Spacewalk-18: A Benchmark for Multimodal and Long-form Procedural Video Understanding in Novel Domains

Zitian Tang* Rohan Myer Krishnan* Zhiqiu Yu Chen Sun
Brown University

Abstract

*Learning from (procedural) videos has increasingly served as a pathway for embodied agents to acquire skills from human demonstrations. To do this, video understanding models must be able to obtain structured understandings, such as the temporal segmentation of a demonstration into sequences of actions and skills, and to generalize the understandings to novel environments, tasks, and problem domains. In pursuit of this goal, we introduce Spacewalk-18, a benchmark containing two tasks: (1) step recognition and (2) video question answering, over a dataset of temporally segmented and labeled tasks in International Space Station spacewalk recordings. In tandem, the two tasks quantify a model's ability to: (1) generalize to novel domains; (2) utilize long temporal context and multimodal (*e.g. *visual and speech) information. Our extensive experimental analysis highlights the challenges of Spacewalk-18, but also suggests best practices for domain generalization and long-form understanding. Notably, we discover a promising adaptation via summarization technique that leads to significant performance improvement without model fine-tuning. The Spacewalk-18 benchmark is released at* https://brown-palm.github.io/Spacewalk-18/.

1. Introduction

This is Ground Control to Major Tom
You've really made the grade
And the papers want to know whose shirts you wear
Now it's time to *leave the capsule*, if you dare

Space Oddity

Figure 1. **Key properties of Spacewalk-18**: (1) Domain generalization: a side-by-side comparison of a sample frame from Spacewalk-18 and Ego4D [18] illustrates our benchmark's novel domain. (2) Multimodal: the visual content of the top frame does not align with its audio/speech. Instead, the speech corresponding to the bottom frame describes the top frame. (3) Long-form: the top frame shows the astronaut working on the solar array and the bottom frame shows him releasing bolts. These contextualize each other to identify that he is releasing the solar array in both frames.

Procedural videos, such as how-to or cooking, are produced to spread knowledge for human learners. With the recent advances in video understanding and robotic learning [3, 5, 43], these videos have become increasingly valuable learning resources for robots too. However, in order for machines to efficiently learn complex tasks and work alongside humans, they must be able to distill complex procedures into series of steps with only a few examples, often in a previously unseen environment. This challenge inspires the development of procedural video understanding systems that are able to generalize to previously unseen scenarios, and require minimal human supervision.

We introduce a new benchmark, **Spacewalk-18**, to advance multimodal, long-form, and procedural video understanding in a *novel* domain. Whereas most of the existing procedural video benchmarks are sourced from daily household scenarios [10, 18, 19, 24, 46, 48], the Spacewalk-18 dataset comprises video clips from 18 recorded extravehicular activities (spacewalks) outside the International Space Station. Videos in this domain are naturally limited by the number of recorded spacewalks. As illustrated in Fig. 1, our dataset is first and foremost a benchmark to test pre-trained video-language models' (*e.g*. [2, 55]) capability to *leave the capsule* and generalize to novel domains. Spacewalk-18 is also inherently multimodal and requires effective incorporation of long-form temporal context.

In Spacewalk-18, astronauts go on spacewalks for a variety of reasons, including to perform experiments, test equipment, or carry out maintenance/repairs. The recordings follow a fairly rigid agenda, often illustrated through a short animated sequence giving an overview of the steps planned for the spacewalk. Since our goal is to evaluate a (pre-trained) model's generalization capability, we follow the design choice of recent benchmarks with similar goals (*e.g.*, [31, 36]) and provide a moderate-sized training data for adaptation rather than pre-training. In order to collect detailed and dense temporal annotations for the training, validation, and test videos, we introduce a new protocol for temporal segmentation and action labeling to efficiently annotate the recordings and categorize content from the mission into the corresponding animated steps.

As illustrated in Figure 2, the collected annotations provide a structured representation of each spacewalk recording as a sequence of steps, each of which is in the form of a text description (*e.g.*, "install thermal blanket on degraded antenna"), an animated illustration, and the temporal boundaries within the recording where the step is performed. We define two proxy tasks for evaluation:

1. The *step recognition* task evaluates the model's ability to generalize to the spacewalk domain and incorporate video and text content into predictions.
2. The *question answering* task benchmarks the model's ability to perform spatiotemporal reasoning.

We evaluate state-of-the-art video-language models, including contrastive video-language models (VLMs) [28, 55, 60] and video large language models (VLLMs) [8, 53, 66]. Through our experiments, we identify several best practices for incorporating multimodal information and long-form temporal context. Our human evaluation shows that an average English speaker achieves 67% step recognition performance after viewing 3.5-minute "training" example video, far exceeding any state-of-the-art VLLMs or proprietary APIs (*e.g.*, GPT-4o), suggesting a large room for improvement. Surprisingly, we discover that the most effective adaptation technique is to provide a frozen VLLM a summarized video as its context, and the relative gain is significantly larger compared to directly fine-tuning a model on the training split of Spacewalk-18.

Our contributions are three-fold: First, we propose a new procedural video benchmark on learning structured video representations and reasoning. Second, we collect the Spacewalk-18 dataset with a new protocol for efficient annotation. It contains 96 hours of densely annotated videos and spans over 455 animated steps. Finally, we conduct extensive experimental analyses and discover the adaptation via summarization technique for effective domain generalization in Spacewalk-18.

2. Related Work

Procedural Video Understanding has important applications in video summarization [35], human machine interaction [34], procedural planning [7], and robotic learning [9, 58]. Common tasks defined for procedural video understanding include temporal action segmentation [13, 14] and detection [38], step localization [48, 69] and prediction [1, 17, 18]. They either use dense segment-level annotations or weak video-level labels [32, 39, 45]. Existing procedural video benchmarks are mainly sourced from two domains, how-to videos from online platforms [32, 44, 48, 69], and ego-centric videos collected from recruited actors [10, 18, 42] in kitchens and other household scenarios. More recently, the Perception Test [36] benchmark, aims to evaluate the zero-shot generalization capabilities of video-language foundation models. Our Spacewalk-18 benchmark is complementary to all of the existing procedural video understanding benchmarks: The spacewalk recordings are scripted and narrated in detail, yet cannot be solved by the visual or language modality alone. They exhibit strong structures and dependencies, and unfold over long temporal horizons. More importantly, unlike existing videos that are captured in kitchens and on earth, the space station is a novel domain, which simulates the real-world scenario of model deployment in unseen environments.

