In Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2025

Which Viewpoint Shows it Best? Language for Weakly Supervising View Selection in Multi-view Instructional Videos

Sagnik Majumder[1,2] Tushar Nagarajan[2] Ziad Al-Halah[3] Reina Pradhan[1] Kristen Grauman[1,2]
[1]UT Austin [2]FAIR, Meta [3]University of Utah

Abstract

Given a multi-view video, which viewpoint is most informative for a human observer? Existing methods rely on heuristics or expensive "best-view" supervision to answer this question, limiting their applicability. We propose a weakly supervised approach that leverages language accompanying an instructional multi-view video as a means to recover its most informative viewpoint(s). Our key hypothesis is that the more accurately an individual view can predict a view-agnostic text summary, the more informative it is. To put this into action, we propose LANGVIEW*, a framework that uses the relative accuracy of view-dependent caption predictions as a proxy for best view pseudo-labels. Then, those pseudo-labels are used to train a view selector, together with an auxiliary camera pose predictor that enhances view-sensitivity. During inference, our model takes as input only a multi-view video—no language or camera poses—and returns the best viewpoint to watch at each timestep. On two challenging datasets comprised of diverse multi-camera setups and how-to activities, our model consistently outperforms state-of-the-art baselines, both with quantitative metrics and human evaluation. Project:* https://vision.cs.utexas.edu/projects/which-view-shows-it-best.

1. Introduction

Videos are an essential vehicle for communicating how to perform a new skill, as evidenced by the millions of "how-to" videos online, for everything from frosting a cake to perfecting a basketball layup. The more intricate the task, however, the more important the *viewpoint* used to film the instructional video. For example, a close-up view of the hands is desirable when a knitter shows how to add stitches of yarn to a needle, or when a rock-climber demonstrates a particular hold—whereas a view from afar may be preferable when the knitter shows the knitted sweater being worn, or the climber shows their selected path up the wall. In general, the information available in any given viewpoint of an activity varies. Not all views are created equal.

Shooting a video with multiple cameras provides a holistic view of the activity taking place in a scene, by capturing it from different locations and angles, and *multi-view video* is developing as a new frontier in computer vision research [37, 49, 75, 96, 102], especially in instructional settings [37, 74]. However, multi-view videos are generally not suitable for direct human consumption [24]: digesting multiple views at once imposes a high cognitive burden.

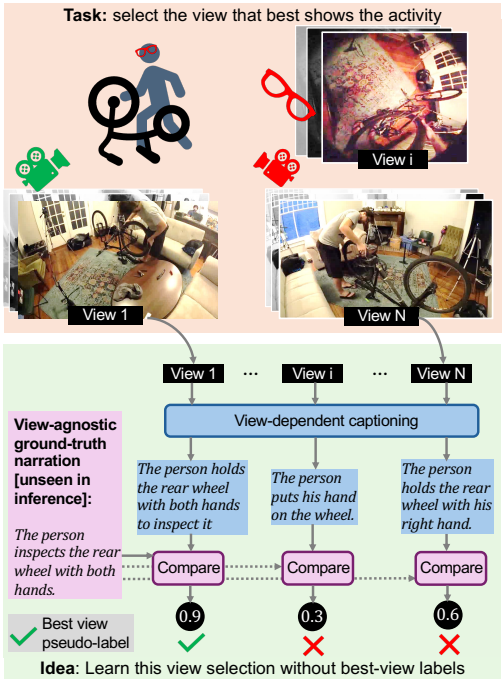


Figure 1. LANGVIEW idea: given multi-view instructional videos, we aim to learn a view selection model that can identify the best view for seeing how to perform the activity shown in the videos, in the *absence of best view labels*. To achieve this, we compare each estimated view-dependent caption to the view-agnostic ground-truth video narration of the human activity, and use their respective accuracies as a proxy for view quality. These quality scores then serve as pseudo-labels for learning to select the most informative view. In this example, the 1st view most clearly shows all entities involved in the activity—the wheel and the person's hands, and how they interact—and hence, produces a caption that best matches the ground-truth, making it a positive pseudo-label for view selection.

Thus, in practice, the status quo is to orchestrate view selection in how-to videos manually with either active camera work or post-production video editing tools, which is time consuming and tedious.

What if instead a vision model learned to automatically perform *view selection*, at every time step deciding which camera from the multi-view video to adopt? View selection has traditionally been studied in the context of automatic cinematography for specialized domains, e.g., 360° panoramas [62, 99], sports clips [13, 48], virtual environments [30, 44, 45, 76], or lecture videos [31, 101, 121]. Aside from their specialized domains, existing work is limited by relying on hand-coded heuristics [4, 30, 44] or assuming access to manual labels indicating the favored views for training [13, 18, 48, 62, 99, 112]. Such labels are expensive and quite special purpose.

Conscious of these shortcomings, we propose to learn view selection in multi-view instructional videos in the *absence of best view labels*. Towards that goal, we hypothesize that *view-agnostic* natural language descriptions of the activity shown in the videos [37, 49, 56]—commonly referred to as "narrations"[1], can act as a source of weak supervision. Specifically, our core idea is that for any multi-view video clip, the viewpoint that is most predictive of such a narration is likely to be the most informative of the activity, and hence, can be pseudo-labeled as the best view for training a view selector. For example, given a multi-view video of a person repairing a bike (Figure 1), independent captions on each view will emphasize different visible components of the scene (the wheel, the person's hands, other objects in the scene, etc.); the caption most aligned with the view-agnostic narration "*the person removes the rear wheel with both hands*" indicates which view is most informative for the *whole* activity content in that clip. Unlike explicit best-view labels, the vision-language annotations that fuel today's captioners are open-world, versatile, and widely available.

To validate our hypothesis, we design a novel framework called LANGVIEW, which is composed of two key elements: a best view pseudo-labeler, and a best view selector. The pseudo-labeler automatically generates best view pseudo-labels for a multi-view video during training, by using off-the-shelf video captioners [64, 122] to score and rank views on the basis of how well the predicted narration from a view matches the view-agnostic ground-truth narration. The selector takes a multi-view video as input, and predicts the best view labels. During training, the selector also solves an auxiliary task of predicting the relative camera pose between different views, to increase its view-sensitivity and improve its selection accuracy. At inference, our model requires as input only a multi-view video, but no language or camera poses.

We evaluate LANGVIEW using two challenging multi-view instructional video datasets encompassing diverse activity scenarios and multi-camera setups, Ego-Exo4D [37] and LEMMA [56]. On both, our model outperforms multiple baselines and state-of-the-art methods for view selection on several automatic and human evaluation metrics. More broadly, our work offers a novel way for language to elicit the information content of video.

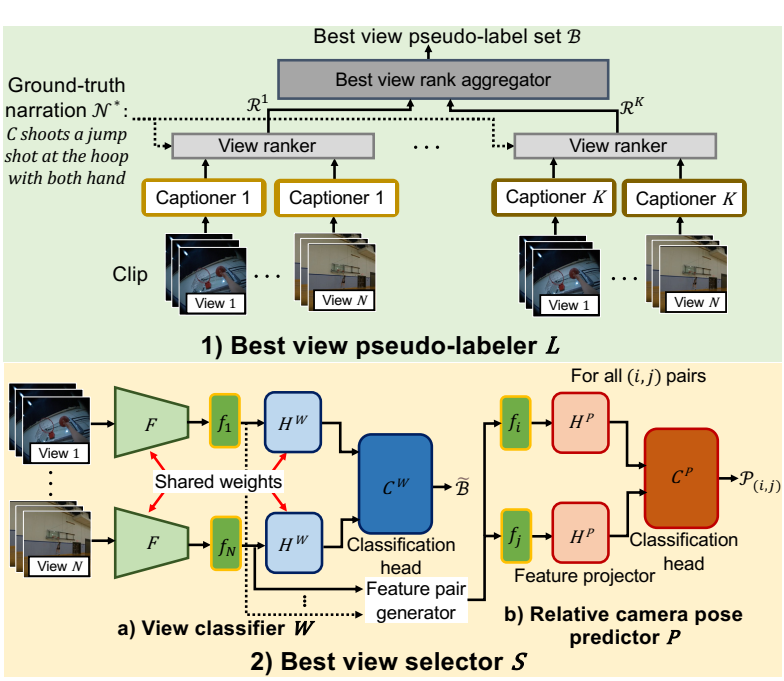
[1]Narrations in multi-view datasets [37, 49, 56] are produced by human annotators who watch all views and write down a view-independent description of how the activity is performed, and in the wild they correspond to the "how-to" descriptions spoken by a person demonstrating a task [74].

2. Related work

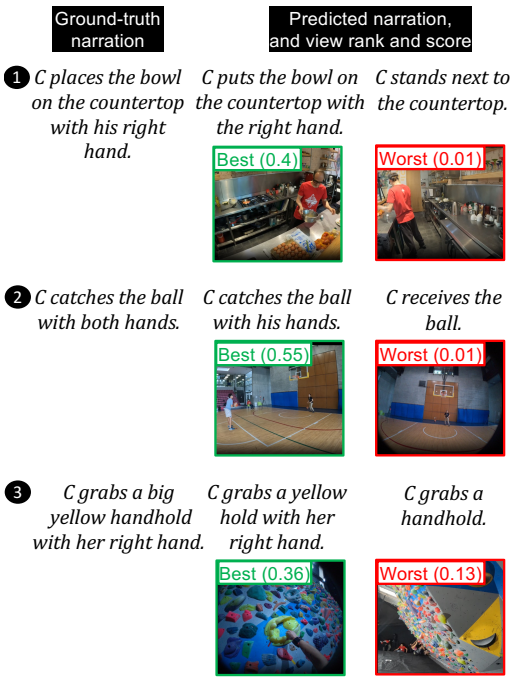
Temporal video summarization. Temporal video summarization [9, 43, 78, 81, 85, 92] creates a synopsis of a long video by identifying and stitching together its most representative clips. Early unsupervised methods use hand-crafted features, optimizing for metrics like object saliency [71] and motion attention [80], while recent efforts [35, 42, 63, 66] use labeled data [41, 97] for supervised training. To mitigate the supervision cost, recent work leverage web priors [11, 58, 59] to target settings with insufficient or unreliable annotations [58, 83, 97, 115] or unpaired data [78, 91, 118]. Different from all of the above, we tackle the problem of label scarcity in the context of *view selection* in multi-view videos, a distinct task from video summarization. In multi-video summarization, the input videos are either captured with multi-view cameras in indoor surveillance [32, 82] or street [29, 32, 123] settings, or grouped using shared visual concepts [19, 73, 81, 93, 110]. Such methods can select multiple views at the same time, and hence, are not applicable in our setting. Unlike all video summarization methods, where the goal is to create a *sparse* temporal summary of one or more videos, our task requires choosing the appropriate camera view at *each* time step (clip).

