

# Helping Hands: An Object-Aware Ego-Centric Video Recognition Model

Chuhan Zhang  
 VGG, University of Oxford  
 czhang@robots.ox.ac.uk

Ankush Gupta  
 Google DeepMind, London  
 ankushgupta@google.com

Andrew Zisserman  
 VGG, University of Oxford  
 az@robots.ox.ac.uk

## Abstract

We introduce an object-aware decoder for improving the performance of spatio-temporal representations on ego-centric videos. The key idea is to enhance object-awareness during training by tasking the model to predict hand positions, object positions, and the semantic label of the objects using paired captions when available. At inference time the model only requires RGB frames as inputs, and is able to track and ground objects (although it has not been trained explicitly for this).

We demonstrate the performance of the object-aware representations learnt by our model, by: (i) evaluating it for strong transfer, i.e. through zero-shot testing, on a number of downstream video-text retrieval and classification benchmarks; and (ii) by using the representations learned as input for long-term video understanding tasks (e.g. Episodic Memory in Ego4D). In all cases the performance improves over the state of the art—even compared to networks trained with far larger batch sizes. We also show that by using noisy image-level detection as pseudo-labels in training, the model learns to provide better bounding boxes using video consistency, as well as grounding the words in the associated text descriptions.

Overall, we show that the model can act as a drop-in replacement for an ego-centric video model to improve performance through visual-text grounding<sup>1</sup>.

## 1. Introduction

In visual-language models there has been a recent move to explicitly build object awareness into the vision module by adding specialized and bespoke components, or using entirely object-centric architectures. The motivation for this partly comes from the attractive compositional nature of objects and their inter-relationships in language, which enables inexhaustible novel combinations [10, 45], and partly from infant cognitive studies that stress the im-

portance of objects in early visual development [29, 56, 60]. Examples in the video domain include explicit internal object representations [2], e.g., through RoI-align [17] pooled features either from a pre-trained region-proposal network (RPN) [2, 52, 57, 62], or from bounding-box coordinates taken as input [19, 42, 48, 71]. This contrasts with the large body of work where standard representations are learnt end-to-end without any explicit factorization into objects/entities, such as dual-encoder vision-language models in the image [21, 49] and video domains [4, 64].

In this paper, we take a different (middle) path and instead use a vanilla video transformer architecture and induce object-awareness into the video representation by tasking the model to predict object-level properties, such as their localization and semantic categories, only during training.

Our target domain is ego-centric video [11, 16], and we tailor the object properties used to this. In ego-centric videos the actor [57] is often present through their hands, and we therefore task the network to predict both the hands and the principal objects they interact with. As will be seen, this simple object-aware training boosts the performance of pre-trained video-language architectures significantly, and leads to state-of-art performance across multiple ego-centric benchmarks. During inference, the model requires only RGB frames as input, and operates as a standard video-language network.

In more detail, our model is built on top of a pre-trained video-language dual encoder architecture (where there are separate encoders for the video and text data). We add an additional, but vanilla, transformer decoder head [61], and train with DETR/Mask2former [7, 9] query vectors and object loss for hands and other objects. The intuition is that these additional query vectors help the model to attend to and track the hands and salient objects in the scene (these are the ‘helping hands’ of the paper title). Importantly, we do not require dense frame level ground truth for this training. Rather, we obtain somewhat **noisy** and temporally **sparse** annotations automatically from a hand object detector [53], and use these to provide prediction targets for the frames where they are available. This *opportunistic* use of annotations is pragmatic as object detectors trained on third-

<sup>1</sup>Code and models available at: [https://github.com/Chuhanxx/helping\\_hand\\_for\\_egocentric\\_videos](https://github.com/Chuhanxx/helping_hand_for_egocentric_videos)

person datasets (such as COCO) do not perform so well on the ego-centric domain, where the scenes are more crowded and objects are often small and can be motion blurred. By only requiring annotations for a subset of frames, where they can be reliably produced automatically, **we are able to train on large-scale data without requiring expensive manual supervision.**

Although we train with noisy and sparse image-level object localization, our model can learn to predict better and denser bounding-box trajectories through large-scale training due to the spatio-temporal consistency which naturally presents in videos. Also, it is able to predict semantic grounding by learning to map the object appearance to the nouns in the video captions.

It is worth noting that we are using hand detectors because hands are a common and important object in ego-centric videos. However, the object-centric method we are proposing has greater scope than ego-centric videos and can be applied to other scenarios with other object types providing the ‘helping-hand’.

In summary, we make the following contributions: (i) We propose a method to induce object-awareness in video-language models for an architecture composed of standard neural modules. The model only requires RGB frames as inputs, and thus is a drop-in replacement for any ego-centric video model.

(ii) The model can be trained opportunistically using available and sparse frame-level and noisy annotations, produced automatically.

(iii) We demonstrate state-of-the-art strong (zero-shot) transfer for cross-modal retrieval to other ego-centric datasets namely, EpicKitchens-MIR and EGTEA improving prior art by 2-4%.

(iv) We evaluate the grounding quantitatively using the EpicKitchens-VISOR dataset [11, 12] and find that the model outperforms the base hand-object detector used for training supervision.

(v) Finally, we also demonstrate that the representations learned can be used as input in long-term video understanding tasks like EgoNLQ and EgoMQ. The objectiveness in the representation helps the model outperform other models trained on the same training set on these two tasks.

## 2. Related Work

**Vision and Language Representation Learning.** Different from transferring representations learned for classification on a fixed set of object categories [27, 54], recent vision-language pre-training (VLP) works leverage large-scale supervision from free-form text descriptions of images and videos. These methods use image captions [49] or video sub-titles [44, 64] with either independent dual encoders for the visual and text modalities [4, 21, 41], or via

joint encoders with cross-attention across modalities [1, 30, 31, 69]. We also use dual-encoders which are kept frozen due to compute limitations. To explicitly build-in object-awareness into *image* representations, object-level features extracted from pre-trained object-detectors are aligned with the text descriptions [8, 33, 40, 58, 75, 78]. The object-level text alignment is further augmented with the object-box prediction task for grounding in [13, 23, 70, 73]. *VLP for ego-centric videos* has recently been explored [35, 77] to bridge the domain gap between representations learned from third-person videos found commonly [44], and first-person ego-centric videos. We further extend image based object-aware VLP methods to *ego-centric videos* by training to predict the auto-generated hand-object boxes extracted from pre-trained detectors [53], while requiring only RGB input during inference, making our model a drop-in replacement for ego-centric VLP models albeit with enhanced object-awareness.

**Weakly Supervised Video Text Grounding.** A key challenge for grounding in videos is the lack of large-scale object-level annotations for videos. While such annotations are readily available for synthetic datasets [68], expensive manual annotation is required for real videos.

Hence, weakly supervised methods have been developed which leverage the video sub-titles/descriptions to map nouns/verbs to regions in frames. This is typically achieved by first extracting bounding-box/segmentation regions from pre-trained detectors for objects and humans, and aligning them with keywords using max-margin [79] or contrastive [32] objectives. Similarly, [43] also align words in the video captions to regions from pre-trained RPN by modelling the region-word associations as latent variables of a conditional generative variational auto-encoder. More recently, [59] use cross-attention across text and candidate regions, and find (soft-) associations based on attention weights. While our model similarly absorbs object bounding-boxes and category information obtained from pre-trained object and hand detectors during training to induce object-aware representations, these detectors are not required during inference.