Video-language Foundation Models. Inspired by the success with large language models [6, 12], video-language foundation models have been proposed by training with large amounts of image and video data with masked token prediction [16] or video-task matching [67]. The videos are often accompanied by text descriptions such as speech narration [16, 47, 63, 64]. When video and language are encoded separately, a contrastive learning objective can be employed [28, 60] in a similar fashion as its image-based counterpart [37]. The objectives can be combined [62] and the encoders for different modalities can be shared [52]. In addition to joint multimodal pretraining, researchers have demonstrated the effectiveness of adapting visual descriptions into a large language model, such as with gated cross-attention [2], instructional tuning [8, 30, 66], linearly projecting the visual embeddings into the language space [26, 33], or augmenting large language models with video frame captions [53, 65]. Despite their amazing progress, we have shown that the state-of-the-art video-language models cannot generalize to Spacewalk-18 benchmark, whether zero-shot or with fine-tuning.

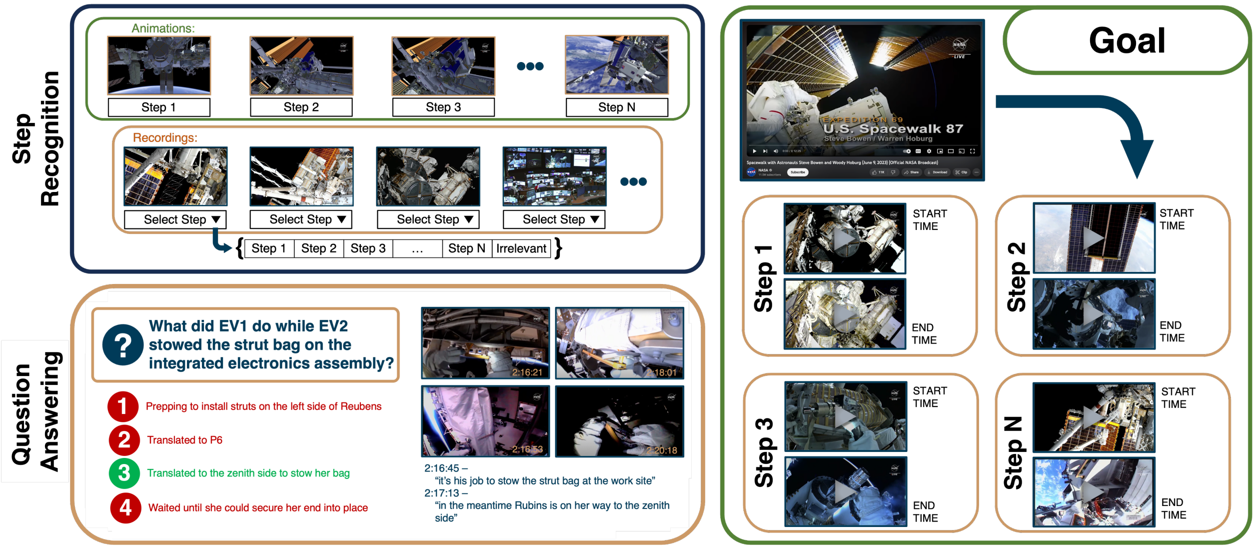


Figure 2. A spacewalk recording can be 7 or 8 hours long. The *step recognition* task aims to assign each video clip in the recording a step label, which is illustrated by a short animation and a text description. The *question answering* task targets video reasoning with long-term multimodal context. Both serve as intermediate benchmarks towards the "Goal", which aims to represent a long procedural video as a sequence of steps and their corresponding video demonstrations for understanding and reasoning.

Long-form Video Understanding is an important open question for video representation learning. Earlier approaches [49] aim to extract high-level information such as character relationships from movies, where the solution is dominated by language-based approaches. Wu *et al*. [56] proposed an object-centric self-supervised framework, along with a benchmark for long-form video understanding. To encode long-form temporal context, memory-based approaches [57] have been proposed, along with new architectures that better scale with longer sequences [22, 27]. Our work is inspired by the recent dataset, EgoSchema [31], for question answering from long-form videos. Spacewalk-18 is complementary to EgoSchema, which is based on the same household videos as in Ego4D [18]. We also compute the temporal certificate [31] to measure the context needed for humans to solve a video understanding task, and observe that our task requires 40% more temporal context than EgoSchema.

3. The Spacewalk-18 Dataset

As shown in Figure 2, our goal is to segment spacewalk recordings into series of steps. A typical spacewalk video contains an animated preview of the steps to be performed, followed by the recording of astronauts performing the steps over multiple hours. We aim to annotate the spacewalk into a sequence of these steps. Each step corresponds to multiple video clips from recording, denoted by start and end times. Each step is accompanied with a text description annotated by a human expert, and illustrated by a video clip from the animated preview. We obtained 18 spacewalk recordings from YouTube. The total duration is 96 hours. For each video, we chunk the audio files into 10 minute clips and feed them to Deepgram to extract speech transcripts.

3.1. Labeling Process

Building our dataset requires temporally segmenting and labeling very long videos. Due to the presence of video clips irrelevant to the spacewalk (*e.g.*, scene of the mission control center in Figure 1), the temporal boundaries of a single step can be fragmented, making annotating the temporal boundaries time-consuming and measuring the inter-annotator agreement challenging. We propose to over-segment the recordings into short clips, each of which corresponds to at most a single step. We then ask the annotators to label each clip from a pre-defined list of steps for a spacewalk, while having access to an unbounded-length context from its neighboring video clips. The human annotators are thus free from having to draw detailed temporal boundaries, or to take multiple passes over the long video for step definition, temporal segmentation, and step labeling. We observe that this design drastically reduced human worker hours.

Define the Label Space. Our label space is derived from the animated preview for each spacewalk. We manually segment the animations into steps ourselves, and label each segment with a step caption. On average, this results in around 25 steps per spacewalk video.