Automatic cinematography. Automatic cinematography involves automatically choosing the best camera angles, positions, and zoom levels for human consumption of a video scene. Existing work explores camera control in virtual environments [30, 44, 45, 76], or addresses a narrow domain like lecture videos [31, 33, 46, 101, 121], social settings with multiple egocentric cameras [4], or panoramic 360° input videos [18, 62, 98, 99, 112]. Prior learning-based methods require manual labels to guide selection [13, 48, 62, 98, 99]. In contrast, we show how to train a view selector *without* manual labels, by exploiting the accuracy of predicted narrations from different views as a proxy for view quality. In continued complementary research, we explore how to leverage single-view instructional videos during training [72]

Active next view selection. Active next view selection [5] requires an embodied agent to smartly control its camera



(a) Language-guided view selection training framework



(b) Ground-truth narration and pseudo-labeler output samples

Figure 2. (a) Our model uses language guidance to train a view-selector for multi-view instructional videos, such that the chosen views help best understand the shown activity. To do so, we first generate best view pseudo-labels during training by leveraging clip narrations, where each narration is a *view-agnostic* and detailed description of the activity. Specifically, given a training clip, we use off-the-shelf video captioners to predict a caption per view, score the views by comparing their captions to the ground-truth narration, and finally rank the views to generate a best view pseudo-label for the clip. Given the multi-view clip, our view classifier (bottom-left) encodes it with a visual encoder, and predicts a pseudo-label estimate. We also solve an auxiliary task of relative camera pose prediction (bottom-right) that increases the view sensitivity of the classifier. **(b)** Examples of predicted narrations, and the ranks and scores of the views per our pseudo-labeler, shown alongside ground-truth view-agnostic narrations. "C" refers to the person who is performing the activity. Note that at inference time, there is no ground truth narration, just the video input.

for solving tasks like object recognition [3, 15, 28, 51, 53, 88, 89], reconstruction [52, 55, 88, 89, 95], and semantic segmentation [94, 95] within a time budget. Whereas such setups require moving the camera to capture a better view for the agent performing its task, our goal is to select the best view from multiple cameras recording simultaneously to facilitate human viewing.

Captioning for weak supervision. Prior work studies using ground-truth [25, 105, 119] or predicted [27, 116] captions for weakly supervising action recognition [40], object detection [61, 65, 68, 100, 117, 120], semantic segmentation [109, 111, 113], and visual question answering [6, 39, 114]. On the contrary, we tackle view selection, a distinct task, by using the quality of predicted [64, 122] captions (narrations) as weak supervision.

3. Approach

Our goal is to train a model to identify a sequence of best views for watching a multi-view instructional video, such that the identified views are most informative of the activity in the video. Importantly, we aim to do this in the *absence of best view labels.* We first formally define our task (Sec. 3.1), and then decompose it into three key questions: (1) how to source the best view pseudo-labels to train our model (Sec. 3.2), (2) how to model multi-view videos to discriminate between visually similar views when identifying the best one (Sec. 3.3), and (3) how to train the model (Sec. 3.4).

3.1. View selection task

Given a set of instructional videos with multiple camera views, the goal is to automatically predict views that most comprehensively capture the fine-grained details—minute aspects of the actions and objects involved, precise body movements—of the human activity, and are, consequently, likely useful for skill learning. Critically, we aim to achieve this without any manually provided best-view labels, but by instead using natural language narrations of the videos as a source of weak supervision.

Let V be an instructional video recorded with multiple cameras (Fig. 1). The video V consists of M clips, such that $V = [\mathcal{V}_1, ..., \mathcal{V}_M]$. Each clip $\mathcal{V}_m$ has N RGB image streams, one from each camera/viewpoint, such that $\mathcal{V}_m = \{\mathcal{V}_{m,1}, ..., \mathcal{V}_{m,N}\}$. Our goal is to select the best view $\mathcal{B}^*_m$

for each clip $\mathcal{V}_m$ to create an output video that is ideal for understanding the activity in the video. Therefore, we aim to find $\mathcal{B}^* = [\mathcal{B}^*_1, \ldots, \mathcal{B}^*_M]$, where $\mathcal{B}^*_m \in \{\mathcal{V}_{m,i}\}_{i=1}^N$.

Rather than assume any best-view labels on the training clips $\mathcal{V}_m$, we turn instead to *narrations*—human-provided descriptions of the activity in the video. Narrations are common in instructional videos [21, 36, 37, 56, 74, 124], and for recent multi-view datasets [37, 56] they are specifically gathered in a *view-agnostic* manner: human annotators watch a collage of *all* views of a clip and write down a holistic description of the person's actions and the objects involved. See Fig. 2 (center). While the narrations themselves are a form of annotation, they are more widely available, versatile, and scalable than specialized best-view labels, and hence provide a compelling source of weak supervision.

Each clip thus has a ground-truth view-agnostic narration $\mathcal{N}^*_m$. We stress that each $\mathcal{N}^*_m$ captures the activity as viewed from any and all angles; it is view-independent. The descriptions include details about actions taken by the camera wearer to the activity, and as well as relevant events from the environment and important objects. We aim to use these narrations to train a model $\mathcal{F}$ that, given the video V, predicts a viewpoint sequence i.e., $\mathcal{F}(V) = \mathcal{B}^*$. We stress that the narrations are available only for training videos, not at test time.

3.2. Sourcing best view pseudo-labels for training

Next, we describe our framework (Fig. 2a) for tackling this *language-guided* task. We first source best-view pseudo-labels for training our view selector. We hypothesize that the relative quality of predicted narrations from different views indicates how accurately each view captures the fine details of the activity, and we show how this view-dependent quality of predicted narrations can be used to train a view selector. For example, consider a video showing how to fix a bicycle (Fig 1). Some camera angles may have the person's body or the bike blocking the view, making it hard to see what is happening. As a result, the captioner cannot properly describe the activity using such views, indicating that these views are of poor quality. In contrast, a view that clearly shows the hands, bike parts, tools, etc., will allow the captioner to accurately describe the activity, making it a more informative view.

Best view pseudo-labeler L. We devise our pseudo-labeler L to generate the best-view pseudo-label for a training clip $\mathcal{V}_m$ by first predicting the narration for each view separately, and then scoring the views by comparing their predicted narrations to the *view-agnostic* ground-truth narration $\mathcal{N}^*_m$.[2] We aggregate results over multiple independent captioners in order to bolster robustness, essentially smoothing over their outputs. To this end, we use K off-the-shelf video captioning models: 1) Video-Llama [122] with Llama2 [104] LLM decoder, 2) Video-Llama [122] with Vicuna [104] LLM decoder, and 3) VideoChat2 [64]. See Fig. 2a top.

In particular, we predict the narrations for the N views separately using each captioner, where $\mathcal{N}^k = \{\mathcal{N}^k_1, \ldots, \mathcal{N}^k_N\}$ denotes the predicted narrations from the k^{th} captioner. Next, we pass each set $\mathcal{N}^k$ to a view ranker, which scores its narrations by comparing them to the ground-truth narration $\mathcal{N}^*$ using a standard captioning metric [7, 84, 108], and computes a set of ranks $\mathcal{R}^k$ for the corresponding views. Finally, a best view rank aggregator is used to reach an agreement across all captioners. The rank aggregator extracts the consensus: it takes as input all rank sets $\{\mathcal{R}^1, \ldots, \mathcal{R}^K\}$, finds the views—there could be multiple such views for a captioner if the estimated narration from more than one view produces the same highest score—ranked the highest within each individual captioner, and uses *all* views that are top-ranked across all captioners to build a best view pseudo-label set $\mathcal{B}$.

Essentially, our pseudo-labeler uses multiple strong captioners to rank views based on the accuracy of their predicted narrations, and achieves consensus on the top ranked views, thereby automatically producing high-quality best view pseudo-labels. See Fig. 2b and Supp. for examples.

3.3. Best view selector S

We use the pseudo-label set $\mathcal{B}$ from Sec. 3.2 to train our best view selector S, which must reason across all views to identify the best one [3]. Simply using a view classifier as the selector is not enough in our setting, as our captioning-based labels can sometimes be insufficiently discriminative, *i.e.*, multiple views might be pseudo-labeled as the best view due to the very nature of the off-the-shelf video captioners' training. They were originally designed to learn features that are predictive of the narration regardless of the view, and consequently, can end up collapsing all views into similar representations that are unable to capture the important nuances between them.

To tackle this, we design a view selector composed of 1) a view classifier W, and 2) a relative camera pose predictor P. During training, while our view classifier (Fig. 2a bottom-left) tries to identify the best viewpoint given our pseudo-labels, the pose predictor (Fig. 2a bottom-right) simultaneously acts as a regularizer and mitigates the effect of spurious pseudo-labels by solving an auxiliary task of relative camera pose prediction for each pair of viewpoints. This ensures that the features learned by our view classifier remain sensitive to viewpoint changes, and our model does not overfit to the pseudo-labels during training, thereby

[2] For simplicity, we omit the clip index m from subscript, henceforth.

[3] Note that it is not possible to simply apply the pseudo-labeler at inference time, since it requires access to ground-truth narrations.

improving the quality of view selection, as we will see in results.

View classifier W. Our view classifier W consists of a visual encoder F with a TimeSformer [8, 86] architecture, which encodes each view n into a set of visual features f_n that spatially correspond with frame patches in the input view, where $f_n = F(\mathcal{V}_n)$. Owing to its patch-level nature and the end-to-end training (cf. Sec. 3.4) of the classifier using our pseudo-labels, f_n provides fine-grained cues about the human activity—what parts of it are visible in a viewpoint, and how the dynamic elements (*e.g.* moving objects, body parts) evolve over time, thereby facilitating high-quality view selection. Next, the model uses a projector H^W to embed f_n into a lower-dimensional feature h_n providing a higher-level representation of a view's ability to capture activity details. Formally, $h_n = H^W(f_n)$. Finally, we concatenate h_n from all views, and feed them to a classification head C^W. The classification head compares these representations across views and outputs its best view estimate $\tilde{\mathcal{B}}$, such that $\tilde{\mathcal{B}} = C^W([h_1, \ldots, h_N])$.

Relative camera pose predictor P. Our relative camera pose predictor P uses the view classifier's visual features to predict the relative camera pose for all view pairs. Specifically, we formulate the pose prediction as a classification task [14, 77]. Given the ground-truth relative camera pose $P^*_{(i,j)}$ for an arbitrary pair of viewpoints (i, j), we discretize the angles of its direction of displacement and rotation matrix using bins of a fixed size. This formulation ensures that our pose prediction task is tractable, and helps our model learn view-dependent visual features that improve task performance. That is because it requires predicting the rough direction of one camera center relative to another instead of the exact relative displacement, which is ill-posed due to unknown object sizes.

To perform this classification for a viewpoint pair (i, j), our pose predictor P uses the fine-grained visual features f_i and f_j produced by our view classifier for viewpoints i and j, and embeds them into more abstract representations h^P_i and h^P_j by using a feature projector H^P, such that $[h^P_i, h^P_j] = [H^P(f_i), H^P(f_j)]$. Whereas the fine-grained features f help learn patch-level correspondences [8, 86], which are crucial for accurate pose prediction, the features h act as a bridge between these detailed representations and the higher-level measure of relative pose. Finally, similar to VLocNet++ [87], we concatenate features h^P_i and h^P_j, and pass them to a pose classification head C^P, which computes their inter-feature correlation [10] and outputs a relative pose estimate $P_{i,j}$. Formally, $P_{i,j} = C^P([h^P_i, h^P_j])$. The estimates are used in an auxiliary loss defined below. See Fig. 2a bottom-right.