**Object-Oriented Learning in Videos.** Most vision tasks for images involve objects [22, 24, 25, 36]. In videos, there are also a broad range of object-oriented tasks: some works treat learning object-level information as an end task, they design models to predict object bounding-boxes and masks [5, 26, 37, 38, 65, 67, 74]; other works use object knowledge to achieve some other high-level tasks, for instance, object-centric scene generation [28, 66] and decomposition [14, 18, 55], and action recognition [3, 19, 20, 47, 62, 71, 80]. Our method uses object information to as-

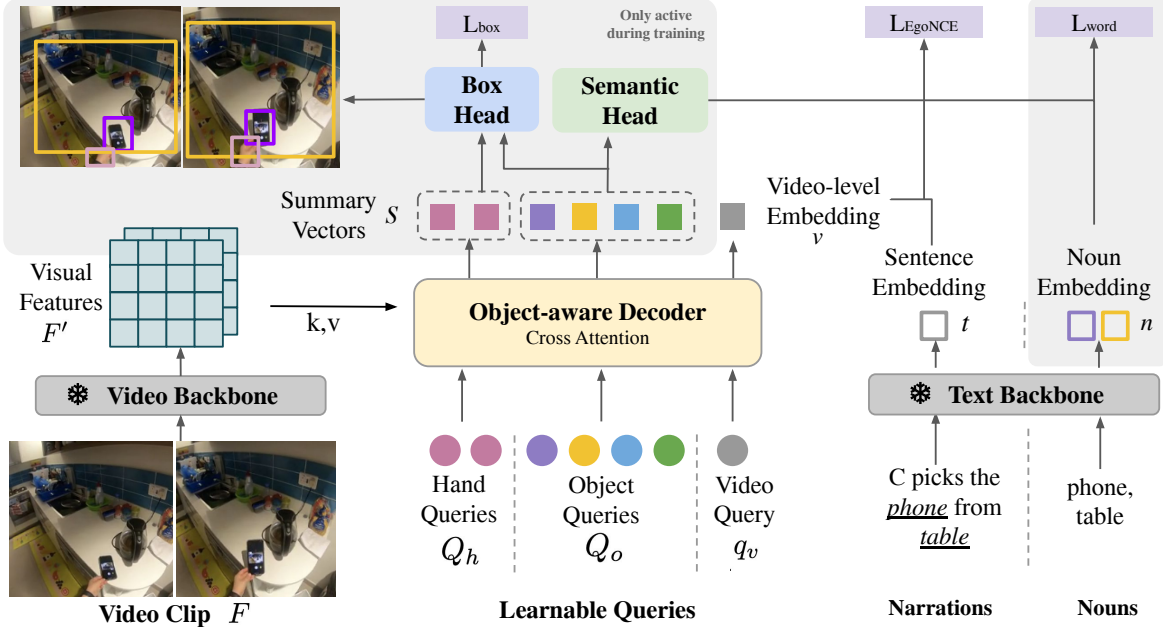


Figure 1. **The object-aware vision-language model architecture.** The architecture is made up of three parts: A video backbone, a text backbone and a object-aware decoder. The decoder is a cross-attention transformer, it takes the visual feature map  $F'$  as keys and values, which are attended by a set of learnable queries. In these permutation-invariant query vectors, the hand and object queries  $Q_h, Q_o$  are trained to be *object-aware* and predict the localization and class of hands and objects. The video query  $q_v$  attends to both the visual feature map through cross-attention layers, and the object feature map through self-attention layers in the Transformer decoder and output a video-level embedding  $v$  to be matched with the text embedding  $t$ .

sist vision-language alignment in egocentric videos, so that the model learns grounded video representations that can generalize better. SViT [3] uses object queries shared between images and videos in order to predict hand-object bounding boxes in videos, whilst only requiring image-level supervision during training. However, the object queries are not used for vision-language alignment. Our previous work [71] showed that encoding object-level information in the model helps transfer learning in various video understanding tasks, but the model required GT bounding boxes as input during inference. In contrast, in this work we model does not require this information during training, and we show box prediction and vision-language alignment can be combined and benefit both in-domain and out-of-domain tasks.

### 3. Object-Aware Vision-Language Learning

We first describe the task of object-aware vision-language learning and the architecture of our model. These are followed by the training objectives for vision-language matching and weakly-supervised text grounding.

#### 3.1. Overview

In vision-language representation learning the training data consists of short video clips (RGB frames) and an

associated free-form text string containing words that describe the visual content. Typically, dual encoders are trained on this paired data, with a visual-encoder that ingests the video clip and a language-encoder that ingests the text [4, 35, 41, 49, 64]. The dual encoders are trained with a contrastive objective [46] such that the cosine similarity for matching vision-text pairs is optimized to be higher than the similarity of negative/not-matching pairs. This pre-training objective enables evaluation on downstream vision-language tasks like video-text retrieval and action classification in a zero-shot manner [49].

Our object-aware model follows the data, training and evaluation pipeline as above, except that the model is also tasked to output object-level information (e.g., bounding boxes and object categories) *during training*. By tasking the model to predict object bounding boxes and names which can be matched to the nouns in the narration, the model learns grounded and fine-grained correspondence between the modalities. The object-level prediction is used as an auxiliary objective in training but not used at inference time.

In more detail, as shown in figure 2, there are two types of object-level prediction: (a) hand and object bounding boxes; and (b) object names. Since ground truth of boxes and object names are not available, and most traditional detectors [7, 81] fail to identify objects well in ego-centric

videos, we cannot rely on strong supervision for the predictions. Instead, we generate bounding box targets (for the hands and other objects) using a robust off-the-shelf hand-object detector [53], though these targets will only be available for some of the frames and are noisy. While for object name prediction, we use a weakly supervised method to align the predicted names with nouns in the paired narration (detailed in section 3.3). In both cases, the supervision is *opportunistic* and only applied when available.

### 3.2. Architecture

**Dual Encoder.** We use dual encoders as our visual and text backbone for efficiency. The visual encoder ingests a video clip  $F$  of RGB frames  $F = (f_1, f_2, \dots, f_T)$ , where  $f_i \in \mathbb{R}^{H \times W \times 3}$  and  $T$  is the number of frames. The clip  $F$  is encoded by a Video Transformer [6] which tokenizes the frames by 3D patches to produce a spatially downsampled feature map  $F' = (f'_1, f'_2, \dots, f'_T)$ , where  $f'_i \in \mathbb{R}^{H' \times W' \times C}$ . It outputs a visual embedding  $v \in \mathbb{R}^C$ .

The text encoder is a Transformer [61] that inputs words tokenized by a BPE tokenizer [15]. It encodes two type of inputs: (a) a narration sentence which describe the contents of a clip; and (b) a noun set which contains noun phrases from the narration sentence. At the output, the embedding corresponding to the EOS token is taken to be the embedding for the full sentence  $t \in \mathbb{R}^C$ , and multiple noun embeddings  $n \in \mathbb{R}^C$ .