Create Video Clips. We segment each video into short clips using PySceneDetect's shot boundary detection algorithm. We find that due to the long nature of the steps and tendency of the camera to switch angles often in spacewalk recordings, a given shot-segmented clip rarely spans over one step: we randomly inspect the annotated clips and observe that only 6% of them contain more than one step. Overall, only 4% of the frames are mislabeled due to shot boundaries. While imperfect, we believe the benefit of scaling up annotation significantly outweights the minimal impacts on evaluation metrics.

Annotation Interface. To collect annotations, we build the Spacewalk Video Annotation Tool (example interface is shown in Appendix A.1). The interface displays the animated steps and clips for human annotators to label. They watch the annotation clips and select a label for each clip from the set of step labels, "Irrelevant", or "Unsure". This interface allows them to view the entire spacewalk recording as context. The "Irrelevant" label categorizes any clip that does not contain footage of one of the steps for the given spacewalk. This includes shots of the mission control center, noisy shots (*e.g.*, blue screen), and shots of "get-ahead" steps that were not originally planned for the spacewalk. We have three human workers annotate each clip and we choose the most commonly selected label as the true label. Our annotation tool is publicly released.

Merge Adjacent Clips. Since we intentionally over-segment the long spacewalk recordings before collecting annotations, we include a final step to use the collected annotations to obtain true temporal boundaries by merging all adjacent clips with identical labels.

3.2. Dataset Statistics, Diversity, and Difficulty

We collect a total of 102,814 human annotations for 31,812 clips, spanning over 96 hours. We ensure that at least three workers annotate each clip, and on average each clip is annotated by 3.23 workers. Adjacent clips with the same step labels are merged as post-processing, resulting in 3,753 merged clips with an average length of 92 seconds. There are 455 animated step labels across the dataset (excluding "Irrelevant"). Each step takes on average 5 merged clips and has an average of 9 minutes of video. Our dataset has 51 diverse objects (*e.g. battery* and *bracket*) and 47 atomic actions (*e.g. install* and *ingress*) (See Appendix A.3).

Spacewalk-18 is *open-vocabulary*, as the list of step labels varies across different videos. Example labels in the training set include "EV1 & EV2 remove the first battery" and "EV2 ingresses foot restraint". In the test set, there are similar labels, such as "Chris & Bob remove new battery from slot A" and "Chris enters foot restraint".

The visual content in Spacewalk-18 is out-of-domain for most existing models. The shots of astronauts working in space are visually distinct from any worldly actions. There is also a mix of first- and third-person point of view camera angles. Additionally, the two astronauts in each spacewalk often work on different steps in the same time, leading to steps appearing in multiple non-continuous segments. To reduce the impact of task difficulty on annotation accuracy, we offer training to annotators from online platform, and those who achieve at least 80% accuracy on a held-out set after training are recruited to annotate the dataset.

3.3. Task Definition on Spacewalk-18

The 18 recordings in Spacewalk-18 contain varying numbers of steps. To ensure each split contains a balanced number of steps, we manually split the dataset into training, validation, and test sets in a ratio of 10:2:6. We introduce two multimodal long-form video understanding tasks on Spacewalk-18, step recognition and question answering:

Task 1 - Step Recognition. Our dataset divides the multi-hour spacewalk recording into several steps. Each step is described by an animated video clip, the transcript of its narration, and an annotated caption. Our step recognition task aims to recognize these steps from the spacewalk recording. Given a timestamp t and the list of K steps in the corresponding spacewalk mission, a model is tasked with determining the step occurring at timestamp t by predicting a label from $\{0, 1, 2, \cdots, K\}$ (label 0 stands for "Irrelevant"). We notice that temporal contexts are crucial for recognizing the steps, from which we can see the astronauts' actions and know the completed and remaining steps. However, current video-language models are incapable of digesting hours-long spacewalk recordings. So we set a context window length w and offer the video clip $[t - w/2, t + w/2]$ to the model. The clip includes both visual content and textual speech transcripts. We test models with varying w to investigate their ability to understand temporal contexts.

We construct 2000 samples from each training, validation, and test video, resulting in an average of 1 sample for every 10 seconds of video. As the visual content of spacewalk videos does not change rapidly, these samples are sufficient to represent an entire video. To balance the categories, $2000/(K + 1)$ timestamps are uniformly sampled from each step's corresponding video clips. These timestamps are used as the middle timestamps t in the task definition above across different context lengths.

To evaluate model performance, we calculate Accuracy and mean Average Precision (mAP) to measure how accurately each sample is recognized. By merging temporally adjacent samples with the same predictions into intervals, we derive a temporal segmentation of each spacewalk mission. We also adopt the standard intersection-over-union (IoU) metric with the ground truth step boundary annotations to measure the segmentation correctness. We found that sampling more training or validation examples (thus denser in time) has little impact on the IoU metric, which are defined on continuous temporal boundaries. Detailed metric discussions can be found in Appendix B.

Task 2 - Question Answering. For ease of evaluating video-language models on temporal understanding and reasoning with Spacewalk videos, we further introduce the question answering task. We first split each Spacewalk

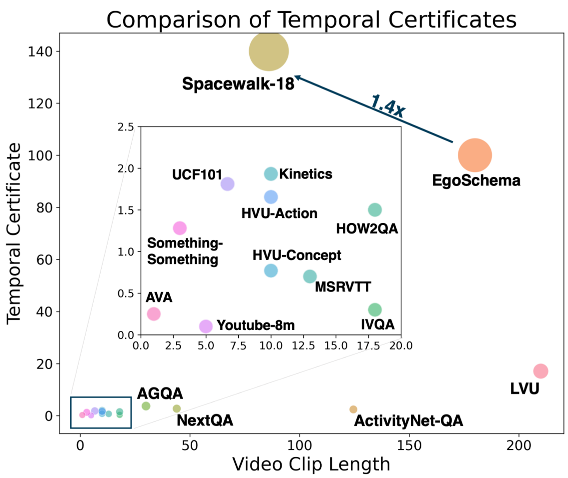


Figure 3. Temporal certificate ("long-form-ness") lengths across commonly adopted datasets with action recognition and question answering annotations. Spacewalk-18 is 1.4x the length of the nearest comparable (EgoSchema). Figure adapted from [31].

video into hour-long consecutive video segments, and then manually collect questions that are either high-level (*e.g. what is the goal of this mission*) or detailed and requires temporal localization and reasoning (*e.g. What type of equipment does the astronaut retrieve first, and how is it utilized during the mission*). Question answering is formulated as multiple-choice, where four candidates are provided. The questions and all possible choices are annotated by experienced annotators. To facilitate the question collection, we leverage the step annotations to automatically generate certain types of questions, such as "*What did EV1 do while EV2 did* `[task]`". Overall, we collected 376 questions for testing a pre-trained VLM's zero-shot performance.