3.4. Model training

Video captioner training. We train the off-the-shelf video captioners [64, 122] in the best view pseudo-labeler L (cf. Sec. 3.2) by initializing them with the pretrained parameters released by their authors, and subsequently finetuning them on our multi-view videos with the standard negative log-likelihood loss [64, 122]. This helps us leverage the knowledge from internet-scale pretraining and improve the captioning performance.

View selector training. We train our view selector with a combination of two losses: a) a view classification loss $\mathcal{L}^W$, and b) a relative camera pose prediction loss $\mathcal{L}^P$. While $\mathcal{L}^W$ provides a direct learning signal to the view classifier W, $\mathcal{L}^P$ helps its visual encoder F generate visual features that capture the important inter-viewpoint differences.

We propose a novel loss formulation for $\mathcal{L}^W$, which accounts for the cases where there is more than one best view pseudo-label in $\mathcal{B}$ (cf. Sec.3.2). Specifically, we set $\mathcal{L}^W$ to

$$\mathcal{L}^W = min\{\mathcal{L}_{CE}(\tilde{\mathcal{B}}, \hat{\mathcal{B}}) \quad \forall \quad \hat{\mathcal{B}} \in \mathcal{B}\}, \tag{1}$$

where $\tilde{\mathcal{B}}$ is our best view estimate (cf. Sec. 3.3) and $\mathcal{L}_{CE}$ denotes the cross-entropy loss. Our formulation for $\mathcal{L}^W$ encourages our view classifier to learn to predict as its estimate whichever among the pseudo-labels it finds the easiest to predict [38, 54, 79], thereby stabilizing training and improving task performance, as we show in results.

We set the relative camera pose prediction loss $\mathcal{L}^P$ to

$$\mathcal{L}^P = \frac{1}{N^2} \sum_{(i,j) \in \mathbb{N}} \mathcal{L}^P_{CE}(\mathcal{P}_{(i,j)}, \mathcal{P}^*_{(i,j)}). \tag{2}$$

Here, $\mathcal{L}^P_{CE}$ is the average cross-entropy loss over all discretized angles in $\mathcal{P}^*$ (cf. Sec. 3.3), and $\mathbb{N} = \{1, \ldots, N\} \times \{1, \ldots, N\}$ is the set of all possible view pairs.

We set our final training loss $\mathcal{L}^S$ for the view selector to $\mathcal{L}^S = \mathcal{L}^W + w * \mathcal{L}^P$, where w is the weight on the pose prediction loss, and jointly train W and P.

4. Experiments

We overview all setup details and then provide results.

4.1. Experimental setup

Dataset. We evaluate our model on two multi-view instructional video datasets: Ego-Exo4D [37] and LEMMA [56][4]. Ego-Exo4D contains both physical (*e.g.* basketball, dancing) and procedural (*e.g.* cooking, bike repair) activities with each video containing 5 time-synced views—one is egocentric (ego) and recorded by the participant with a head-worn camera, and the remaining views are exocentric (exo)

[4]Note that other instructional video datasets like HowTo100M [74], CrossTask [125], and COIN [103] are single-view and thus not suitable.

and recorded with stationary cameras kept around the scene. LEMMA has videos of people performing household activities (*e.g.* making juice, watering plant), where each video has the participant's ego view and an exo view from a fixed camera placed facing the activity. Both have narrations: the Ego-Exo4D annotators provide temporally dense written descriptions of the participants' actions and relevant objects involved in the activity, while the LEMMA annotators describe action verbs and objects being interacted with using a pre-defined vocabulary. These two datasets let us evaluate different exo view setups (multiple in EgoExo-4D vs. single in LEMMA) and diverse activity scenarios (physical and procedural in Ego-Exo4D vs. household in LEMMA). Ego-Exo4D and LEMMA provide a total of 86 and 20 hours of video data, resulting in 648,665 and 63,538 clip-narration pairs, respectively. See Supp. for details, including our clip segmentation strategy.

Implementation. We use $K = 3$ captioners in our pseudo-labeler: Video-Llama [122] with Llama-2-Chat [104] or Vicuna [17] as the LLM decoder, and VideoChat2 [64], respectively, and the CIDEr [108] captioning metric to score views. We finetune the captioners on our datasets before using them to score views. On the basis of a disjoint validation set, we set the bin size to 30 degrees for generating relative camera pose labels (cf. Sec. 3.3), and the weight on the pose prediction loss to $w = 0.5$. See Supp. for more details.

Baselines. We compare against the following baselines and state-of-the-art methods:

- **Ego-only, Random, Random-exo:** a set of naive baselines that predict the ego view (**Ego-only**), or a view randomly chosen from just exo (**Random-exo**) or all (**Random**) views, as the best view.
- **Hand-object, Body-area, Joint-count:** a set of baselines that predict the view with the highest hand and object detection confidence per a state-of-the-art hand and object detector [16] (**Hand-object**), or the largest body area (**Body-area**) or joint count (**Joint-count**) per a state-of-the-art body pose detector [57], as the best view.
- **Snap angles [12, 112]:** an automatic cinematography method that predicts the view with the highest foreground pixel count as the best view. We upgrade it to use today's SOTA segmentation models [60, 68].
- **Longest-caption:** a baseline that uses our finetuned Video-Llama captioner [122] to predict captions for each input view, and selects the one that produces the longest narration as the best view. Intuitively, here caption length is used a proxy for informativeness. Recall that our model does not infer captions on test data.

While the first set of baselines are naive heuristics to test if intelligent view selection is even necessary, the second set accounts for the prior that visibility of people and person-object interactions are strongly linked to the informativeness of a view, and the third represents the most relevant existing methods in the literature. Finally, the Longest-caption baseline offers an alternative, more naive way to incorporate language for our task.

Evaluation metrics. We perform both *automatic* and *human* evaluation. For automatic evaluation, we measure how well the selected view predicts two things: narrations and action/object terms. For the former, we use a state-of-the-art video captioner [64, 122] to predict the narrations given our chosen views, and then compare the predictions with the view-agnostic ground-truth narrations through standard captioning metrics: **CIDEr** [108] and **METEOR** [7]. For the latter, we use 1) **Verb IoU (V-IoU)** , 2) **Noun IoU (N-IoU)**, and 3) **Noun-chunk IoU (NC-IoU)**, which measure the overlap in the sets of verbs, nouns, and noun chunks between the ground-truth and predicted narrations, where noun chunks are nouns grouped with their modifiers (*e.g.* adjective, article). In short, the more the view deemed as "best" by our model predicts things consistent with the comprehensive view-agnostic ground truth, the better it is. We stress that this suite of metrics, together with the human evaluation below, goes beyond CIDEr (used in our language-guided training) to gauge the quality of the selected views.

Through **human evaluation**, we assess two important model aspects: 1) pseudo-label quality, and 2) view selection performance. We conduct both assessments through pairwise comparison of views, which reduces the cognitive load on human judges and increases their reliability [2, 20, 34]. Given a view pair, the human judge can select either view or both, depending on if they prefer one over the other specifically for the purpose of activity understanding (i.e., a *win* for the preferred view, and a *loss* for the other), or find them equally informative (*tie*). Critically, we do not show the evaluators the ground-truth narrations. In other words, our study *directly evaluates the human-preferred viewpoints*—independent of narrations—and hence is unbiased by the fact our model leverages language during training. We obtain human judgments on 1) pseudo-label quality by pairing the best and the worst views per our pseudo-labeler (cf. Sec. 3.2), and 2) view prediction quality by pairing the views predicted by our and the best baseline's predicted views. We decide the view order in each pair randomly. We do each study for both datasets with 10 participants and 70 randomly chosen test view pairs. Our inter-evaluator agreement rate is 78.5%.

4.2. Results

Automatic evaluation. Table 1 top shows our results for automatic evaluation. The naive baselines are generally the worst performers, indicating that blindly choosing the ego view at all times, or picking a random viewpoint is not enough for our challenging task. Interestingly, while random-

Model	Ego-Exo4D [37]					LEMMA [56]				
	Captioning		*Actions and objects*			*Captioning*		*Actions and objects*		
	CIDEr [108]	METEOR [7]	V-IoU	N-IoU	NC-IoU	CIDEr [108]	METEOR [7]	V-IoU	N-IoU	NC-IoU
Ego-only	12.2	47.2	32.2	36.7	30.6	41.7	71.1	38.2	41.3	17.5
Random	11.5	45.9	30.4	36.6	31.0	30.9	63.1	31.2	33.2	12.8
Random-exo	11.9	46.0	30.5	37.0	30.9	17.7	51.3	21.6	22.4	6.8
Hand-object	12.6	47.4	33.6	36.7	29.6	40.7	72.7	38.5	41.5	17.9
Body-area	12.9	48.2	32.5	37.2	31.1	42.1	73.8	38.6	41.3	17.6
Joint-count	12.6	46.6	31.5	29.1	27.7	17.8	51.4	21.7	22.4	6.7
Snap angles [12, 112]	12.2	46.7	30.7	35.8	29.1	38.9	70.6	37.1	40.2	17.1
Longest-caption	10.7	47.3	30.5	34.6	28.8	32.7	65.4	36.9	37.9	15.3
LANGVIEW (Ours)	**13.5**	**48.4**	**33.7**	**39.2**	**32.9**	**42.7**	**74.4**	**40.1**	**42.9**	**18.9**

Table 1. View selection results. All metrics are in % and higher is better.

exo improves over ego-only and random on Ego-Exo4D [37], it fares worse on LEMMA [56], possibly because activities like rock climbing or basketball in Ego-Exo4D involve more head and body motion than the household activities (*e.g.* cooking, watching TV) in LEMMA, and consequently, require stationary exo cameras for better coverage. Using intelligent heuristics like hand and object detection confidence (Hand-object), body visibility (Joint-count and Body-area), or foreground object prevalence (Snap angles [12, 112]) generally improve task performance, showing that these methods provide useful cues for our task. However, the Longest-caption baseline generally underperforms the naive baselines, possibly because the view with the longest predicted narration provides excessive details about the scene, which are irrelevant to the activity.

Our model significantly outperforms all baselines across metrics, on both datasets. It shows our idea of pseudo-labeling by leveraging the view-dependent quality of predicted narrations is effective in practice. Furthermore, our model's improvement over the heuristics illustrates that our language-guided training facilitates complex but essential reasoning about the interplay between human actors and interacting objects, more than what is possible with hand and object, or body pose detectors. Finally, our superior performance on both datasets underlines the efficacy of our design and its ability to generalize to different activity types—both physical and non-physical in Ego-Exo4D vs. household in LEMMA— and camera setups—single vs. multiple exo cameras in LEMMA and Ego-Exo4D, respectively. See Supp. for results on the single exo camera and 3-fold evaluation with Ego-Exo4D.