**Object-Aware Module.** The object-aware module is a cross-attention Transformer which has a permutation-invariant set of learnable vectors as queries (similar to DETR [7] and Mask2former [9]). The queries are at video-level, shared between frames. They consist of three sets: two hand queries  $Q_h = (q_{h1}, q_{h2})$  for the left and right hands;  $K$  object queries  $Q_o = (q_{o1}, q_{o2} \dots, q_{ok})$ ; and a video-level query  $q_v$ . These queries are learned, and attend to the visual feature map  $F'$  from the visual backbone and output a set of summary vectors  $S = (s_{h1}, s_{h2}; s_{o1}, \dots, s_{ok})$  corresponding to each input query.

The object-aware module operates on the visual content without any interaction with the text information. It consists of six cross-attention blocks. As in a traditional Transformer decoder [61], in each block, there is a multi-head self-attention layer and a multi-head cross-attention layer. The self-attention layer enables interactions between hand, object and video queries, and the cross-attention layer allows the query to extract object-oriented information from the visual content.

**Bounding Box Head.** The hand query vectors  $Q_h$  and object query vectors  $Q_o$  are trained to predict the bounding box of the hands and objects respectively in each frame.

Note these query vectors and summary vectors are at the video level; to predict boxes at frame level, we condition a summary vector  $s_j$  of object  $j$  on a learnable frame index vector  $x_i$  by concatenation of  $s_j$  and  $x_i$ , and use a multi-layer perceptron  $F_{box}$  to project them onto a bounding box  $\hat{b}_{j,1}$ , where  $i$  is the frame number:

$$\hat{b}_{ji} = F_{box}(s_j; x_i) \quad (1)$$

As a result, we will have a time series of bounding boxes  $(\hat{b}_{j,1}, \hat{b}_{j,2}, \dots, \hat{b}_{j,T})$  from each  $j^{\text{th}}$  summary vector.

**Semantic Head.** We assign semantic meanings to object summary vectors, standing for the object name/class. To achieve this, we project  $s_j$  onto a word embedding  $\hat{n}$  with a multi-layer perceptron  $F_{semantic}$ :

$$\hat{n} = F_{semantic}(s_j) \quad (2)$$

### 3.3. Training Objectives

**Vision-Text Matching.** We follow EgoVLP [35] and use EgoNCE loss as the objective for matching between video-level embedding  $v$  and sentence-level embedding  $t$  of the narration. In one batch  $\mathcal{B}$ , the positive sample set  $\mathcal{P}_m$  is made up of a sample  $i$  and other samples that share at least one noun and one verb with it:  $\mathcal{P}_m = \{n \in \mathcal{B} \mid \text{noun}(n) \cap \text{noun}(m) \neq \emptyset, \text{verb}(n) \cap \text{verb}(m) \neq \emptyset\}$ . And for each sample  $i$ , there is a hard negative sample  $i'$  sampled from the same video. Hence, the samples in the original batch  $\mathcal{B}$  and their hard negative counterparts together form the new batch  $\tilde{\mathcal{B}}$ .

The objective matching video-to-text (v2t) for a video embedding  $v$  is formulated as below; in practice the symmetric text-to-video (t2v) matching objective is also used (omitted for brevity):

$$\mathcal{L}_{v2t}^{\text{ego}} = \frac{1}{|\mathcal{P}_m|} \sum_{k \in \mathcal{P}_m} \log \frac{\exp(v^T t_k / \tau)}{\sum_{n \in \tilde{\mathcal{B}}} (\exp(v^T t_n / \tau) + \exp(v^T t_{n'} / \tau))}. \quad (3)$$

**Bounding Box Prediction.** We use the 100DOH off-the-shelf hand object detector [53] to produce bounding boxes of two classes on each frame as supervision: hand and objects that are in contact with hands.

There are two challenges in using the detections for training supervision: 1) the image detector acts at the image level independently and does not provide box-ID association over different frames in a clip; 2) many hands and objects are missed due to motion blur and domain gap in ego-centric videos. Therefore, we apply Hungarian matching between predicted boxes  $\hat{b}$  and ground-truth boxes  $b_i$  on single frames independently, so that for each  $b_i$ , we find the matched prediction  $\hat{b}_{\sigma(i)}$  to minimize the global matching

cost. The final loss on bounding boxes is computed as the sum of the  $\ell_1$  loss and the Generalized IoU loss  $L_{iou}$  [51] on paired boxes:

$$\mathcal{L}_{\text{box}}(b_i, \hat{b}_{\sigma(i)}) = \mathcal{L}_{\text{iou}}(b_i, \hat{b}_{\sigma(i)}) + \|b_i - \hat{b}_{\sigma(i)}\|_1 \quad (4)$$

and, to tackle the problem of missing objects, we do not penalize boxes that are not matched to nouns unlike traditional detection tasks.

**Object Class Prediction.** We have noun embeddings from the video description and a set of predicted object name embeddings from the summary vectors, the task is to find the correspondence between them so that we can use the ground-truth nouns from the description as supervision for the predicted object names. As shown in figure 2, we align the nouns in the narrations and the names of object-boxes in two steps:

(1) **Object-noun alignment:** We score the predicted object name embeddings  $\hat{n}$  against the ground-truth noun embeddings  $n$  to construct a similarity matrix  $C \in \mathbb{R}^{K \times N}$ , where  $K$  is the number of object queries and  $N$  is the number of noun phrases in the description as following:

$$C(n, \hat{n}) = \frac{n \cdot \hat{n}}{\|n\| \|\hat{n}\|} \quad (5)$$

Cost matrix  $-C$  is used in Hungarian matching to select the matched summary vector for each noun phrase.

(2) **Word-level contrastive training:** We apply InfoNCE loss on the matched embeddings  $\hat{n}_j$  and  $n_j$  against the embeddings of all the nouns  $n'_k$  in Ego4D taxonomy dictionary  $\mathcal{D}$  [16]:

$$\mathcal{L}_{\text{word}} = -\frac{1}{N} \sum_{j=1}^N \log \frac{\exp(\hat{n}_j^T n_j / \tau)}{\sum_{k \in \mathcal{D}} \exp(\hat{n}_j^T n'_k / \tau)} \quad (6)$$

**Training Loss.** The total training objective is the sum of the vision-text matching loss and the auxiliary losses on object vectors:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{v2t}}^{\text{ego}} + \mathcal{L}_{\text{t2v}}^{\text{ego}} + \mathcal{L}_{\text{box}} + \lambda_{\text{word}} \mathcal{L}_{\text{word}} \quad (7)$$

### 3.4. Inference

Once trained, the model acts as a standard ego-centric vision-language video model which operates just on video frames and text descriptions, without requiring further access to object boxes or detectors. However, if desired, hand and object box detections and names can be read out for each frame at inference using the summary vectors, which can be used for grounding the input text description.