3.4. Characteristics of Spacewalk-18

Our benchmark requires semantic understanding and temporal reasoning abilities in a truly *unique* domain. A comparable benchmark is the recently released Perception Test [36], which is also designed to measure similar skills and generalization capabilities. Spacewalk-18 nicely complements benchmarks like Perception Test as they focus on different domains (daily life vs. space) and has comparable total video durations (74 vs. 96 hours).

Long-form-ness: Spacewalk-18 not only contains long videos, but also requires high amounts of context (long-form video understanding) in order to annotate. We use the temporal certificate [31] metric to quantify the long-form-ness of our video dataset. The temporal certificate measures the amount of video context required for a given dataset/task. Human verifiers are provided with an annotated video clip and are asked to select the amount of the video they require to be confident that the provided label is correct. EgoSchema [31] defines benchmarks with certificate length around 1 second as short video tasks, 10 seconds as long-form, and 100 seconds as very long-form.

Benchmarks	Duration (h)	# Annotations	Domain
Step recog. (task 1)		Temporal Segments	
Ikea-FA [50]	4	2 k	Assembling
Breakfast [24]	77	8 k	Cooking
YouCook2 [68]	176	14 k	Cooking
EPIC-KITCHEN [11]	100	89 k	Cooking
Spacewalk-18 (ours)	96	4 k	Spacewalk
Question ans. (task 2)		Questions	
EgoSchema [31]	250	5 k	Open-domain
Video-MME-L [15]	206	900	Open-domain
P. Test 1hr-walk [20]	10	70	Tours
Spacewalk-18 (ours)	96	376	Spacewalk

Table 1. Statistics comparison with single-domain step recognition benchmarks and recent long-form video QA benchmarks.

To collect this metric, we split each clip into 5-second chunks and task human workers with selecting the minimum subset of these clips that they need to be confident in the given label. In Figure 3, we benchmark our dataset with eight annotators over 2.5 hours of spacewalk video. Our dataset has an average clip length of 89 seconds and a temporal certificate length of 140 seconds. This is 1.4x the length of the nearest dataset and places it in the category of "very long-form video datasets" [31].

Multi-modality: We also observe that the temporal understanding and reasoning tasks in Spacewalk-18 are inherently multimodal. Table 4 shows the human performance when asked to solve the step recognition task, where they can freely explore the temporal context from the entire spacewalk video when needed. We can see that humans perform the best when both video and audio (speech transcipts) are available, with an accuracy of 67.0%. This multimodal performance is higher than when humans only have access to video (52.2%) or audio (39.1%).

Comparison with Relevant Datasets: Table 1 compares Spacewalk-18 with other single-domain step recognition datasets. Our dataset has comparable total duration as relevant datasets, while focusing on a unique domain than cooking or assembling furniture. We also compare with recent long-form video QA benchmarks, which shows that our question size is reasonable as a single-domain benchmark.

4. Recognition and Reasoning Models

For step recognition, we benchmark both contrastive video-language models (contrastive VLMs) and video large language models (VLLMs). Contrastive VLMs can provide aligned video and text embeddings while VLLMs can process video and language simultaneously and perform video question answering. Both of them are evaluated in zero-shot scenario to demonstrate how pre-trained models generalize to Spacewalk-18. We also evaluate contrastive VLMs in fine-tuning scenario, which can effectively adapt pre-trained models to novel domains [40]. Both last-layer and all-layer fine-tuning are considered here.

In Spacewalk-18, the animation of the i-th step is described by an animation video V_i^a, a transcript of its narration T_i^a, and a step caption C_i^a. A spacewalk recording clip centered at timestamp t with length w contains a video clip $V_{t,w}$ and a transcript $T_{t,w}$. When evaluating contrastive VLMs, we format recognition as retrieval, similar to how contrastive VLMs can be used for zero-shot classification. We employ separate video encoder $F_v(\cdot)$ and text encoder $F_t(\cdot)$ to derive clip features, and match the spacewalk recording clips with step animations. To evaluate VLLMs, we feed the videos and transcripts into the models and treat our tasks as multi-choice video question answering. Unless otherwise mentioned, all the models use Sparse Frame Sampling to process videos, where we uniformly sample k frames from the entire clip $V_{t,w}$ regardless of the video length and feed them into the video encoder. k is the number of frames used during model pre-training, which may differ between different models. For the question answering task, a model is given the entire hour-long input video, and it needs to search for the relevant evidence from the long-form input, making the task more challenging.

4.1. Step Recognition and QA by VLLMs

Zero-shot. We format the step recognition task as multi-choice video question answering as the following. Given a spacewalk clip with K steps in the mission, we provide the video $V_{t,w}$ and transcript $T_{t,w}$ to the model and ask "*Which step does the frame in the middle of this video belong to?*" We also provide the caption C_i^a and transcript T_i^a of each step and require the model to choose a step index between 0 and K. See our prompt in Appendix C.2. We follow the same strategy for the question answering task, with the only difference that the choice candidates are provided by each question. We focus on the zero-shot setup and report performance not only on state-of-the-art VLLMs, but also proprietary APIs, such as GPT-4o.