Human evaluation. Table 2 shows our human evaluation results using *win*, *loss* and *tie* percentages. In our pseudo-label quality study, the human preference for the best view per our pseudo-labels is significantly higher than the worst view. Given that the participants were asked to base their responses on the views' suitability for activity understanding, this validates our hypothesis that the view-specific quality of predicted narrations is correlated with human preference for informative views, and can be exploited for sourcing

Assessment type	Win	Loss	Tie
Pseudo-label: best vs. worst view			
Ego-Exo4D [37]	**53.3**	28.9	17.8
LEMMA [56]	**46.7**	38.9	14.4
View prediction: ours vs best baselines			
Ego-Exo4D [37]			
Ours vs. Hand-object	**52.2**	40.0	7.8
Ours vs. Body-area	**55.1**	39.3	5.6
LEMMA [56]			
Ours vs. Hand-object	**43.3**	41.1	15.6
Ours vs. Body-area	**46.7**	35.5	17.8

Table 2. Human evaluation results for our pseudo-label and view prediction quality. All values are in %. Significance, $p \leq 0.05$.

best view pseudo-labels. Furthermore, a significant *tie* rate indicates a considerable presence of instances with multiple high-quality views, re-emphasizing the challenging nature of our task.

In our view selection assessment, we find that our selected views are preferred significantly more than the two top baselines on both datasets. We achieve a strong win rate boost of at least ∼11% over the baselines across different evaluation scenarios. The only exception is our model vs. Hand-object [16] on LEMMA, where the improvement margin is a more modest ∼2%, likely because the view that best shows hand-object interactions is a reasonably good view for showing LEMMA-style activity (cooking, watering plant, etc.). These results show that our weakly-supervised method is better capable of implicitly modeling human view preference and using this model to perform more accurate view selection. Finally, our consistent performance gains on human evaluation reinforce our automatic evaluation metrics, and all model analysis we do using them.

Qualitative examples. Fig. 3 (left) shows some success cases. Note how our model chooses views that clearly show all the essential elements of the actions, including the different entities involved and and their motion. E.g., in clip 2, our model chooses the ego view as it clearly shows the horizontal knife motion and vegetables on the cutting board, whereas in clip 3, it selects an exo view that best shows the joined hands

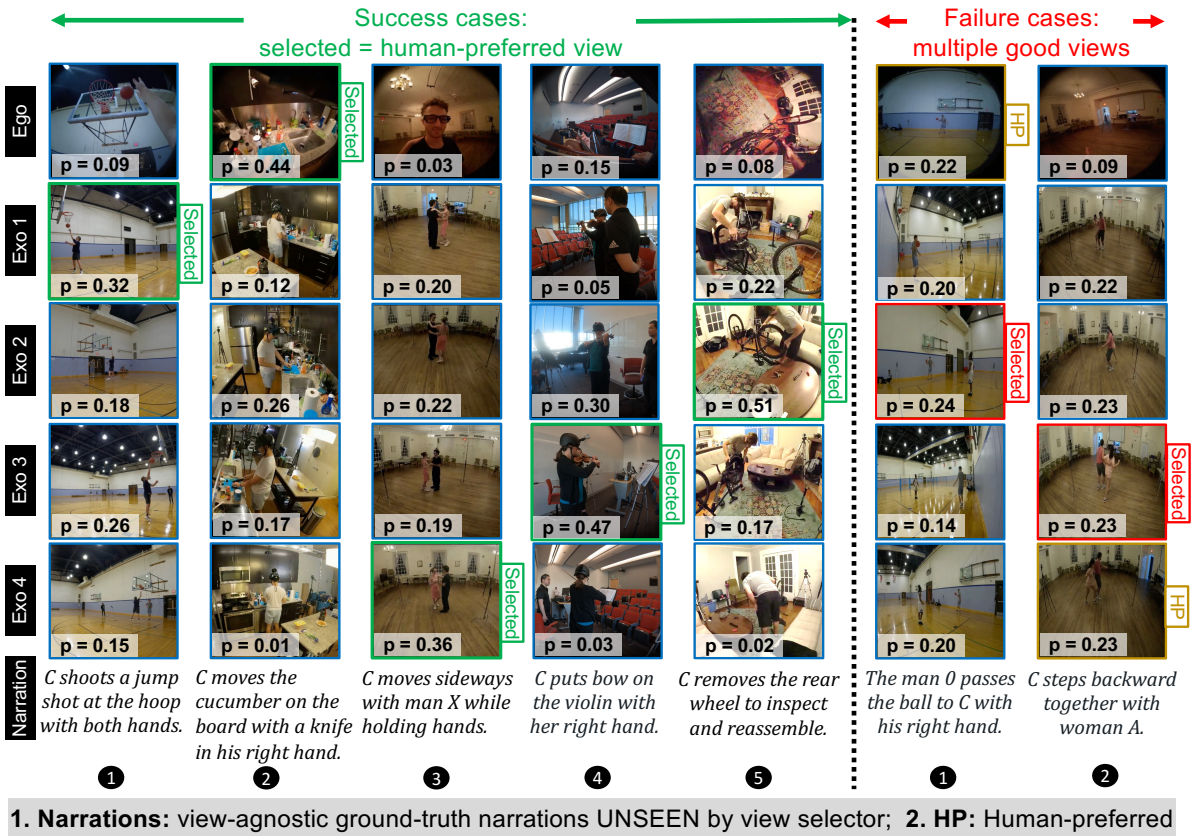


Figure 3. Left: sample successful predictions by our view selector. For each clip, our model chooses the view that shows the action, and the objects and body parts involved in it, most clearly, and hence, is most informative. **Right**: Sample failure cases for our model, where there are multiple high-quality views that differ only in certain nuances, which are discernible by a human but not our model trained through narration guidance. Whereas humans prefer a view that better captures the direction of the ball towards the camera-wearer in sample 1, or shows the full backward motion of the dancers in sample 2, our model choose a view that shows all entities mentioned in the narration.

and the sideways movement of the dancers. Thus, depending on the activity, our model adaptively chooses a view that accurately shows its important details. See Supp. video for more examples.

We also observe two common failure types. The first type occurs when there are multiple high-quality views. See Fig. 3 (right). In the second one, our model chooses different views for very similar activities, which involve very similar types, positions and motion of relevant objects and body parts, even when the views are not equally good. This occurs possibly because our selector occasionally picks on spurious cues that are not as viewpoint-dependent as the activity itself but also do not help with its understanding.

Ablations. In Table 3 (top) we report our model ablation results on the large-scale Ego-Exo4D [37] dataset. Not predicting inter-view camera pose in training significantly hurts performance on all metrics except METEOR [7]. This shows that our pose predictor enhances the selector's view sensitivity. Training our model with a standard cross-entropy loss by randomly choosing a sample from our pseudo-label set, instead of our proposed loss (c.f. Sec. 3.4) hurts performance on N- and NC-IoU. This occurs possibly because the captioners in our pseudo-labeler sometimes hallucinate and add less informative views to the pseudo-label set, which when sampled causes training instability. Removing the rank aggregator to obtain pseudo-label consensus also negatively impacts performance on N- and NC-IoU, as having multiple captioners vote on the best view reduces captioning noise and improves pseudo-label quality. Finally, removing all three components consistently degrades performance across

	Captioning		*Actions and objects*		
Model	CIDEr	METEOR	V-IoU	N-IoU	NC-IoU
Ours w/o P	13.2	**49.2**	33.4	38.7	32.8
Ours w/o $\mathcal{L}^W$	13.5	48.8	34.0	37.4	32.3
Ours w/o G	**13.7**	48.7	**34.3**	38.5	32.8
Ours w/o G, $\mathcal{L}^W$, P	13.2	48.0	33.5	37.7	32.5
Ours	13.5	48.4	33.7	**39.2**	**32.9**

Table 3. Ablation results on the large-scale Ego-Exo4D [37] dataset. G denotes the rank aggregator in our pseudo-labeler. All metrics are in %. Significance, $p \leq 0.05$.

metrics, showing that the model needs *at least one component* to perform well on different metrics.

See Supp. for additional ablations, and analyses of the view dependence of our visual features, our pseudo-labeler, and the impact of the rank of our selector's sampled view on view selection performance, and our model's attention maps.

5. Conclusion

We tackle view selection in multi-view instructional videos in the absence of best view labels. To that end, we design a novel framework composed of a best view pseudo-labeler that uses the view-dependent quality of estimates of video descriptions to automatically generate best view pseudo-labels, and a best view selector that given a video, produces a best view prediction. Our method significantly outperforms several state-of-the-art baselines on two challenging multi-view instructional video datasets. In future work, we will explore future best-view anticipation for improving the energy efficiency of multi-view instructional video capture setups.

Acknowledgements: UT Austin is supported in part by the IFML NSF AI Institute. KG is paid as a research scientist by Meta, and SM was a visiting researcher at the same when this work was done.

References

[1] Abien Fred Agarap. Deep learning using rectified linear units (relu), 2018. cite arxiv:1803.08375Comment: 7 pages, 11 figures, 9 tables. 19

[2] Milind Agarwal, Sweta Agrawal, Antonios Anastasopoulos, Luisa Bentivogli, Ondřej Bojar, Claudia Borg, Marine Carpuat, Roldano Cattoni, Mauro Cettolo, Mingda Chen, William Chen, Khalid Choukri, Alexandra Chronopoulou, Anna Currey, Thierry Declerck, Qianqian Dong, Kevin Duh, Yannick Estève, Marcello Federico, Souhir Gahbiche, Barry Haddow, Benjamin Hsu, Phu Mon Htut, Hirofumi Inaguma, Dávid Javorský, John Judge, Yasumasa Kano, Tom Ko, Rishu Kumar, Pengwei Li, Xutai Ma, Prashant Mathur, Evgeny Matusov, Paul McNamee, John P. McCrae, Kenton Murray, Maria Nadejde, Satoshi Nakamura, Matteo Negri, Ha Nguyen, Jan Niehues, Xing Niu, Atul Kr. Ojha, John E. Ortega, Proyag Pal, Juan Pino, Lonneke van der Plas, Peter Polák, Elijah Rippeth, Elizabeth Salesky, Jiatong Shi, Matthias Sperber, Sebastian Stüker, Katsuhito Sudoh, Yun Tang, Brian Thompson, Kevin Tran, Marco Turchi, Alex Waibel, Mingxuan Wang, Shinji Watanabe, and Rodolfo Zevallos. FINDINGS OF THE IWSLT 2023 EVALUATION CAMPAIGN. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 1–61, Toronto, Canada (in-person and online), 2023. Association for Computational Linguistics. 6

[3] Phil Ammirato, Patrick Poirson, Eunbyung Park, Jana Košecká, and Alexander C Berg. A dataset for developing and benchmarking active vision. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1378–1385. IEEE, 2017. 3

[4] Ido Arev, Hyun Soo Park, Yaser Sheikh, Jessica Hodgins, and Ariel Shamir. Automatic editing of footage from multiple social cameras. *ACM Trans. Graph.*, 33(4), 2014. 2

[5] R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76 (8):966–1005, 1988. 2

[6] Pratyay Banerjee, Tejas Gokhale, Yezhou Yang, and Chitta Baral. Weaqa: Weak supervision via captions for visual question answering. *arXiv preprint arXiv:2012.02356*, 2020. 3

[7] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, 2005. Association for Computational Linguistics. 4, 6, 7, 8, 15, 19