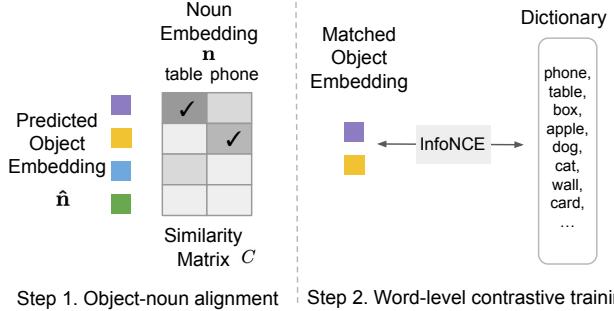


Figure 2. **Training the model to predict object classes.** **Left: Object-noun alignment.** First, the nouns in the video descriptions are matched against the predicted classes using Hungarian matching, to choose the most matched summary vectors. **Right: Word-level contrastive training.** Next, we supervise the matched predicted classes using a contrastive objective [46] against all the nouns in Ego4D taxonomy, to have similar embeddings as the corresponding nouns.

## 4. Implementation

In section 4.1, details of extracting the hand and object detections from 100DOH pre-trained detector are summarized, followed by the architectural details in section 4.2. Finally, in section 4.3 the training pipeline, model and input specification, and optimization details are provided. More details can be found in the supplementary materials.

### 4.1. Weak Supervision from Pre-trained Detector

We uniformly sample four frame from each clip in the EgoClip dataset [35] as the input to the 100DOH hand-object detector [53]. The short side of the frame is resized to 640 pixels. There are 16 million frames in total, and the average number of boxes detected per frame is 1.96 for hand and 1.67 for object. Among all the frames, about 15.8% frames have no hands detected and 17.9% frames have no object detected. The average size of hand boxes is 2.8% of the frame size, while the average size of object boxes is 19.4% of the frame size. We use the top 2 hands and top 4 objects detected in the scene as supervision.

### 4.2. Architecture

The object-aware module is a 6-layer cross-attention transformer with 8 attention-heads in each layer. The hidden dimension in cross-attention layers is 768, and the video embedding, object embeddings and text embeddings are projected to 256 dimensions before computing the cosine similarity score. We set the number of hand queries to 2, number of object queries to 12, which is designed to be larger than the maximum number of objects in the supervision. We use TimeSformer-L (TSF-L) [6] as the visual encoder, and a 12-layer Transformer [61] from LaViLa [77] as the text encoder.

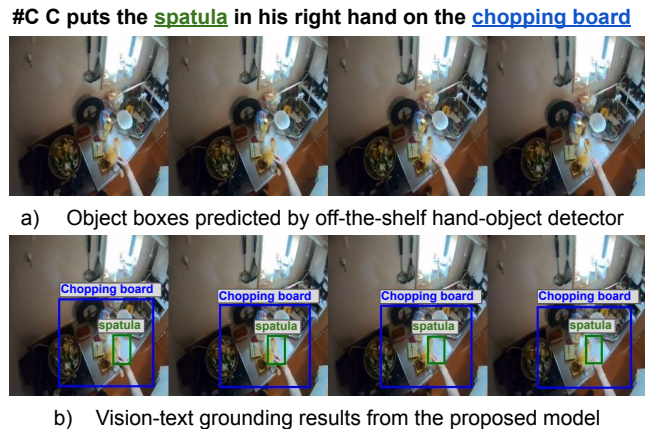
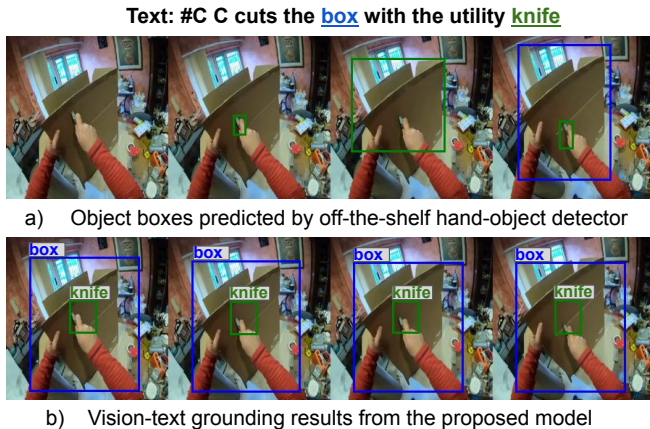


Figure 3. **Visualization of text grounding on EgoClip.** We show the comparison between detections from off-the-shelf 100DOH hand-object detector [53] (used for training supervision) and the predicted boxes from our model respectively. (a) The detections are noisy, objects are missed, and there is no temporal association of the detected boxes across frames. (b) The trained model learns temporally consistent tracks as well as object categorization using only noisy frame-level box-supervision and weak supervision from the texts.

### 4.3. Training Details

In training, we uniformly sample  $4 \times 224 \times 224$  RGB frames from each clip. Hand and object boxes are pre-extracted from these 4 frames using off-the-shelf detector and used as supervision. We keep the visual and text encoder frozen in training. Only the object-aware decoder, query vectors, and the MLP projection layer on text embedding parameters are learned during training. We train the model for 5 epochs on one A6000 GPU, with batch size 128. We use AdamW [39] as the optimizer and set the learning rate to  $3e-5$ .  $\lambda_{word}$  is set to 0.5 to balance the scale of the four losses.

## 5. Experiments

Section 5.1 introduces all the datasets we used for training and evaluation, followed by the evaluation protocols in section 5.2. Finally, we discuss the ablation studies (section 5.3) and compare to prior SOTA methods on different benchmarks (section 5.4).

### 5.1. Datasets

**Ego4D/EgoNLQ/EgoMQ/EgoClip/EgoMCQ** [16, 35]. Ego4D is a massive-scale dataset focusing on ego-centric videos. It contains 3670 hours video for many different tasks, including action anticipation, AV diarization, etc. EgoNLQ and EgoMQ is a subset for natural language queries and moment query, designed for testing the models' episodic memory and long-term video understanding. [35] proposes a new subset EgoClip for vision-language pre-training, comprising 3.8M clip-text pairs. They also introduce EgoMCQ (*i.e.*, Egocentric Multiple-Choices-Question) as a downstream evaluation dataset for the pre-training. Given a text query, the model tasked to

choose the paired video clip from 5 candidates. The evaluation metrics is 'intra-video' and 'inter-video' accuracy, depending on where the candidates are chosen from.

**Epic-Kitchens-MIR** [11]. Epic-Kitchens is a large-scale ego-centric dataset with 100-hour activities recorded in kitchens. Epic-Kitchens-MIR is a subset with about 9881 clip-text pairs for vision-language retrieval. It is designed for multi-instance retrieval. The model is evaluated on retrieving the paired text/video given a query text/video. The evaluation metrics are mean average Precision (mAP) and normalized Discounted Cumulative Gain (nDCG).

**VISOR** [12] is a benchmark built on Epic-Kitchens for segmenting hands and active objects in egocentric video. It has pixel-level annotations covering 36 hours of untrimmed video and 257 object classes. We utilize this annotations in its val split for evaluations on vision-text grounding.

**EGTEA** [34] contains 28 hours of cooking activities from 86 unique sessions of 32 subjects. Fine-grained actions are classified into 106 classes. We retrieve the text-descriptions of action classes to evaluate the model for action classification on the test set of all three splits. We measure the performance using mean-class accuracy and top1 accuracy.

### 5.2. Evaluation Protocol

We evaluate the performance of our model in the following three aspects:

**Zero-shot transfer.** To test the transferability and generalization ability, we conduct zero-shot evaluation on multiple-choice questions (EgoMCQ), multi-instance retrieval (Epic-Kitchens-MIR), action classification (EGTEA). Among these datasets, videos in EgoMCQ are from the same data

source as our pre-training dataset Ego4D. Other datasets demonstrate a domain gap, hence, evaluate for transferable representations.