4.2. Step Recognition by Contrastive VLMs

Zero-shot. To derive a feature for a step animation, we first extract the video, transcript, and caption features respectively, and then concatenate them. Formally, the feature of the i-th step is $f_i^a = [F_v(V_i^a),\ F_t(T_i^a),\ F_t(C_i^a)]$. To construct a feature for the "Irrelevant" category, we write a textual description of it and extract its text feature. The description is *DES="The mission control center, noisy shots (e.g. blue screen), or tasks not planned for the spacewalk."* Hence the "Irrelevant" category feature is $f_0^a = [F_t(\text{DES}),\ F_t(\text{DES}),\ F_t(\text{DES})]$. For a recording clip, we concatenate the video and transcript features: $f^s = [F_v(V_{t,w}),\ F_t(T_{t,w}),\ F_t(T_{t,w})]$. Here, the transcript feature is repeated so that the feature dimensionality is the same as those of the animation features.

To form a prediction, we compute the similarities $f^s \cdot f_i^a$ between a recording clip and an animation step, and pick the step with the highest similarity.

Last-layer Fine-tuning. In this setting, we fine-tune a linear layer upon the pre-trained models using the training set. Specifically, we freeze the models and train a linear layer $G_\theta(\cdot)$ mapping the spacewalk clip features to the animiation features. We minimize the cross entropy loss during training, which is

$$\mathcal{L} = -\log \frac{\exp\left(G_\theta(f^s) \cdot f_y^a\right)}{\sum_{1 \le i \le K} \exp\left(G_\theta(f^s) \cdot f_i^a\right)}. \tag{1}$$

After training, instead of constructing or learning a feature for the "Irrelevant" category, we find a threshold τ and recognize all clips whose similarities to task steps are all below τ as "Irrelevant". This is more reasonable than an "Irrelevant" feature because this category contains a subspace formed by various concepts rather than a single concept. With the threshold τ, we make predictions by

$$\hat{y} = \begin{cases} \arg\max_{1 \le i \le K} G_\theta(f^s) \cdot f_i^a, & \max G_\theta(f^s) \cdot f_i^a \ge \tau, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

We find the τ with the best F1 score on the the validation set and use it in the test phase. In practice, this method performs better than constructing an "Irrelevant" feature through text descriptions.

All-layer Fine-tuning. We fine-tune the entire backbone of a pre-trained model to encode spacewalk video clips. We freeze the animation features and use the same cross entropy loss to fine-tune the model. The threshold τ for the "Irrelevant" category is also set using the validation set.

Incorporating Longer Temporal Context. Our Sparse Frame Sampling approach naturally incorporates long temporal context with the same computational cost, but may lose important details due to sampling. We therefore explore several alternatives to incorporate temporal context: **Dense Frame Sampling** at 1 FPS, which are directly encoded by the video encoder to obtain a single video embedding. This approach is bounded by GPU memory since the number of frames scales linearly with the context duration. **Long-term Feature Bank (LFB)** [57], which first divides the long context into seconds-long segments and leverages a frozen video encoder to extract one video embedding for each segment with a frame sampling rate of 1 FPS.

We explore the following LFB variants following [57]: Average pooling over the query and all of the context features to form a single embedding (**LFB Avg**); Average pooling the history context and future context separately, and concatenate them together with the query embedding (**LFB Cat**); Learning to aggregate temporal context via non-local blocks [54] (**LFB NL**) or a two-layer Transformer encoder (**LFB TF**). More details can be found in Appendix C.5.

Method	w = 1 min			w = 3 min			w = 5 min		
	Acc.	mAP	IoU	Acc.	mAP	IoU	Acc.	mAP	IoU
Random	4.22	-	1.51	4.22	-	1.51	4.22	-	1.51
Human*	-	-	-	-	-	-	67.0	-	-
Zero-shot									
EgoVLP	7.18	8.28	1.90	7.71	9.08	1.67	7.59	9.83	1.59
VideoCLIP	8.00	10.24	2.12	6.58	**11.11**	2.65	7.85	10.68	2.38
InternVideo	9.35	**10.78**	2.96	9.48	11.01	2.73	9.07	**11.56**	2.97
LLaVA-Next-Video	10.35	-	3.77	13.71	-	5.07	13.82	-	4.92
VideoLLaMA2	9.34	-	2.88	14.37	-	5.45	17.32	-	6.28
Caption-enhanced LLM	**18.56**	-	**9.33**	**26.32**	-	**12.89**	**28.49**	-	**13.16**
GPT-4o	15.47	-	6.36	21.65	-	9.41	26.40	-	11.55
Last-layer Fine-tuning									
EgoVLP	6.34	8.92	2.17	9.68	10.66	3.03	**10.21**	10.86	3.18
VideoCLIP	8.40	9.80	3.21	9.98	11.14	3.71	8.87	10.39	3.46
InternVideo	**10.12**	**12.17**	**4.02**	**11.13**	**12.68**	**4.04**	10.08	**12.53**	**4.04**
All-layer Fine-tuning									
InternVideo	13.21	11.77	4.60	13.34	12.93	4.54	12.93	13.09	4.63

Table 2. Model performances on Spacewalk-18 step recognition task. Caption-enhanced LLM performs the best among all the models in zero-shot setting. Both last-layer and all-layer fine-tuning improves the performances of contrastive VLMs. $*$: Humans have access to unlimited context.

Method	#Frames	Accuracy (%)
Random	-	25.00
LLaVA-Next-Video-7B	32	25.53
LLaVA-Next-Video-34B	32	29.52
VideoLLaMA2-7B	8	27.13
VideoLLaMA2-7B	16	31.65
Caption-enhanced LLM	60	30.37
GPT-4o	8	31.65
GPT-4o	16	30.32
GPT-4o	32	32.45
GPT-4o	60	31.91

Table 3. Results on Spacewalk-18-QA.

5. Experiments

Our experiments show that current video-language models perform significantly worse than humans on Spacewalk-18. We also demonstrate the importance of both vision and language on our tasks. Furthermore, the capability of LFB in incorporating temporal contexts is verified.

5.1. Evaluated Models

As described in Section 4, we evaluate contrastive VLMs and VLLMs in our experiments. For contrastive VLMs, we test **EgoVLP** [28], **VideoCLIP** [60], and **InternVideo** [55]. For VLLMs, we test open-source **LLaVA-Next-Video** [66] and **VideoLLaMA2** [8], and proprietary **GPT-4o**. Moreover, as recent works [53, 65] solve VideoQA tasks effectively by feeding generated video frame captions into LLMs, we also develop a **caption-enhanced LLM** with `LLaVA-1.5-13B` [29] as the video frame captioner and GPT-4o as the LLM reasoner. See the selected checkpoint and number of sampled frames in Appendix C.1.