[8] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, page 4, 2021. 5, 18

[9] Uttaran Bhattacharya, Gang Wu, Stefano Petrangeli, Viswanathan Swaminathan, and Dinesh Manocha. Highlightme: Detecting highlights from human-centric videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8157–8167, 2021. 2

[10] Ruojin Cai, Bharath Hariharan, Noah Snavely, and Hadar Averbuch-Elor. Extreme rotation estimation using dense correlation volumes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14566–14575, 2021. 5

[11] Sijia Cai, Wangmeng Zuo, Larry S. Davis, and Lei Zhang. Weakly-supervised video summarization using variational encoder-decoder and web prior. In *Computer Vision – ECCV 2018 - 15th European Conference, 2018, Proceedings*, pages 193–210. Springer-Verlag, 2018. 15th European Conference on Computer Vision, ECCV 2018 ; Conference date: 08-09-2018 Through 14-09-2018. 2

[12] Seunghoon Cha, Jungjin Lee, Seunghwa Jeong, Younghui Kim, and Junyong Noh. Enhanced interactive 360° viewing via automatic guidance. *ACM Trans. Graph.*, 39(5), 2020. 6, 7, 18, 19, 20

[13] Jianhui Chen, Keyu Lu, Sijia Tian, and Jim Little. Learning sports camera selection from internet videos. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1682–1691. IEEE, 2019. 2

[14] Kefan Chen, Noah Snavely, and Ameesh Makadia. Wide-baseline relative camera pose estimation with directional learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3258–3268, 2021. 5

[15] Ricson Cheng, Ziyan Wang, and Katerina Fragkiadaki. Geometry-aware recurrent neural networks for active visual recognition. *Advances in Neural Information Processing Systems*, 31, 2018. 3

[16] Tianyi Cheng, Dandan Shan, Ayda Sultan Hassen, Richard Ely Locke Higgins, and David Fouhey. Towards a richer 2d understanding of hands at scale. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 6, 7, 19

[17] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 6

[18] Shih-Han Chou, Yi-Chun Chen, Kuo-Hao Zeng, Hou-Ning Hu, Jianlong Fu, and Min Sun. Self-view grounding given a narrated 360 {\deg} video. *arXiv preprint arXiv:1711.08664*, 2017. 2

[19] Wen-Sheng Chu, Yale Song, and Alejandro Jaimes. Video co-summarization: Video summarization by visual co-occurrence. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3584–3592, 2015. 2

[20] Andrew P. Clark, Kate L. Howard, Andy T. Woods, Ian S. Penton-Voak, and Christof Neumann. Why rate when you could compare? using the "elochoice" package to assess pairwise comparisons of perceived physical strength. *PLOS ONE*, 13(1):1–16, 2018. 6

[21] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *European Conference on Computer Vision (ECCV)*, 2018. 4

[22] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023. 18

[23] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359, 2022. 18

[24] Hannah Mary Dee and Sergio A. Velastin. How close are we to solving the problem of automated visual surveillance? a review of real-world surveillance, scientific progress and evaluative mechanisms. *Machine Vision and Applications*, 19(5-6):329–343, 2008. Dee, H. M.; Velastin, S. A. How close are we to solving the problem of automated visual surveillance? A review of real-world surveillance, scientific progress and evaluative mechanisms Machine Vision and Applications, volume 19(5-6) pages 329-343, Springer, October 2008. Sponsorship: EPSRC. 1

[25] Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11162–11173, 2021. 3

[26] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 18

[27] Sivan Doveh, Assaf Arbelle, Sivan Harary, Roei Herzig, Donghyun Kim, Paola Cascante-Bonilla, Amit Alfassy, Rameswar Panda, Raja Giryes, Rogerio Feris, et al. Dense and aligned captions (dac) promote compositional reasoning in vl models. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[28] Ruoyi Du, Wenqing Yu, Heqing Wang, Ting-En Lin, Dongliang Chang, and Zhanyu Ma. Multi-view active fine-grained visual recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1568–1578, 2023. 3

[29] Mohamed Elfeki, Liqiang Wang, and Ali Borji. Multi-stream dynamic video summarization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 339–349, 2022. 2

[30] David K. Elson and Mark O. Riedl. A lightweight intelligent virtual cinematography system for machinima production. In *Proceedings of the Third AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, page 8–13. AAAI Press, 2007. 2

[31] J. Foote and D. Kimber. Flycam: practical panoramic video and automatic camera control. In *2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532)*, pages 1419–1422 vol.3, 2000. 2

[32] Yanwei Fu, Yanwen Guo, Yanshu Zhu, Feng Liu, Chuanming Song, and Zhi-Hua Zhou. Multi-view video summarization. *IEEE Transactions on Multimedia*, 12(7):717–729, 2010. 2

[33] Michael L. Gleicher, Rachel M. Heck, and Michael N. Wallick. A framework for virtual videography. In *Proceedings of the 2nd International Symposium on Smart Graphics*, page 9–16, New York, NY, USA, 2002. Association for Computing Machinery. 2

[34] Lucas Goncalves, Prashant Mathur, Chandrashekhar Lavania, Metehan Cekic, Marcello Federico, and Kyu J Han. Peavs: Perceptual evaluation of audio-visual synchrony grounded in viewers' opinion scores. *arXiv preprint arXiv:2404.07336*, 2024. 6

[35] Boqing Gong, Wei-Lun Chao, Kristen Grauman, and Fei Sha. Diverse sequential subset selection for supervised video summarization. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2014. 2

[36] 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. 4, 18

[37] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. *arXiv preprint arXiv:2311.18259*, 2023. 1, 2, 4, 5, 7, 8, 15, 16, 18, 19, 20

[38] Karol Gregor, George Papamakarios, Frederic Besse, Lars Buesing, and Theophane Weber. Temporal difference variational auto-encoder. *arXiv preprint arXiv:1806.03107*, 2018. 5

[39] Jiaxian Guo, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Boyang Li, Dacheng Tao, and Steven CH Hoi. From images to textual prompts: Zero-shot vqa with frozen large language models. *arXiv preprint arXiv:2212.10846*, 2022. 3

[40] Sonal Gupta and Raymond Mooney. Using closed captions as supervision for video activity recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 24(1): 1083–1088, 2010. 3

[41] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos. In *Computer Vision – ECCV 2014*, pages 505–520, Cham, 2014. Springer International Publishing. 2

[42] Michael Gygli, Helmut Grabner, and Luc Van Gool. Video summarization by learning submodular mixtures of objectives. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3090–3098, 2015. 2

[43] Bo He, Jun Wang, Jielin Qiu, Trung Bui, Abhinav Shrivastava, and Zhaowen Wang. Align and attend: Multimodal summarization with dual contrastive losses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14867–14878, 2023. 2

[44] Li-wei He, Michael F. Cohen, and David H. Salesin. The virtual cinematographer: a paradigm for automatic real-time camera control and directing. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, page 217–224, New York, NY, USA, 1996. Association for Computing Machinery. 2

[45] Li-wei He, Michael F. Cohen, and David H. Salesin. *The Virtual Cinematographer: A Paradigm for Automatic Real-Time Camera Control and Directing*. Association for Computing Machinery, New York, NY, USA, 1 edition, 2023. 2

[46] Rachel Heck, Michael Wallick, and Michael Gleicher. Virtual videography. In *Proceedings of the 14th ACM International Conference on Multimedia*, page 961–962, New York, NY, USA, 2006. Association for Computing Machinery. 2

[47] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 18

[48] Hou-Ning Hu, Yen-Chen Lin, Ming-Yu Liu, Hsien-Tzu Cheng, Yung-Ju Chang, and Min Sun. Deep 360 pilot: Learning a deep agent for piloting through 360deg sports videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3451–3460, 2017. 2

[49] Yifei Huang, Guo Chen, Jilan Xu, Mingfang Zhang, Lijin Yang, Baoqi Pei, Hongjie Zhang, Lu Dong, Yali Wang, Limin Wang, et al. Egoexolearn: A dataset for bridging asynchronous ego-and exo-centric view of procedural activities in real world. *arXiv preprint arXiv:2403.16182*, 2024. 1, 2

[50] Sergey Ioffe and Christian Szegedy. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, page 448–456. JMLR.org, 2015. 19

[51] Dinesh Jayaraman and Kristen Grauman. Look-ahead before you leap: end-to-end active recognition by forecasting the effect of motion. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 489–505. Springer, 2016. 3

[52] Dinesh Jayaraman and Kristen Grauman. Learning to look around: Intelligently exploring unseen environments for unknown tasks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1238–1247, 2018. 3

[53] Dinesh Jayaraman and Kristen Grauman. End-to-end policy learning for active visual categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1601–1614, 2019. 3

[54] Dinesh Jayaraman, Frederik Ebert, Alexei A Efros, and Sergey Levine. Time-agnostic prediction: Predicting predictable video frames. *arXiv preprint arXiv:1808.07784*, 2018. 5

[55] Abhishek Jha, Soroush Seifi, and Tinne Tuytelaars. Simglim: Simplifying glimpse based active visual reconstruction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 269–278, 2023. 3

[56] Baoxiong Jia, Yixin Chen, Siyuan Huang, Yixin Zhu, and Song-chun Zhu. Lemma: A multi-view dataset for le arning m ulti-agent m ulti-task a ctivities. In *European Conference on Computer Vision*, pages 767–786. Springer, 2020. 2, 4, 5, 7, 16, 18, 20

[57] Tao Jiang, Peng Lu, Li Zhang, Ning Ma, Rui Han, Chengqi Lyu, Yining Li, and Kai Chen. Rtmpose: Real-time multi-person pose estimation based on mmpose. *ArXiv*, abs/2303.07399, 2023. 6, 18, 19

[58] Aditya Khosla, Raffay Hamid, Chih-Jen Lin, and Neel Sundaresan. Large-scale video summarization using web-image priors. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2698–2705, 2013. 2

[59] Gunhee Kim and Eric P. Xing. Reconstructing storyline graphs for image recommendation from web community photos. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3882–3889, 2014. 2

[60] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 6, 20

[61] Fanjie Kong, Yanbei Chen, Jiarui Cai, and Davide Modolo. Hyperbolic learning with synthetic captions for open-world detection. *arXiv preprint arXiv:2404.05016*, 2024. 3

[62] Sangho Lee, Jinyoung Sung, Youngjae Yu, and Gunhee Kim. A memory network approach for story-based temporal summarization of 360° videos. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1410–1419, 2018. 2

[63] Yong Jae Lee and Kristen Grauman. Predicting important objects for egocentric video summarization. *International Journal of Computer Vision*, 114(1):38–55, 2015. 2

[64] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. *arXiv preprint arXiv:2311.17005*, 2023. 2, 3, 4, 5, 6, 15, 18

[65] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, Kai-Wei Chang, and Jianfeng Gao. Grounded language-image pre-training. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10955–10965, 2022. 3

[66] Yandong Li, Liqiang Wang, Tianbao Yang, and Boqing Gong. How local is the local diversity? reinforcing sequential determinantal point processes with dynamic ground sets for supervised video summarization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 151–167, 2018. 2