**Episodic memory.** To evaluate the richness of the representations learned by our model, we use the video representations to solve Episodic memory tasks in Ego4D. Following [35, 77], we pre-extract the video features from our model first. Using these pre-computed features as input, we train a VSLNet [72] for temporal localization in NLQ, and train a VSGN [76] for moment retrieval in MQ.

**Vision-language grounding.** Due to the lack of ground-truth for object-grounding in EgoClip, we evaluate the grounding ability of the model on VISOR instead. VISOR is an egocentric dataset with scenes in the kitchens, where frames are annotated sparsely with segmentation masks and object names. We re-propose the manually annotated segmentation masks in it to extract ground-truth bounding-boxes for hands and in-contact objects. To carry out the zero-shot evaluation, we take the annotated frames in the val split, filter out the not-in-contact objects in each frame, and convert all the segmentation masks to bounding boxes as ground truth. The predicted object boxes are matched with ground-truth object boxes using noun alignment as during training (eq. (5)), while the left/right hands are predicted from the first and the second hand queries respectively. We repeat a single frame 4 times temporally, and resize it to  $224 \times 224$  pixels to be consistent with the pre-training resolution. The predicted bounding-boxes are evaluated to be correct if their centers lie inside the ground-truth bounding-boxes.

### 5.3. Ablations

**Losses.** We ablate the combination of losses on three downstream benchmarks in table 1. Results showing that having both  $L_{box}$  and  $L_{word}$  leads to the best performance. With the same architecture, when training the model using only  $L_{Ego}$  without introducing any object-awareness, the performance is 2% lower on EK100-MIR and EGTEA compared to the object-aware one.

| Losses                         | EgoMCQ      |             | EK100-MIR   |             | EGTEA       |             |
|--------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|
|                                | Inter       | Intra       | Avg mAP     | Avg nDCG    | Top1        | Mean        |
| $L_{Ego}$                      | 93.7        | 61.8        | 35.9        | 36.6        | 44.9        | 37.6        |
| $L_{Ego} + L_{box}$            | 94.2        | 62.7        | 36.7        | 37.4        | 45.3        | 38.5        |
| $L_{Ego} + L_{word}$           | 93.7        | 61.9        | 36.7        | 37.4        | 45.8        | 38.1        |
| $L_{Ego} + L_{box} + L_{word}$ | <b>94.5</b> | <b>63.0</b> | <b>37.5</b> | <b>37.8</b> | <b>46.6</b> | <b>39.1</b> |

Table 1. **Ablation on training objectives for zero-shot transfer tasks.** Introducing the object-awareness by having box and word supervision helps the model to achieve better transfer results on EK100-MIR and EGTEA.

| Detector input res | EgoMCQ      |             | EK          |             | EGTEA       |             | VISOR       |
|--------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
|                    | Inter       | Intra       | Avg mAP     | Avg nDCG    | Top1        | Mean        | Loc Acc     |
| 256p               | 94.2        | <b>63.2</b> | 35.7        | 34.6        | 42.0        | 36.0        | -           |
| 256p               | <b>94.5</b> | 63.0        | 36.9        | 37.0        | 44.3        | 38.9        | 68.2        |
| 640p               | <b>94.5</b> | 63.0        | <b>37.5</b> | <b>37.8</b> | <b>39.1</b> | <b>46.6</b> | <b>78.7</b> |

Table 2. **Ablation on box quality for zero-shot transfer tasks.** We extract boxes using 256p and 640p images as input to the detector respectively, resulting in boxes of different qualities as supervision in training.

| # Obj Queries | EgoMCQ      |             | EK100-MIR   |             | EGTEA       |             |
|---------------|-------------|-------------|-------------|-------------|-------------|-------------|
|               | Inter       | Intra       | Avg mAP     | Avg nDCG    | Top1        | Mean        |
| 4             | 94.1        | 62.8        | 37.8        | 37.6        | 45.9        | 38.1        |
| 8             | <b>94.5</b> | 62.7        | 37.7        | <b>38.0</b> | 45.5        | 37.9        |
| 12            | <b>94.5</b> | <b>63.0</b> | <b>37.5</b> | <b>37.8</b> | <b>46.6</b> | <b>39.1</b> |

Table 3. **Ablation on the number of object queries for zero-shot transfer tasks.** The number of queries do not have a big impact on EK100. Larger number of queries shows a boost on mean-class accuracy on EGTEA, and smaller number of queries is better on intra-video accuracy.

**Quality of detected boxes.** The extent to which a model can acquire object-level information is constrained by the quality of the bounding boxes from the off-the-shelf detector. To investigate how much the quality of boxes affects our training, we detect hand and object boxes on EgoClip training set using 100DOH [53] with 256p and 640p images as input – with larger image size, the objects should be more precisely delineated. We use the two sets of boxes as supervision in our training and show results in table 2. Even when training with noisy boxes from 256p, our model is better than the previous SOTA model. When boxes from 640p are used, the averaged zero-shot transfer performance is further improved by 1% on Epic-Kitchens and EGTEA, showing that our method can bring larger improvement over the non-object-aware method when given better boxes. Furthermore, a significant boost on the grounding results is also observed on VISOR when using boxes with better quality.

**Number of object queries.** We use different number of query vectors in the object-aware decoder to see its impact on both vision-language tasks and the grounding task. We show zero-shot transfer results in table 3 and vision-text grounding results in table 7. The number of queries has relatively small impact on vision-language representation learning, the performance gaps on zero-shot transfer tasks are mostly small than 1%. However, a smaller number of object queries leads to much better results for in-contact object grounding on VISOR, 4 queries is better than 12 objects by 3.6% in localization accuracy. The reason is that too

| Method  | Backbone | Batch size | Object aware | Hard neg | EgoMCQ      |             | EK100-MIR   |             |             |             |             |             | EGTEA       |             |
|---------|----------|------------|--------------|----------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
|         |          |            |              |          | Inter-video | Intra-video | mAP         |             |             | nDCG        |             |             | Top1-Acc    | Mean-Acc    |
|         |          |            |              |          |             |             | V-T         | T-V         | Avg         | V-T         | T-V         | Avg         |             |             |
| EgoVLP  | TSF-B    | 512        | ✗            | ✓        | 90.6        | 57.2        | 26.0        | 20.6        | 23.3        | 28.8        | 27.0        | 27.9        | 17.6        | -           |
| LaViLa  | TSF-B    | 1024       | ✗            | ✗        | 93.8        | 59.9        | 35.1        | 26.6        | 30.9        | 33.7        | 30.4        | 32.0        | -           | 28.9        |
| LaViLa  | TSF-L    | 1024       | ✗            | ✗        | <b>94.5</b> | 63.1        | 40.0        | 32.2        | 36.1        | 36.1        | 33.2        | 34.6        | 40.1        | 34.1        |
| LaViLa* | TSF-L    | 1024       | ✗            | ✗        | 94.2        | <b>63.2</b> | 39.7        | 31.7        | 35.7        | 36.1        | 33.2        | 34.6        | 42.0        | 36.0        |
| Ours*†  | TSF-L    | 128        | ✗            | ✗        | 93.7        | 60.5        | 39.7        | 30.3        | 35.0        | 37.3        | 34.5        | 35.9        | 44.8        | 36.3        |
| Ours*‡  | TSF-L    | 128        | ✗            | ✓        | 93.7        | 61.8        | 40.7        | 31.1        | 35.9        | 38.3        | 35.0        | 36.6        | 44.9        | 37.6        |
| Ours*   | TSF-L    | 128        | ✓            | ✓        | <b>94.5</b> | 63.0        | <b>42.3</b> | <b>32.7</b> | <b>37.5</b> | <b>39.3</b> | <b>36.2</b> | <b>37.8</b> | <b>46.6</b> | <b>39.1</b> |