5.2. Human Performance Evaluation

On the step recognition task, we allow human performers to access to unlimited video context, and compute human performance as the average accuracy:

$$\text{Accuracy} = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|J_i|}\sum_{j\in J_i}\mathbb{1}_{a_{ij}=t_j}, \tag{3}$$

where n is the total number of annotators, J_i is the set of clip indices annotated by worker i, a_{ij} is worker i's annotation for clip j, and t_j is the true label for clip j. It can be viewed as the "upperbound" performance on our task.

5.3. Main Results

Table 2 shows the model performances on the step recognition task. Due to the high computational cost of all-layer fine-tuning, it is only employed on InternVideo. We report the results under a few context lengths and conduct a thorough exploration of it in Section 5.5. The best accuracy on this task is achieved by caption-enhanced LLM, which is only 28.49%, far less than the human performance of 67%. This verifies that our task is reasonable for humans but challenging to VLMs. VLLMs generally outperforms contrastive VLMs, and significantly when the context window is long. Among the contrastive VLMs, InternVideo performs the best. Both last-layer and all-layer fine-tuning can boost constrastive VLMs, and all-layer fine-tuning yields higher performance than last-layer fine-tuning. However, the improvements are marginal, indicating that adapting models to extremely rare domains remains challenging.

Table 3 shows the model performances on the question answering task. Regardless the number of frames we use, none of the method, including GPT-4o, is able to outperform the random baseline by a significant margin. For both tasks, we provide **qualitative results** showcasing the success and failure scenarios in Appendix D.4.

5.4. Effect of Modality

In Table 4, we ablate the input modality on the step recognition task using zero-shot video-language models. And we also measure the human performances in multimodal and unimodal cases. In most cases, human and open-source video-language models perform the best when both modalities are provided, demonstrating the inherently multimodal nature of our task. However, GPT-4o and caption-enhaced LLM (which uses GPT-4o API) achieve the highest accuracy when only text is given, probably due to the language prior of LLMs. Besides, we notice that humans can better use the visual inputs, while our models benefit more from the texts. We hypothesize that humans excel in extracting abstract concepts from videos and match them with other modalities, while it is difficult for video-language models. Moreover, while the overall performance of InternVideo is worse than VLLMs, it is the best video-only model, demon-

Method	Accuracy					
	w = 1 min			w = 5 min		
	V	T	V+T	V	T	V+T
Human*	-	-	-	52.2	39.1	67.0
InternVideo	**7.84**	9.15	9.35	6.26	9.68	9.07
LLaVA-Next-Video	5.22	9.38	10.35	4.93	14.65	13.82
VideoLLaMA2	4.39	8.03	9.34	4.56	13.03	17.33
Caption-enhanced LLM	5.48	**18.92**	**18.56**	5.13	**31.03**	**28.49**
GPT-4o	6.86	**18.92**	15.48	**6.68**	**31.03**	26.4

Table 4. Ablation about input modality on step recognition task. The models are evaluated in zero-shot scenario. V: video; T: text (captions and transcripts). *: Human has unlimited context and accesses transcripts in the form of audio.

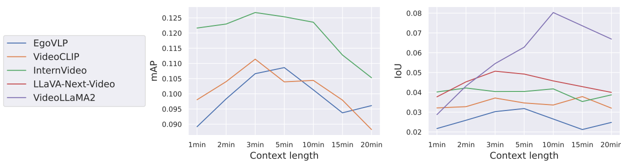


Figure 4. Ablation on context length. We test the models under various context lengths. Contrastive VLMs are last-layer fine-tuned while MLLMs are zero-shot. When the temporal context is extremely long, the models can no longer benefit from it.

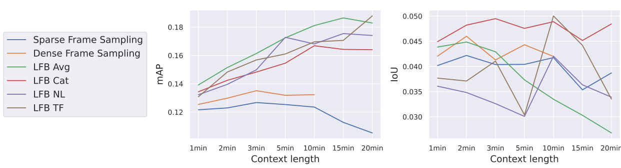


Figure 5. Performances of different temporal context incorporation methods built upon frozen InternVideo features. While LFB methods yield increasing mAP when the temporal context extends, one-time feed-forward models with either sparse or dense frame sampling cannot benefit from the context.

strating its better visual generalization to rare domains.

5.5. Leveraging Temporal Context

As shown in Figure 4, unlike humans that have temporal certificate of 2.3 minutes, the open-source video-language models cannot benefit from very long context on both tasks (proprietary ones are not tested due to API cost). As the context window length increases, their performances initially improve, but start to decline after reaching their peaks.

To address this issue, we explore the temporal context incorporation approaches for contrastive VLMs introduced in Section 4.2 on the step recognition task. As Table 4 shows that contrastive VLMs have the most robust visual capability to rare domains, it is more effective to explore video context incorporation using them than VLLMs. The performance curves with respect to context lengths are shown in Figure 5. The two one-time feed-forward methods – Sparse/Dense Frame Sampling – have similar performances, and they are not improved given expanded temporal context windows as well. This indicates that naively sampling more frames cannot help pre-trained video-language models to understand long-form videos. In contrast, the mAP curves of all the LFB methods show significant upward trends, verifying their ability to incorporate temporal contexts in long videos. Among them, LFB Cat, which concatenates past, present, and future video features, outperforms other methods on the accuracy and IoU metrics, demonstrating its effectiveness.

Task	Method	Summary	Accuracy (%)
Step Recog.	InternVideo	-	9.48
	InternVideo (fine-tuned)	-	13.34
	Caption-enhanced LLM	-	23.50
	Caption-enhanced LLM	Step Oracle	28.49
QA	GPT-4o	-	32.45
	GPT-4o	Animation	55.59
	GPT-4o	Step Oracle	81.12

Table 5. Adaptation via summarization significantly improves Spacewalk-18 performance on both tasks.