[67] Kevin Qinghong Lin, Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Z Xu, Difei Gao, Rong-Cheng Tu, Wenzhe Zhao, Weijie Kong, et al. Egocentric video-language pretraining. *Advances in Neural Information Processing Systems*, 35:7575–7586, 2022. 18

[68] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 3, 6, 20

[69] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 18

[70] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 18, 20

[71] Zheng Lu and Kristen Grauman. Story-driven summarization for egocentric video. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, page 2714–2721, USA, 2013. IEEE Computer Society. 2

[72] Sagnik Majumder, Tushar Nagarajan, Ziad Al-Halah, and Kristen Grauman. Switch-a-view: Few-shot view selection learned from edited videos. *arXiv preprint arXiv:2412.18386*, 2024. 2

[73] Jingjing Meng, Suchen Wang, Hongxing Wang, Yap-Peng Tan, and Junsong Yuan. Video summarization via multi-view representative selection. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 1189–1198, 2017. 2

[74] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2630–2640, 2019. 1, 2, 4, 5, 18

[75] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 1

[76] Peter Mindek, Ladislav Čmolík, Ivan Viola, Eduard Gröller, and Stefan Bruckner. Automatized summarization of multiplayer games. In *Proceedings of the 31st Spring Conference on Computer Graphics*, page 73–80, New York, NY, USA, 2015. Association for Computing Machinery. 2

[77] Tushar Nagarajan, Santhosh Kumar Ramakrishnan, Ruta Desai, James Hillis, and Kristen Grauman. Egoenv: Human-centric environment representations from egocentric video. *Advances in Neural Information Processing Systems*, 36: 60130–60143, 2023. 5

[78] Medhini Narasimhan, Arsha Nagrani, Chen Sun, Michael Rubinstein, Trevor Darrell, Anna Rohrbach, and Cordelia Schmid. Tl; dw? summarizing instructional videos with task relevance and cross-modal saliency. In *European Conference on Computer Vision*, pages 540–557. Springer, 2022. 2

[79] Alexander Neitz, Giambattista Parascandolo, Stefan Bauer, and Bernhard Schölkopf. Adaptive skip intervals: Temporal abstraction for recurrent dynamical models. *Advances in Neural Information Processing Systems*, 31, 2018. 5

[80] Chong-Wah Ngo, Yu-Fei Ma, and Hong-Jiang Zhang. Automatic video summarization by graph modeling. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 104–109 vol.1, 2003. 2

[81] Rameswar Panda and Amit K Roy-Chowdhury. Collaborative summarization of topic-related videos. In *Proceedings of the IEEE Conference on computer vision and pattern recognition*, pages 7083–7092, 2017. 2

[82] Rameswar Panda and Amit K Roy-Chowdhury. Multi-view surveillance video summarization via joint embedding and sparse optimization. *IEEE Transactions on Multimedia*, 19 (9):2010–2021, 2017. 2

[83] Rameswar Panda, Abir Das, Ziyan Wu, Jan Ernst, and Amit K. Roy-Chowdhury. Weakly supervised summarization of web videos. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3677–3686, 2017. 2

[84] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, 2002. Association for Computational Linguistics. 4

[85] Jungin Park, Jiyoung Lee, Ig-Jae Kim, and Kwanghoon Sohn. Sumgraph: Video summarization via recursive graph modeling. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pages 647–663. Springer, 2020. 2

[86] Shraman Pramanick, Yale Song, Sayan Nag, Kevin Qinghong Lin, Hardik Shah, Mike Zheng Shou, Rama Chellappa, and Pengchuan Zhang. Egovlpv2: Egocentric video-language pre-training with fusion in the backbone. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5285–5297, 2023. 5, 18, 20

[87] Noha Radwan, Abhinav Valada, and Wolfram Burgard. Vlocnet++: Deep multitask learning for semantic visual localization and odometry. *IEEE Robotics and Automation Letters*, 3(4):4407–4414, 2018. 5

[88] Santhosh K Ramakrishnan and Kristen Grauman. Sidekick policy learning for active visual exploration. In *Proceedings of the European conference on computer vision (ECCV)*, pages 413–430, 2018. 3

[89] Santhosh K Ramakrishnan, Dinesh Jayaraman, and Kristen Grauman. Emergence of exploratory look-around behaviors through active observation completion. *Science Robotics*, 4 (30):eaaw6326, 2019. 3

[90] 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. 18

[91] Mrigank Rochan and Yang Wang. Video summarization by learning from unpaired data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7902–7911, 2019. 2

[92] Mrigank Rochan, Mahesh Kumar Krishna Reddy, Linwei Ye, and Yang Wang. Adaptive video highlight detection by learning from user history. In *Computer Vision – ECCV 2020*, pages 261–278, Cham, 2020. Springer International Publishing. 2

[93] Abhimanyu Sahu and Ananda S. Chowdhury. Shot level egocentric video co-summarization. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 2887–2892, 2018. 2

[94] Soroush Seifi and Tinne Tuytelaars. Attend and segment: Attention guided active semantic segmentation. In *European Conference on Computer Vision*, pages 305–321. Springer, 2020. 3

[95] Soroush Seifi, Abhishek Jha, and Tinne Tuytelaars. Glimpse-attend-and-explore: Self-attention for active visual exploration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16137–16146, 2021. 3

[96] Gunnar A Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. Actor and observer: Joint modeling of first and third-person videos. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7396–7404, 2018. 1

[97] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5179–5187, 2015. 2

[98] Yu-Chuan Su and Kristen Grauman. Making 360° video watchable in 2d: Learning videography for click free viewing. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1368–1376, 2017. 2

[99] Yu-Chuan Su, Dinesh Jayaraman, and Kristen Grauman. Pano2vid: Automatic cinematography for watching 360 videos. In *Asian Conference on Computer Vision*, pages 154–171. Springer, 2016. 2

[100] Chen Sun, Chuang Gan, and Ram Nevatia. Automatic concept discovery from parallel text and visual corpora. In *Proceedings of the IEEE international conference on computer vision*, pages 2596–2604, 2015. 3

[101] Xinding Sun, J. Foote, D. Kimber, and B. S. Manjunath. Region of interest extraction and virtual camera control based on panoramic video capturing. *Trans. Multi.*, 7(5):981–990, 2005. 2

[102] Hao Tang, Dan Xu, Nicu Sebe, Yanzhi Wang, Jason J Corso, and Yan Yan. Multi-channel attention selection gan with cascaded semantic guidance for cross-view image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2417–2426, 2019. 1

[103] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1207–1216, 2019. 5

[104] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 4, 6

[105] Michael Tschannen, Manoj Kumar, Andreas Steiner, Xiaohua Zhai, Neil Houlsby, and Lucas Beyer. Image captioners are scalable vision learners too. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[106] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9 (86):2579–2605, 2008. 16

[107] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 19

[108] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575, 2014. 4, 6, 7, 15, 19

[109] Vibashan VS, Ning Yu, Chen Xing, Can Qin, Mingfei Gao, Juan Carlos Niebles, Vishal M Patel, and Ran Xu. Mask-free ovis: Open-vocabulary instance segmentation without manual mask annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23539–23549, 2023. 3

[110] Jiaxin Wu, Sheng-Hua Zhong, and Yan Liu. Mvsgcn: A novel graph convolutional network for multi-video summarization. In *Proceedings of the 27th ACM International Conference on Multimedia*, page 827–835, New York, NY, USA, 2019. Association for Computing Machinery. 2

[111] Jianzong Wu, Xiangtai Li, Henghui Ding, Xia Li, Guangliang Cheng, Yunhai Tong, and Chen Change Loy. Betrayed by captions: Joint caption grounding and generation for open vocabulary instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 21938–21948, 2023. 3

[112] Bo Xiong and Kristen Grauman. Snap angle prediction for 360° panoramas. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 2, 6, 7, 18, 19, 20

[113] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2955–2966, 2023. 3

[114] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Learning to answer visual questions from web videos. *arXiv preprint arXiv:2205.05019*, 2022. 3

[115] Huan Yang, Baoyuan Wang, Stephen Lin, David Wipf, Minyi Guo, and Baining Guo. Unsupervised extraction of video highlights via robust recurrent auto-encoders. In *Pro-*

ceedings of the IEEE international conference on computer vision, pages 4633–4641, 2015. 2

[116] Kaicheng Yang, Jiankang Deng, Xiang An, Jiawei Li, Ziyong Feng, Jia Guo, Jing Yang, and Tongliang Liu. Alip: Adaptive language-image pre-training with synthetic caption. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2922–2931, 2023. 3

[117] Keren Ye, Mingda Zhang, Adriana Kovashka, Wei Li, Danfeng Qin, and Jesse Berent. Cap2det: Learning to amplify weak caption supervision for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 3

[118] Qinghao Ye, Xiyue Shen, Yuan Gao, Zirui Wang, Qi Bi, Ping Li, and Guang Yang. Temporal cue guided video highlight detection with low-rank audio-visual fusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7950–7959, 2021. 2

[119] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 3

[120] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14393–14402, 2021. 3

[121] Cha Zhang, Yong Rui, Jim Crawford, and Li-Wei He. An automated end-to-end lecture capture and broadcasting system. *ACM Trans. Multimedia Comput. Commun. Appl.*, 4 (1), 2008. 2

[122] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 2, 3, 4, 5, 6, 15, 18

[123] Zhensong Zhang, Yongwei Nie, Hanqiu Sun, Qing Zhang, Qiuxia Lai, Guiqing Li, and Mingyu Xiao. Multi-view video synopsis via simultaneous object-shifting and view-switching optimization. *IEEE Transactions on Image Processing*, 29:971–985, 2020. 2

[124] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018. 4

[125] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3537–3545, 2019. 5

Model	*Captioning* CIDEr [108]	METEOR [7]	*Actions and objects* V-IoU	N-IoU	NC-IoU
Ours w/o captioner finetuning in our pseudo-labeler L	0.4	12.2	1.4	6.5	4.8
Ours w/o direction prediction between camera centers in our relative camera pose predictor P	12.9	48.1	32.5	36.8	31.6
Ours	**13.5**	**48.4**	**33.7**	**39.2**	**32.9**

Table 4. Ablation results on the large-scale Ego-Exo4D [37] dataset, in addition to what is provided in 'Ablations' in Sec. 4.2 in main. For the ablation that does not predict the direction between camera centers during relative pose prediction, we predict the exact differences in locations between camera centers instead. Significance, $p \leq 0.05$.