Table 4. **Comparison to SOTA results on zero-shot transfer to EgoMCQ, EK100-MIR and EGTEA.** We compared to EgoVLP and LaViLa, two previous SOTA models pre-trained on EgoClip. Our object-aware model has achieved comparable results on multiple-choice questions on EgoMCQ, and SOTA results on multi-instance retrieval task on EpicKitchens and action classification on EGTEA. Model without \* use center cropping in evaluation, while \* denotes the usage of resizing instead of cropping. Ours† and Ours‡ stands for different variants of our model depending on whether object-aware losses and hard negative sampling is used in training.

| Method | Predicted Boxes | EgoMCQ      |             | EK          |             | EGTEA       |             |
|--------|-----------------|-------------|-------------|-------------|-------------|-------------|-------------|
|        |                 | Inter       | Intra       | Avg mAP     | Avg nDCG    | Mean        | Top1        |
| Ours   | obj             | 94.0        | 62.1        | 36.8        | 37.1        | 45.1        | 37.6        |
|        | hand+obj        | <b>94.5</b> | <b>63.0</b> | <b>37.5</b> | <b>37.8</b> | <b>46.6</b> | <b>39.1</b> |

Table 5. **The impact of having hand boxes in the box prediction.** Having hand boxes in the prediction helps on all the zero-shot evaluation benchmarks.

many queries result in a large number of predicted boxes, which increases the probability of mis-matching.

**Impact of hand boxes.** We ablate the impact of having hand boxes as supervision in our training. Results are shown in table 5, training the model to predict hand as well as objects help the model to get about 1% higher zero-shot transfer performance on EK and EGTEA.

#### 5.4. Comparison to the SOTA

**Zero-shot transfer.** We compare to previous SOTA in table 4. Our model is comparable on EgoMCQ and better on EK-MIR and EGTEA, showing its good zero-shot transferability. Due to limited compute resources, we are not able to unfreeze the visual backbone to train end-to-end or increase the batch size further. Despite these disadvantages, our method outperforms the previous SOTA on two tasks for models that have been trained end-to-end.

The main difference between LaViLa(L) and ours is the object-aware training and hard negative sampling; Ours† in table 4 is a LaViLa(L) model with an extra Transformer decoder, which is trained with only InfoNCE loss on video and sentence embeddings. Without object-awareness and hard sampling, it gets better accuracy on EGTEA and better nDCG on EK100 due to more parameters added, but falls behind on EgoMCQ and mAP on EK100. Applying hard negative sampling (Ours‡) and object-aware training (Ours) brings improvement across the board. The most obvious boost comes from inducing object-awareness, bringing 1.5% improvement on average. And the results could be further improved by obtaining better pseudo-boxes, as the

magnitude of boost from learning objects is closely related to the box quality (as shown in table 2).

**Episodic memory.** Results on EgoNLQ and EgoMQ are shown in table 6. The two tasks test the video understanding on several-minutes long videos. In these experiments, trained video and text backbones are used as feature extractor, and extra modules are trained on the long feature sequences for natural language querying and memory querying. Therefore, the richer the information encoded in the features, the better the results will be. We list the models trained on the same amount of video and text data in black, results show that our model are better than the previous SOTA on all the metrics in the two tasks. This is because features from the object-aware model have captured more object information, thus enabling better precision and recall on localization and retrieval. InternVideo [63] and NaQ [50] are trained on more video or text data, and are included for completeness. Please note that our method is compatible with any visual language backbones, using better backbones might also lead to better results.

| Method           | Batch size | EgoNLQ      |             |            |             | EgoMQ       |             |             |
|------------------|------------|-------------|-------------|------------|-------------|-------------|-------------|-------------|
|                  |            | mIOU@0.3    |             | mIOU@0.5   |             | R1@0.5      | R5@0.5      | mAP         |
|                  |            | R1          | R5          | R1         | R5          |             |             |             |
| SlowFast         | -          | 5.5         | 10.7        | 3.1        | 6.6         | 25.2        | 46.2        | 6.0         |
| EgoVLP           | 512        | 10.8        | 18.8        | 6.8        | 13.5        | 30.1        | 52.0        | 11.4        |
| LaViLa(B)        | 1024       | 10.5        | 19.1        | 6.7        | 13.6        | -           | -           | -           |
| LaViLa(L)        | 1024       | 12.1        | 22.4        | 7.3        | 15.4        | 32.5        | 56.1        | 13.4        |
| Ours             | 128        | <b>13.2</b> | <b>23.3</b> | <b>7.9</b> | <b>15.6</b> | <b>33.4</b> | <b>56.7</b> | <b>16.0</b> |
| VideoIntern [63] | 14k        | 16.5        | 23.0        | 10.1       | 16.1        | -           | -           | 23.6        |
| ReLER + NaQ [50] | 2048       | 19.3        | 23.6        | 11.6       | 15.5        | -           | -           | -           |

Table 6. **Comparison to SOTA results of fine-tuning on EgoNLQ and EgoMQ.** Our object-aware model encodes richer information in the visual representations, hence obtaining better results on all the metrics in NLQ and MQ task in Ego4D episodic memory benchmark. We list other SOTA models (in grey) trained with more video and text data for completeness.



| Model         | Object Assignment | # queries | Loc Accuracy |
|---------------|-------------------|-----------|--------------|
| Detector [53] | Random            | -         | 37.1         |
|               | GT matching       | -         | 41.3         |
| Ours          | Predicted         | 4         | <b>82.3</b>  |
|               |                   | 8         | 81.2         |
|               |                   | 12        | 78.7         |

Table 7. **In-contact object localization accuracy on VISOR.** Our model does better on in-contact object localization after weakly-supervised training compared to the baseline (the Detector with Random or GT object assignment), which is the source of supervision in our pre-training. The improvement is due to increased recall of our model over the baseline.

### 5.5. Evaluating Object Grounding

**Qualitative results on EgoClip.** In fig. 3 we show the grounding results on EgoClip after training as compared to the supervision from the 100DOH detector. After training, the predictions from our model find the missing objects because we do not penalize extra bounding boxes predicted, but select the active object/object of interest through matching noun embeddings. It also learns temporal association of object bounding-boxes as the result of using the same object summary vector to predict the boxes over all the frames (as in eq. (1)); the summary vector attends to visually similar features corresponding to the same object across the clip without any explicit supervision for temporal consistency.

