front of adult-6; adult-0 next to table-10;
adult-1 in front of adult-0; adult-1 next to adult-2; adult-1 in front of child-3;
adult-1 next to cake-4; adult-1 in front of adult-5; adult-1 in front of adult-6;
adult-1 next to table-10; adult-2 in front of adult-0; adult-2 next to adult-1;
adult-2 next to child-3; adult-2 next to cake-4; adult-2 in front of adult-5;
adult-2 next to adult-6; child-3 in front of adult-0; child-3 in front of adult-1;
child-3 in front of adult-2; child-3 next to cake-4; child-3 in front of adult-5;
child-3 in front of adult-6; child-3 next to table-10; cake-4 in front of adult-0;
cake-4 in front of adult-1; cake-4 in front of adult-2; cake-4 in front of child-3;
cake-4 in front of adult-5; cake-4 in front of adult-6; cake-4 next to table-10;
adult-5 next to adult-2; adult-5 behind child-3; adult-5 behind adult-6;
adult-6 in front of adult-0; adult-6 in front of adult-2; adult-6 behind child-3;
adult-6 in front of adult-5; table-10 next to adult-0; table-10 in front of adult-1;
table-10 in front of adult-2; table-10 in front of child-3; table-10 next to cake-4;
table-10 in front of adult-5; table-10 in front of adult-6

child-2 looking at candle-5;
child-2 blowing candle-5;
candle-5 on cake-7;
adult-1 holding bowl-6;
cake-7 on bowl-6;
adult-1 looking at child-2;
adult-12 touching chair-4;
adult-12 looking at child-2;
child-2 in front of chair-4;
chair-4 in front of adult-12;

Figure A9. **Comparison between VIDOR and PVSG annotations.** The VIDOR annotation uses bounding boxes to annotate objects in the videos. It is shown that some important objects are not annotated, such as candles. Without candles, important relations such as the kid blowing the candles can not be annotated too. In the PVSG dataset, the problem is solved by carefully defining the object classes. The high demand for our panoptic segmentation annotation also solves problems like the kid not being cropped out in Frame 0011. For relation annotation, the VIDOR contains many positional relations. However, most of the positional relations can be figured out in the static images, but the PVSG annotation highlights the dynamic relations in the video,

overlapping masks; 4) when annotating the relations, we focus on the dynamic relations rather than having many positional relations; 5) Panpotic segmentation is able to annotate the background that are critical to comprehensive scene understanding.