6. Supplementary material

In this supplementary material we provide additional details about:

- Video (with audio) for qualitative illustration of our task and qualitative assessment of our view predictions (Sec. 6.1), as referenced in 'Qualitative examples' in Sec. 4.2 in main
- Additional ablations of our model components (Sec. 6.2), as mentioned in 'Ablations' in Sec. 4.2 in main
- Analysis of the view-specificity of our model's learned visual features (Sec. 6.3), as noted in 'Ablations' in Sec. 4.2 in main
- Analysis of the impact of rank our selector's sampled view on view selection performance (Sec. 6.4), as mentioned in 'Ablations' in Sec. 4.2 in main
- Examples of our view selector's attention heatmaps (Sec. 7), as noted in 'Ablations' in Sec. 4.2 in main
- Analysis of our pseudo-labeler (Sec. 7.1), as referenced in Sec. 4.2 in main
- View selection results on Ego-Exo4D [37] with a single exo camera (Sec. 7.2), as mentioned in Sec. 4.2 in main
- 3-fold evaluation of our view selector on Ego-Exo4D [37], as noted in 'Automatic evaluation' in Sec. 4.2 in main
- Analysis of the relation between our model performance and the distribution of different concepts in the ground-truth train narrations (Sec. 10)
- Our pseudo-labeling cost (Sec. 9)
- Dataset details (Sec. 10.1) in addition to what is provided in Sec. 4.1 in main
- Implementation details (Sec. 10.2), as noted in Sec. 4.1 in main

6.1. Supplementary video

The supplementary video qualitatively depicts our task of view-selection in multi-view instructional videos. Moreover, we qualitatively illustrate our key idea, Language for Weakly Supervising View Selection, show our model's view selection quality at the level of both individual clips and long videos (comprising multiple clips), and compare our predictions with those of two best-performing baselines. Some long videos also have the audio commentary of the participant. Please use headphones to hear the audio correctly. The video is available on `http://vision.cs.utexas.edu/projects/which-view-shows-it-best`.

6.2. Additional ablations

In 'Ablations' in Sec. 4.2 of main, we ablate different model components to understand their contribution to our view selection performance. Here, we provide additional ablations to further analyze our model. Table 4 shows the results. Upon keeping the off-the-shelf captioners [64, 122] frozen when generating our best view pseudo-labels using our pseudo-labeler L (Sec. 3.2 in main), the performance declines drastically, indicating that the generic captions generated by frozen off-the-shelf captioners are not at all suitable for activity understanding in instructional videos. Upon predicting the exact displacement of one camera center relative to another, instead of the rough direction between them, when predicting the inter-view relative poses using our relative camera pose predictor P (Sec. 3.3 in main), we against observe a significant drop in view selection performance. This happens possibly because predicting the exact difference in locations between two camera centers can be intractable in our setting, due to the unknown scale of objects and background.

Model	*Captioning* CIDEr [108]	METEOR [7]	*Actions and objects* V-IoU	N-IoU	NC-IoU
Worst	10.9	45.1	29.2	35.8	30.7
Second best	11.9	46.4	30.9	35.8	30.6
Best (Ours)	**13.5**	**48.4**	**33.7**	**39.2**	**32.9**

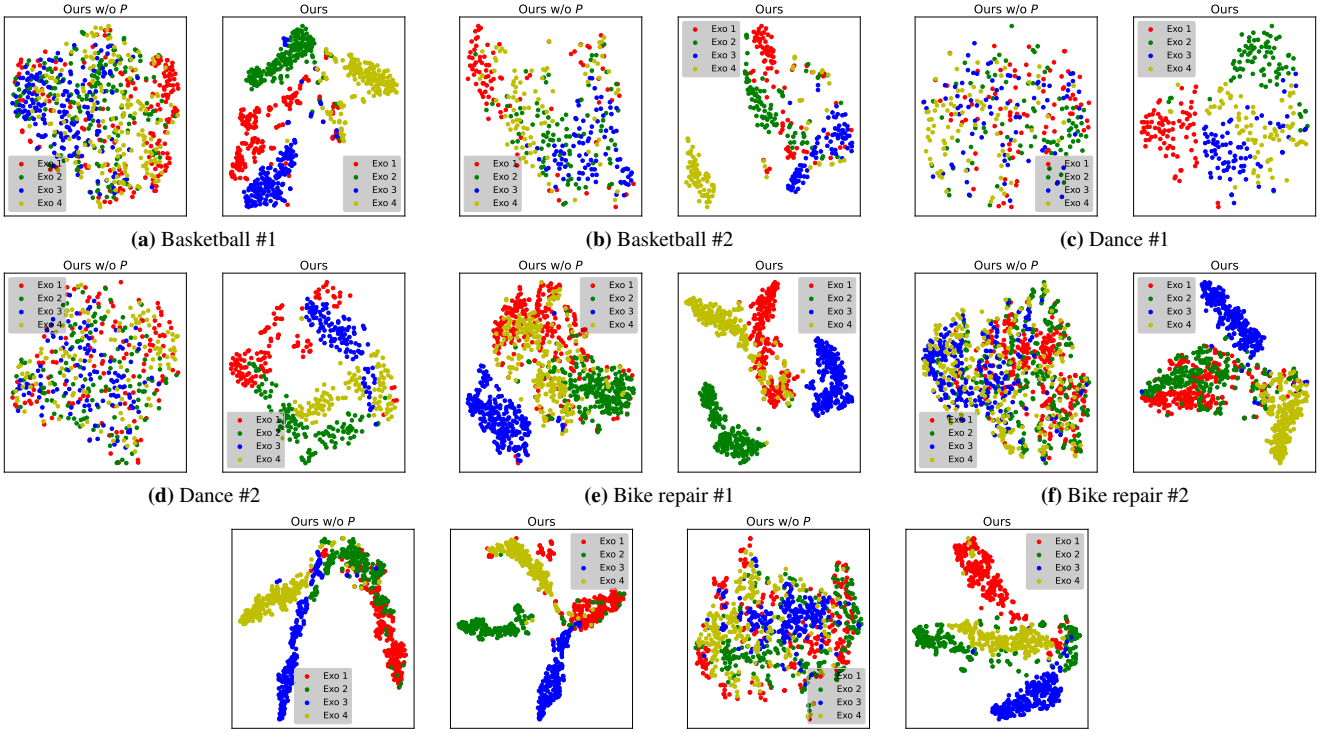
Table 5. Effect of the rank of our sampled view on the view selection performance on Ego-Exo4D [37]. Significance, $p \leq 0.05$.

6.3. View dependence of visual features

Fig. 4 shows the t-SNE visualizations of the visual features corresponding to the exo views of videos from different scenarios–basketball, dance, bike repair and cooking. The scenarios have varying levels of motion of the camera wearer's body and relevant objects–whereas basketball and dance involve moving large and fast movements of the full body and salient objects, bike repair and cooking primarily just involve hands and need less body and object motion. Our learned visual features for the exo cameras when grouped on the basis of the camera ID, produce tighter clusters across samples from different scenarios, compared to the model variant trained without our relative camera pose estimation loss ('View selector training' in Sec. 3.3 in main). This demonstrates that our model's superior ability to learn view-dependent features cuts across different types of activity and different levels of body and object motion, which consequently leads to a stronger view selection performance.

6.4. Sampled view rank

Table 5 shows the impact of the rank of our sampled view on view selection performance. We observe that the lower the rank of our sampled view is, within our model's learned view order, the worse our view selection performance is. This shows that our model's learned ranking of views is highly correlated with the



(a) Basketball #1 **(b)** Basketball #2 **(c)** Dance #1

(d) Dance #2 **(e)** Bike repair #1 **(f)** Bike repair #2

(g) Cooking #1 **(h)** Cooking #2

Figure 4. t-SNE [106] plots of exo visual features of sample Ego-Exo4D [37] videos from basketball, bike repair, dance and cooking scenarios. Our model, when trained with the relative camera pose predictor, produces visual features that form neater clusters when grouped on the basis of different exo views, highlighting their improved view sensitivity.

Ego-Exo4D [37]					*LEMMA [56]*	
Ego	Exo 1	Exo 2	Exo 3	Exo 4	Ego	Exo
20.4	19.8	20.3	19.6	19.9	63.6	36.4

Table 6. Probability distribution in % of our best view pseudo-labels.

view quality, which indicates that our model successfully builds an implicit understanding of which views are more informative.

7. Attention heatmaps of our view selector

In Fig. 5, we provide examples of our model's attention heatmaps on Ego-Exo4D [37]. Our model tends to focus on the salient objects for an activity, even if they are dynamic, indicating its strong activity understanding ability.

7.1. Analysis of our best view pseudo-labeler

Here, we analyze different aspects of our pseudo-labeler L (Sec. 3.2 in main).

In table 6, we report the distribution of our selected views for both Ego-Exo4D [37] and LEMMA [56] datasets. For Ego-Exo4D, our model produces a more or less uniform distribution over all views, indicating that depending on the activity and its level of body and object motion, our model can prefer the ego view or one of the

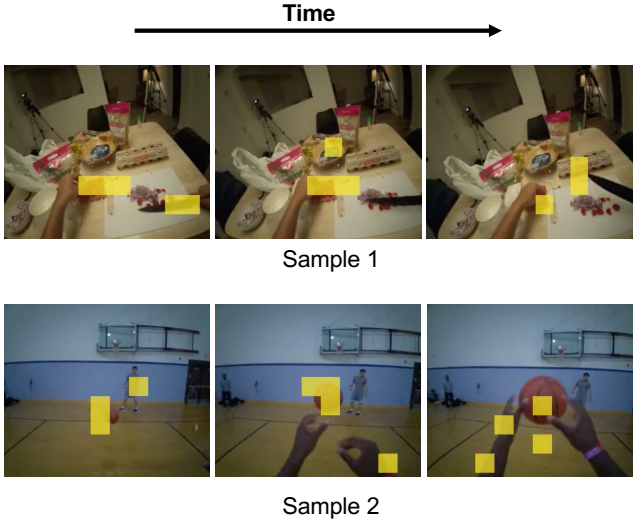


Figure 5. Our model's attention heatmaps on two best view clips from Ego-Exo4D [37]. Yellow patches indicate highest attention.

exo views with almost equal likelihood. However, for LEMMA, our model tends to prefer the ego view much more than the exo view, re-emphasizing the prevalence of household activities that largely require the ego view for capturing their informative aspects

Figure 6. Examples of predicted narrations, and the ranks and scores of the views, per our pseudo-labeler L, shown alongside ground-truth narrations, in addition to what is provided in Sec. 3.2 in main.

Figure 7. Additional examples of best and worst views, and their scores, per our pseudo-labeler L.

('Dataset' in Sec. 4.1 in main).

In addition to the ones provided in Fig. 2b in main, we show more pseudo-labeler outputs, comprising view ranks and predicted narrations, alongside the ground-truth narrations, in Fig. 6. In Fig. 7, we provide more such examples without narrations. We see very similar patterns in these additional samples—the better our pseudo-labeler considers a view to be, the more accurate the narration predicted from the view, is, in terms of capturing important activity

Model	CIDEr	METEOR	V-IoU	N-IoU	NC-IoU
Ours w/ 2 captioners	13.3	**48.4**	**34.2**	38.1	32.5
Ours (w/ 3 captioners)	**13.5**	**48.4**	33.7	**39.2**	**32.9**

Table 7. Impact of captioner count on view selection performance, evaluated with Ego-Exo4D [37]. Significance, $p \leq 0.05$. See row 3 of Table 3, and Sec. 4.2, in main, for results with 1 captioner.

details.

In Table 7, we compare our view selection performance on Ego-Exo4D [37], when using 3 vs. 2 captioners—see row 3 of Table 3, and Sec. 4.2, in main for results with 1 captioner, in our pseudo-labeler (Sec. 3.3 in main). Our view selection performance general improves with the increase in the captioner count in our pseudo-labeler, possibly because having more captioners vote on the best view reduces captioning noise and improves pseudo-label quality.