**Quantitative and qualitative results on VISOR.** In table 7 we show the text-grounding results on VISOR. We take the predictions from the hand-object detector [53] as our baseline. Since the predictions only detect hands and objects without an object class name, we associate the predicted boxes with ground-truth boxes in VISOR in two ways: (a) **random**: we assign predicted object boxes to GT object boxes randomly, (b) **GT matching**: we use Hungarian matching to find the predicted boxes with highest IoU against the GT object boxes. However, even when matched using GT information, the baseline detector does not achieve a high accuracy due to poor recall. Results show grounding ability of our model is 40% better than the baseline using only weak supervision from the video descriptions. We also show the qualitative results on detecting different number of hands and objects in fig. 4, our model has a much higher recall when compared to baseline detector when operating at the same resolution.

## 6. Conclusion

In this paper, we introduce a method to learn object-aware ego-centric video representations using noisy supervision from pre-trained hand-object detectors. The object-representations so learned show strong zero-shot transfer across various downstream tasks and datasets, mirroring

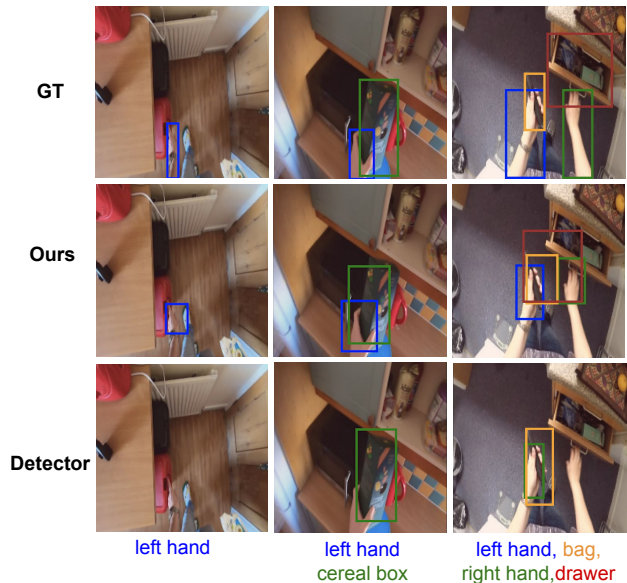


Figure 4. **Grounding visualization on the Epic-Kitchens-VISOR val split.** When operating at the same resolution, our model shows better grounding ability on hands and objects compared to the baseline 100DOH detector [53] (with GT matching) used for training. Note that the low IoU on **hands** on the third column is a result of the GT ‘hand’ segmentation mask covering the arm by definition, while the detector and our model are trained to localize only palm and fingers.

the performance improvement from object-aware training on images [70]. The model uses standard neural modules (i.e., transformers), and does not require any object boxes or detectors as input during inference, making it widely applicable as a drop-in replacement for training video-language models. Even though the model is trained with sparse and noisy object supervision at the frame-level (without temporal associations), during inference dense temporal bounding-box tracks and category predictions can be obtained, which are superior to the predictions from the base hand-object detector used for training. There are several avenues for improvement. Our model uses the pre-trained video encoder operates at a small resolution  $224 \times 224$  which makes detecting small objects difficult. Further, four frames are sampled uniformly from the clip regardless of its length which can cause difficulties due to temporal aliasing. Nevertheless, we hope our work will inspire further research in learning transferable object-aware representations for videos.

**Acknowledgements.** This research is funded by a Google-DeepMind Graduate Scholarship, a Royal Society Research Professorship, and EPSRC Programme Grant VisualAI EP/T028572/1.

## References

- [1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022. 2
- [2] Anurag Arnab, Chen Sun, and Cordelia Schmid. Unified graph structured models for video understanding. In *Proc. ICCV*, 2021. 1
- [3] Elad Ben Avraham, Roei Herzig, Karttikeya Mangalam, Amir Bar, Anna Rohrbach, Leonid Karlinsky, Trevor Darrell, and Amir Globerson. Bringing image scene structure to video via frame-clip consistency of object tokens. In *NeurIPS*, 2022. 2, 3
- [4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proc. ICCV*, 2021. 1, 2, 3
- [5] Sven Bambach, Stefan Lee, David J Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *Proceedings of the IEEE international conference on computer vision*, pages 1949–1957, 2015. 2
- [6] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding. In *Proc. ICML*, 2021. 4, 5
- [7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proc. ECCV*, 2020. 1, 3, 4
- [8] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *Proc. ECCV*, 2020. 2
- [9] Bowen Cheng, Anwesa Choudhuri, Ishan Misra, Alexander Kirillov, Rohit Girdhar, and Alexander G Schwing. Mask2former for video instance segmentation. *arXiv preprint arXiv:2112.10764*, 2021. 1, 4
- [10] Noam Chomsky and David W Lightfoot. *Syntactic structures*. Walter de Gruyter, 1957. 1
- [11] 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 *Proc. ECCV*, 2018. 1, 2, 6
- [12] 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. In *NeurIPS Datasets and Benchmarks Track*, 2022. 2, 6
- [13] Zi-Yi Dou, Aishwarya Kamath, Zhe Gan, Pengchuan Zhang, Jianfeng Wang, Linjie Li, Zicheng Liu, Ce Liu, Yann LeCun, Nanyun Peng, et al. Coarse-to-fine vision-language pre-training with fusion in the backbone. *arXiv preprint arXiv:2206.07643*, 2022. 2
- [14] Gamaleldin F Elsayed, Aravindh Mahendran, Sjoerd van Steenkiste, Klaus Greff, Michael C Mozer, and Thomas Kipf. Savi++: Towards end-to-end object-centric learning from real-world videos. In *NeurIPS*, 2022. 2
- [15] Philip Gage. A new algorithm for data compression. *C Users Journal*, 1994. 4
- [16] 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 *Proc. CVPR*, 2022. 1, 5, 6
- [17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proc. ICCV*, 2017. 1
- [18] Paul Henderson and Christoph H. Lampert. Unsupervised object-centric video generation and decomposition in 3D. In *NeurIPS*, 2020. 2
- [19] Roei Herzig, Elad Ben-Avraham, Karttikeya Mangalam, Amir Bar, Gal Chechik, Anna Rohrbach, Trevor Darrell, and Amir Globerson. Object-region video transformers. *arXiv preprint arXiv:2110.06915*, 2021. 1, 2
- [20] Roei Herzig, Elad Levi, Huijuan Xu, Hang Gao, Eli Brosh, Xiaolong Wang, Amir Globerson, and Trevor Darrell. Spatio-temporal action graph networks. In *Proc. ICCV*, 2019. 2
- [21] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proc. ICML*, 2021. 1, 2
- [22] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. Denscap: Fully convolutional localization networks for dense captioning. In *Proc. CVPR*, 2016. 2
- [23] Aishwarya Kamath, Mannat Singh, Yann LeCun, Ishan Misra, Gabriel Synnaeve, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proc. ICCV*, 2021. 2
- [24] Ranjay Krishna, Ines Chami, Michael Bernstein, and Li Fei-Fei. Referring relationships. In *Proc. CVPR*, 2018. 2
- [25] 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. *IJCV*, 2017. 2
- [26] Matej Kristan, Jiří Matas, Aleš Leonardis, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kämäräinen, Hyung Jin Chang, Martin Danelljan, Luka Cehovin, Alan Lukežič, et al. The ninth visual object tracking vot2021 challenge results. In *Proc. ICCV*, 2021. 2
- [27] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NeurIPS*, pages 1106–1114, 2012. 2
- [28] Abhijit Kundu, Kyle Genova, Xiaoqi Yin, Alireza Fathi, Caroline Pantofaru, Leonidas J Guibas, Andrea Tagliasacchi, Frank Dellaert, and Thomas Funkhouser. Panoptic neural fields: A semantic object-aware neural scene representation. In *Proc. CVPR*, 2022. 2
- [29] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 2017. 1
- [30] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Proc. ICML*, 2022. 2