5.6. Adaptation via Summarization

We finally investigate how to effectively adapt a pre-trained model to solve Spacewalk-18. Our key inspiration is that for long-form, novel-domain videos, a model may adapt to and understand the videos by watching their summarizations.

In the step recognition task, the full list of steps is available to the VLLMs. This list in fact serves as an oracle summarizing the overall spacewalk mission, which might contribute to their better performance than contrastive VLMs. To ablate its influence, we give the caption-enhanced LLM only one step each time and ask it to rate each step from 1 to 10. In Table 5, the step oracle indeed improves its performance from 23.50% to 28.49%, which is an even higher gain than fine-tuning the InternVideo.

For the QA task, we simulate the summarization by either re-purposing the 5-minute animation video of each spacewalk, or by assuming a known list of steps that occur within the question video. These steps are represented by step captions and middle frames. In Table 5, the animation and step oracle provide significant improvements over GPT-4o. While both animation and step oracle should be considered as privileged information, they highlight the direction to adapt pre-trained models via *condensed* knowledge.

6. Conclusion

We introduce the Spacewalk-18 benchmark to evaluate video-language models' capability to generalize to unseen domains, and their comprehension of multimodal information and long-term temporal context. We demonstrate that while average human annotators achieve competitive performance on our benchmark, existing video-language models still struggle with domain generalization and long-form video understanding. Meanwhile, we discover that a promising direction for model adaptation is to provide video summary as its context, even without model fine-tuning.

Acknowledgements: This work is supported by NASA, Samsung Advanced Institute of Technology, and a Richard B. Salomon award for Chen Sun. We thank Karttikeya Mangalam and Raiymbek Akshulakov for their kind help with EgoSchema temporal certificate evaluation, and Professor Stefanie Tellex for her inspiration on looking into Spacewalk videos. Our research was conducted using computational resources at the Center for Computation and Visualization at Brown University.

References

[1] Yazan Abu Farha, Alexander Richard, and Juergen Gall. When will you do what?-anticipating temporal occurrences of activities. In *CVPR*, 2018. 2

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022. 2

[3] Shikhar Bahl, Russell Mendonca, Lili Chen, Unnat Jain, and Deepak Pathak. Affordances from human videos as a versatile representation for robotics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13778–13790, 2023. 1

[4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 8

[5] Homanga Bharadhwaj, Abhinav Gupta, Shubham Tulsiani, and Vikash Kumar. Zero-shot robot manipulation from passive human videos. *arXiv preprint arXiv:2302.02011*, 2023. 1

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020. 2

[7] Chien-Yi Chang, De-An Huang, Danfei Xu, Ehsan Adeli, Li Fei-Fei, and Juan Carlos Niebles. Procedure planning in instructional videos. In *European Conference on Computer Vision*, pages 334–350. Springer, 2020. 2

[8] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, and Lidong Bing. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024. 2, 7, 8

[9] Yilun Dai, Mengjiao Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *arXiv preprint arXiv:2302.00111*, 2023. 2

[10] 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 *ECCV*, 2018. 1, 2

[11] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision*, pages 1–23, 2022. 5

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2

[13] Guodong Ding, Fadime Sener, and Angela Yao. Temporal action segmentation: An analysis of modern techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2

[14] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Counting out time: Class agnostic video repetition counting in the wild. In *CVPR*, 2020. 2, 12

[15] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024. 5

[16] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. VIOLET: End-to-End Video-Language Transformers with Masked Visual-token Modeling. In *arXiv:2111.1268*, 2021. 2

[17] Rohit Girdhar and Kristen Grauman. Anticipative video transformer. In *ICCV*, 2021. 2

[18] 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 *CVPR*, 2022. 1, 2, 3

[19] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 1

[20] Joseph Heyward, João Carreira, Dima Damen, Andrew Zisserman, and Viorica Pătrăucean. Perception test 2024: Challenge summary and a novel hour-long videoqa benchmark. *arXiv preprint arXiv:2411.19941*, 2024. 5

[21] Frank Hutter Ilya Loshchilov. Decoupled weight decay regularization. In *ICLR*, 2019. 20

[22] Md Mohaiminul Islam and Gedas Bertasius. Long movie clip classification with state-space video models. In *ECCV*, 2022. 3

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 9

[24] H. Kuehne, A. B. Arslan, and T. Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *CVPR*, 2014. 1, 5

[25] Sateesh Kumar, Sanjay Haresh, Awais Ahmed, Andrey Konin, M Zeeshan Zia, and Quoc-Huy Tran. Unsupervised action segmentation by joint representation learning and online clustering. In *CVPR*, 2022. 12

[26] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with

frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2

[27] Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Mvitv2: Improved multiscale vision transformers for classification and detection. In *CVPR*, 2022. 3

[28] Kevin Qinghong Lin, Alex Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Zhongcong Xu, Difei Gao, Rongcheng Tu, Wenzhe Zhao, Weijie Kong, et al. Egocentric video-language pretraining. *arXiv preprint arXiv:2206.01670*, 2022. 2, 7, 8

[29] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023. 7, 8

[30] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 2

[31] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *arXiv preprint arXiv:2308.09126*, 2023. 2, 3, 5, 6

[32] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019. 2, 8

[33] Seungwhan Moon, Andrea Madotto, Zhaojiang Lin, Tushar Nagarajan, Matt Smith, Shashank Jain, Chun-Fu Yeh, Prakash Murugesan, Peyman Heidari, Yue Liu, et al. Anymal: An efficient and scalable any-modality augmented language model. *arXiv preprint arXiv:2309.16058*, 2023. 2

[34] Tushar Nagarajan, Yanghao Li, Christoph Feichtenhofer, and Kristen Grauman. Ego-topo: Environment affordances from egocentric video. In *CVPR*, 2020. 2

[35] 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 *ECCV*, 2022. 2

[36] Viorica Pătrăucean, Lucas Smaira, Ankush Gupta, Adrià Recasens Continente, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Joseph Heyward, Mateusz Malinowski, Yi Yang, et al. Perception test: A diagnostic benchmark for multimodal video models. *arXiv preprint arXiv:2305.13786*, 2023. 2, 5