7.2. Ego-Exo4D with single exo camera

Here, we evaluate our view selector on the single exo camera variant of Ego-Exo4D [36] in order to emulate more typical instructional settings [56, 74] that consist of a single exo camera, but also retain the challenges in the Ego-Exo4D data arising from the diversity in scenarios, varying degrees of body and object motion, etc. Table 8 shows the results, where all metrics are first computed separately for each possible ego-exo view pair and then averaged over all pairs. Our model significantly outperforms all baselines across metrics, showing that it is robust to different camera setups even on challenging datasets with diverse activity scenarios and varying levels of motion of the objects and body parts involved in the activity.

8. 3-fold evaluation on Ego-Exo4D

In Table 9, we report the results from 3-fold evaluation with Ego-Exo4D [37]. Our model significantly outperforms Body-Area, the best baseline. This shows that our model's improvement over the baselines sustains across multiple test datasets.

9. Pseudo-labeling cost

We use 8 NVIDIA V100 GPUs for training and performing inference with the captioners in our pseudo-labeler (Sec. 3.2 in main). When pseudo-labeling Ego-Exo4D [37], it takes ∼2.5 days with VideoLlama captioners, and 3 hours with VideoChat2. For LEMMA [56], the same takes 1 hour per captioner. Importantly, this is a one-time cost since we pseudo-label only once per dataset, and we do not use any captioner when training or evaluating our view selector.

10. Model performance vs. distribution of concepts in ground-truth train narrations

Fig. 8 plots our *test* gains over Body-area [57], the strongest baseline, versus the frequency (most to least) of occurrence of different concepts in the ground-truth *train* narrations. The lack of a strong correlation demonstrates that our view selection is not biased by the dominant concepts in the training narrations.

10.1. Dataset details

Here, we provide additional dataset details. For both Ego-Exo4D [37] and LEMMA [56], we uniformly sample 8 frames from each clip and resize each frame to 224×224. Further, we normalize each pixel in a frame by first dividing it by 255 so that its value lies in $[0, 1]$, then subtracting the pixel mean and finally dividing by the pixel standard deviation, where the pixel mean and standard deviation are channel-specific. We set the mean and standard deviation to $[0.48145466, 0.4578275, 0.40821073]$ and $[0.26862954, 0.26130258, 0.27577711]$, respectively, for our view selector and Video-Llama [122] captioners, and $[0.485, 0.456, 0.406]$ and $[0.229, 0.224, 0.225]$, respectively, for our VideoChat2 [64] captioner, where the channels follow the RGB order.

We split the Ego-Exo4D videos into sequences of clips, each coupled with a narration, by adopting the "contextual variable length clip pairing strategy" strategy [67, 90], which generates temporal windows for extracting clip-narration pairs. To split the LEMMA videos into clips, we group contiguous frames using their verb and noun annotations (Sec. 4.1 in main).

For Ego-Exo4D, we preprocess each narration by denoting each activity participant mentioned in the narration using 'Xi', where i is the participant's position in the sequence in which the participants appear in the time-sorted narrations for each full video (a take in Ego-Exo4D). The value of i starts from 0. We produce narrations for LEMMA by appending the verb and object annotations, where each narration has the following structure: 'verb1: object1_1, object1_2, ...; verb2: object2_1, object2_2, ...; ...' .

10.2. Implementation details

Here, we provide additional implementation details for different components of our framework, and our Pixel-objectness [12, 112] baseline.

10.2.1. Captioner

For our VideoLlama [122] and VideoChat2 [64] captioners, we use a model with the same architecture as proposed in the original paper and initialize the parameters from the checkpoints released by the authors. We freeze the ViT [26] encoder and LLM (without LoRA [47], wherever it is used) in all captioners, and train all other modules with an AdamW [70] optimizer for a maximum of 1.6 million iterations. We use a cosine annealing learning rate schedule [69] with a linear warmup over 5000 iterations, where we set the starting learning rate to 10^{-6}, the peak learning rate to 3×10^{-5}, and the minimum learning rate during cosine annealing to 1×10^{-5}. We set the total batch size to 8, and the (β_1, β_2) and weight decay in AdamW to $(0.9, 0.999)$ and 5×10^{-2}, respectively. Furthermore, for VideoChat2, we turn off flash attention [22, 23]. Finally, we set the LLM prompt to 'What is the person wearing smart glasses doing in the video?' for Ego-Exo4D [37] and 'What is the person wearing a head-mounted camera in the video doing?' for LEMMA [56].

10.2.2. View selector

We use the EgoVLPv2 [86] vision encoder, pretrained on Ego-Exo4D [37], to obtain visual features f in our view selector S (Sec. 3.3 in main). The EgoVLPv2 encoder is a 12-layer TimeSformer [8] model, where we set the prediction head ($head$), predic-

Model	*Captioning*		*Actions and objects*		
	CIDEr [108]	METEOR [7]	V-IoU	N-IoU	NC-IoU
Ego	10.2	45.2	30.2	34.1	29.1
Random	9.8	44.5	29.0	34.9	28.5
Random-exo	9.6	43.8	28.0	34.2	27.4
Hand-object [16]	11.5	46.8	32.2	36.8	30.5
Body-area [57]	10.3	45.4	30.2	34.4	28.4
Joint-count [57]	9.9	44.6	28.6	34.1	28.1
Pixel-objectness [12, 112]	11.2	46.1	30.9	35.9	29.4
Longest-caption	0.0	0.0	0.0	0.0	0.0
Ours	**12.7**	**47.1**	**32.7**	**37.3**	**30.9**

Table 8. View selection with Ego-Exo4D, when the candidate viewpoints comprise the ego view and one exo view. All metrics, expressed in % are averaged over all possible ego-exo view pairs. Significance, $p \leq 0.05$.

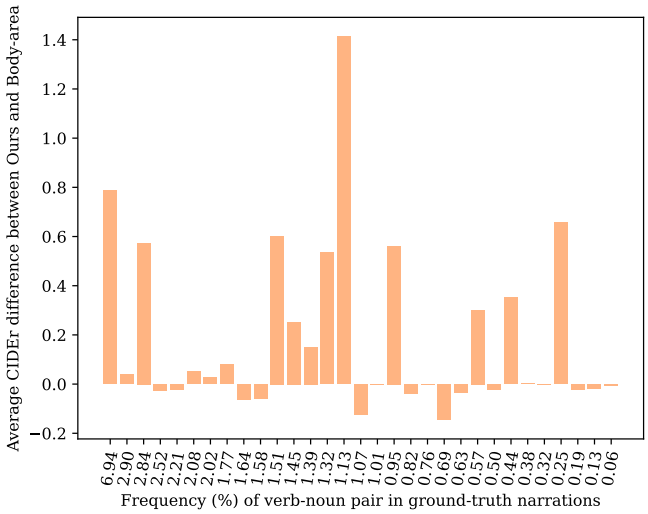


Figure 8. *Test* CIDEr difference between our model and the Body-area [57] baseline vs. verb-noun pair frequency in *train* narrations, sorted in decreasing order

Model	CIDEr	METEOR	V-IoU	N-IoU	NC-IoU
Body-area	10.5	46.6	30.0	35.2	30.4
Ours	**11.4**	**46.9**	**31.2**	**37.0**	**31.9**

Table 9. Average view selection results over three disjoint test splits from Ego-Exo4D [37]. Significance, $p \leq 0.05$.

tion logits (pre_logits) and fully-connected layer (fc) to identity functions from PyTorch. We attach a shared convolution layer to the encoder for producing shared features for both view classification in W (Sec. 4.1 in main) and pose prediction in P (Sec. 4.1 in main). The shared convolution has a kernel size, padding and stride of 1, 768 input channels and 192 output channels. The output of the shared convolution goes into a view selection head and a pose prediction head.

The view selection head begins with the following layers: 1) a Batch Normalization [50] layer with 192 input channels, 2) a ReLU [1] activation, 3) a convolution layer with a kernel size of 4, stride of 2, padding of 1, and 192 and 96 input and output channels, respectively, 4) a Batch Normalization layer with 96 input channels, 5) a ReLU activation, and 6) a convolution layer with a kernel size of 4, stride of 2, padding of 0, and 96 and 24 input and output channels, respectively. We feed the output of the last convolution from above to a a transformer [107] encoder, which comprises 2 layers with 8 heads and 768 channels. Each layer uses a dropout of 0.1 and uses sinusoidal positional encodings [107]. We then feed the output of the transformer encoder to a 2-layer MLP that comprises 1) a linear layer with 768 input channels and 128 output channels, 2) a Batch Normalization layer with 128 input channels, 3) a ReLU activation, 4) a dropout layer with the dropout probability set to 0.1, and 5) a linear layer with 128 input channels and the output channel count set to the number of views.

The pose prediction head comprises a convolution-only and linear-layer-only component. The convolution-only component comprises 1) a Batch Normalization [50] layer with $192 \times 2 = 384$ input channels, 2) a ReLU [1] activation, 3) a dropout layer with the dropout probability set to 0.1, and 4) a convolution layer with a kernel size of 4, stride of 2, padding of 1, and 384 and 48 input and output channels, respectively. The linear-layer-only

component is comprised of 1) a Batch Normalization layer with 2352 input channels, 2) a ReLU activation, 3) a dropout layer with the dropout probability set to 0.1, 3) a linear layer with 2352 input dimensions and 53 output dimensions. We feed the outputs of the convolution-only component to the linear-layer-only component.

We employ $resize$ and $reshape$ operations from PyTorch wherever necessary.

We train our view selector using AdamW [70] with a learning rate of 10^{-5} for the EgoVLPv2 [86] vision encoder and 10^{-4} for the rest of the model. We set the total batch size to 24, and the (β_1, β_2) and weight decay in AdamW to $(0.9, 0.999)$ and 10^{-5}, respectively.

For all our model components, we stop training once the validation loss starts increasing.

10.2.3. Baseline: Snap angles [12, 112]

This baseline ('Baselines' in Sec. 4.1 in main) is an upgrade to the most relevant existing methods [12, 112] in the literature. It predicts the view with the highest count of pixels belonging to foreground [12, 112] and salient [12] objects but not lying near the frame boundaries [112], as the best view. To do so, we treat the set of all objects mentioned in the training narrations as foreground and salient, and query a model composed of GroundingDino [68] and Segment Anything (SAM) [60] with this set to detect its constituent pixels. Specifically, we first feed GroundingDino with the foreground-and-salient object set to compute the corresponding bounding boxes. Next, we feed these bounding boxes to SAM to mark all pixels of relevance. Finally, for each view, we compute a score that is a weighted sum of its average foreground-and-salient pixel count across all frames and a penalty term that lowers the count by the inverse of the view's frame count, for every pixel found within a certain distance from the frame boundaries. We set the weights on the foreground-and-salient pixel count to 1.0, and the penalty term to 0.1 and 0.02 for Ego-Exo4D [37] and LEMMA [56], respectively, through validation, and the distance for using a foreground-and-salient pixel in computing the penalty term, to 6.25% [112] of the frame size.