- [31] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *NeurIPS*, 2021. 2
- [32] Shuang Li, Yilun Du, Antonio Torralba, Josef Sivic, and Bryan Russell. Weakly supervised human-object interaction detection in video via contrastive spatiotemporal regions. In *Proc. ICCV*, 2021. 2
- [33] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *Proc. ECCV*, 2020. 2
- [34] Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *Proc. ECCV*, 2018. 6
- [35] Kevin Qinghong Lin, Alex Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Zhongcong Xu, Difei Gao, Rongcheng Tu, Wenzhe Zhao, Weijie Kong, et al. Egocentric video-language pretraining. *arXiv preprint arXiv:2206.01670*, 2022. 2, 3, 4, 5, 6, 7
- [36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proc. ECCV*, 2014. 2
- [37] Miao Liu, Siyu Tang, Yin Li, and James M Rehg. Forecasting human-object interaction: joint prediction of motor attention and actions in first person video. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 704–721. Springer, 2020. 2
- [38] Shaowei Liu, Subarna Tripathi, Somdeb Majumdar, and Xiaolong Wang. Joint hand motion and interaction hotspots prediction from egocentric videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3282–3292, 2022. 2
- [39] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proc. ICLR*, 2019. 6
- [40] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019. 2
- [41] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, 2022. 2, 3
- [42] Joanna Materzynska, Tete Xiao, Roei Herzig, Huijuan Xu, Xiaolong Wang, and Trevor Darrell. Something-else: Compositional action recognition with spatial-temporal interaction networks. In *Proc. CVPR*, 2020. 1
- [43] Effrosyni Mavroudi and René Vidal. Weakly-supervised generation and grounding of visual descriptions with conditional generative models. In *Proc. CVPR*, 2022. 2
- [44] 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 *Proc. ICCV*, 2019. 2
- [45] Marvin Minsky. *Society of mind*. Simon and Schuster, 1988. 1
- [46] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 3, 5
- [47] 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. 2
- [48] Gorjan Radevski, Marie-Francine Moens, and Tinne Tuytelaars. Revisiting spatio-temporal layouts for compositional action recognition. In *Proc. BMVC*, 2021. 1
- [49] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proc. ICML*, 2021. 1, 2, 3
- [50] Santhosh Kumar Ramakrishnan, Ziad Al-Halah, and Kristen Grauman. Naq: Leveraging narrations as queries to supervise episodic memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6694–6703, 2023. 8
- [51] Hamid Rezaatofghi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proc. CVPR*, 2019. 5
- [52] Kate Saenko, Ben Packer, C Chen, Sunil Bandla, Y Lee, Yangqing Jia, J Niebles, Daphne Koller, Li Fei-Fei, Kristen Grauman, et al. Mid-level features improve recognition of interactive activities. Technical report, Dept. of Elec. Engg. and Computer Sc., University of California, Berkeley, 2012. 1
- [53] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F Fouhey. Understanding human hands in contact at internet scale. In *Proc. CVPR*, 2020. 1, 2, 4, 5, 6, 7, 9
- [54] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. ICLR*, 2015. 2
- [55] Gautam Singh, Yi-Fu Wu, and Sungjin Ahn. Simple unsupervised object-centric learning for complex and naturalistic videos. In *NeurIPS*, 2022. 2
- [56] Elizabeth S Spelke, Karen Breinlinger, Janet Macomber, and Kristen Jacobson. Origins of knowledge. *Psychological review*, 99, 1992. 1
- [57] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. In *Proc. ECCV*, 2018. 1
- [58] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*, 2019. 2
- [59] Reuben Tan, Bryan Plummer, Kate Saenko, Hailin Jin, and Bryan Russell. Look at what i’m doing: Self-supervised spatial grounding of narrations in instructional videos. *NeurIPS*, 2021. 2
- [60] Joshua B Tenenbaum, Charles Kemp, Thomas L Griffiths, and Noah D Goodman. How to grow a mind: Statistics, structure, and abstraction. *Science*, 2011. 1
- [61] 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. 1, 4, 5
- [62] Xiaolong Wang and Abhinav Gupta. Videos as space-time

- region graphs. In *Proc. ECCV*, 2018. 1, 2
- [63] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022. 8
- [64] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *arXiv preprint arXiv:2109.14084*, 2021. 1, 2, 3
- [65] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018. 2
- [66] Bangbang Yang, Yinda Zhang, Yinghao Xu, Yijin Li, Han Zhou, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. Learning object-compositional neural radiance field for editable scene rendering. In *Proc. ICCV*, 2021. 2
- [67] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *Proc. ICCV*, 2019. 2
- [68] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. *arXiv preprint arXiv:1910.01442*, 2019. 2
- [69] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 2
- [70] Yan Zeng, Xinsong Zhang, and Hang Li. Multi-grained vision language pre-training: Aligning texts with visual concepts. *arXiv preprint arXiv:2111.08276*, 2021. 2, 9
- [71] Chuhan Zhang, Ankush Gupta, and Andrew Zisserman. Is an object-centric video representation beneficial for transfer? In *Proc. ACCV*, 2022. 1, 2, 3
- [72] Hao Zhang, Aixin Sun, Wei Jing, Liangli Zhen, Joey Tianyi Zhou, and Rick Siow Mong Goh. Natural language video localization: A revisit in span-based question answering framework. *IEEE PAMI*, 2021. 7
- [73] Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Harold Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. Glipv2: Unifying localization and vision-language understanding. In *NeurIPS*, 2022. 2
- [74] Lingzhi Zhang, Shenghao Zhou, Simon Stent, and Jianbo Shi. Fine-grained egocentric hand-object segmentation: Dataset, model, and applications. In *European Conference on Computer Vision*, pages 127–145. Springer, 2022. 2
- [75] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proc. CVPR*, 2021. 2
- [76] Chen Zhao, Ali K Thabet, and Bernard Ghanem. Video self-stitching graph network for temporal action localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13658–13667, 2021. 7
- [77] Yue Zhao, Ishan Misra, Philipp Krähenbühl, and Rohit Girdhar. Learning video representations from large language models. *arXiv preprint arXiv:2212.04501*, 2022. 2, 5, 7
- [78] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *Proc. CVPR*, 2022. 2
- [79] Luowei Zhou, Nathan Louis, and Jason J Corso. Weakly-supervised video object grounding from text by loss weighting and object interaction. *arXiv preprint arXiv:1805.02834*, 2018. 2
- [80] Xingyi Zhou, Anurag Arnab, Chen Sun, and Cordelia Schmid. How can objects help action recognition? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2353–2362, 2023. 2
- [81] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *Proc. ECCV*, 2022. 3