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2

[38] Alexander Richard and Juergen Gall. Temporal action detection using a statistical language model. In *CVPR*, 2016. 2

[39] Alexander Richard, Hilde Kuehne, and Juergen Gall. Weakly supervised action learning with rnn based fine-to-coarse modeling. In *CVPR*, 2017. 2

[40] Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. Domain-adjusted regression or: Erm may already learn features sufficient for out-of-distribution generalization. *arXiv preprint arXiv:2202.06856*, 2022. 5

[41] Saquib Sarfraz, Naila Murray, Vivek Sharma, Ali Diba, Luc Van Gool, and Rainer Stiefelhagen. Temporally-weighted hierarchical clustering for unsupervised action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11225–11234, 2021. 20

[42] Fadime Sener, Dibyadip Chatterjee, Daniel Shelepov, Kun He, Dipika Singhania, Robert Wang, and Angela Yao. Assembly101: A large-scale multi-view video dataset for understanding procedural activities. In *CVPR*, 2022. 2

[43] Sumedh A Sontakke, Jesse Zhang, Sébastien MR Arnold, Karl Pertsch, Erdem Bıyık, Dorsa Sadigh, Chelsea Finn, and Laurent Itti. Roboclip: one demonstration is enough to learn robot policies. *arXiv preprint arXiv:2310.07899*, 2023. 1

[44] Tomáš Souček, Jean-Baptiste Alayrac, Antoine Miech, Ivan Laptev, and Josef Sivic. Look for the change: Learning object states and state-modifying actions from untrimmed web videos. In *CVPR*, 2022. 2

[45] Tomáš Souček, Jean-Baptiste Alayrac, Antoine Miech, Ivan Laptev, and Josef Sivic. Multi-task learning of object state changes from uncurated videos. *arXiv preprint arXiv:2211.13500*, 2022. 2

[46] Sebastian Stein and Stephen J. McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *UBICOMP*, 2013. 1

[47] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019. 2

[48] 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 *CVPR*, 2019. 1, 2

[49] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question-answering. In *CVPR*, 2016. 3

[50] Sam Toyer, Anoop Cherian, Tengda Han, and Stephen Gould. Human pose forecasting via deep markov models. In *2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8. IEEE, 2017. 5

[51] Carl Vondrick, Deva Ramanan, and Donald Patterson. Efficiently scaling up video annotation with crowdsourced marketplaces. In *ECCV*, 2010. 1

[52] Alex Jinpeng Wang, Yixiao Ge, Rui Yan, Ge Yuying, Xudong Lin, Guanyu Cai, Jianping Wu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. All in one: Exploring unified video-language pre-training. In *CVPR*, 2023. 2

[53] Shijie Wang, Qi Zhao, Minh Quan Do, Nakul Agarwal, Kwonjoon Lee, and Chen Sun. Vamos: Versatile action models for video understanding, 2023. 2, 7

[54] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 6, 10

[55] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, Sen Xing, Guo Chen, Junting Pan, Jiashuo Yu, Yali Wang, Limin Wang, and Yu Qiao. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022. 2, 7, 8, 9

[56] Chao-Yuan Wu and Philipp Krahenbuhl. Towards long-form video understanding. In *CVPR*, 2021. 3

[57] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *CVPR*, 2019. 3, 6, 9

[58] Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. Daydreamer: World models for physical robot learning. In *CoRL*, 2023. 2

[59] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *ECCV*, 2018. 8

[60] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *EMNLP*, 2021. 2, 7, 8

[61] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. CVPR, 2016. 8

[62] 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. 2

[63] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. Merlot: Multimodal neural script knowledge models. In *NeurIPS*, 2021. 2

[64] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. Merlot reserve: Neural script knowledge through vision and language and sound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16375–16387, 2022. 2

[65] Ce Zhang, Taixi Lu, Md Mohaiminul Islam, Ziyang Wang, Shoubin Yu, Mohit Bansal, and Gedas Bertasius. A simple llm framework for long-range video question-answering, 2023. 2, 7, 8

[66] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, 2024. 2, 7, 8

[67] Honglu Zhou, Roberto Martín-Martín, Mubbasir Kapadia, Silvio Savarese, and Juan Carlos Niebles. Procedure-aware pretraining for instructional video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10727–10738, 2023. 2

[68] 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. 5

[69] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *CVPR*, 2019. 2

Spacewalk-18: A Benchmark for Multimodal and Long-form Procedural Video Understanding in Novel Domains

Supplementary Material

We first elaborate the construction and statistics of Spacewalk-18 dataset in Appendix A. Appendix B describes the details of the evalutions metrics on the step recognition task. In Appendix C, we introduce the details of our evaluated models, VLLM prompts, fine-tuning strategies, and variants of long-term feature bank. Appendix D provides more experiment results, such as the influence of training data size and qualitative results. Finally, we explore an additional task in Appendix E to investigate the models' ability to recognize the spacewalk mission steps without pre-defined step lists.

A. Dataset Construction

Sourcing Live Streams. First, a list of spacewalks at the International Space Station is sourced from NASA's website. For each spacewalk from 2019 to 2023, we identify whether there is a recording available on YouTube with an animation sequence near the beginning. We find a total of 18 spacewalk recordings that meet this criteria. Table A1 has a list of the NASA summary for each expedition and the duration of each YouTube video.

Building Transcripts. We use Deepgram's Automatic Speech Recognition (ASR) service to transcribe the spacewalk videos. In order to choose an ASR algorithm, we had a human verifier analyze transcriptions from Deepgram, OpenAI's Whisper, NVIDIA's NeMo, YouTube's Auto Generated Captions, and Google's Speech-to-Text on the same spacewalk audio clip. Deepgram proved to be the most accurate. Through a similar process, we found that Deepgram's ASR algorithm performs the best with approximately 10 minute audio clips. Thus, we chunk the multi-hour audio files into 10 minute clips and feed them to Deepgram for transcription. This results in a list of sentences with start and end timestamps for each spacewalk video.

A.1. Labeling Process

Building our dataset requires temporally segmenting and labeling very long videos (many hours). To do this, we introduce a new annotation protocol and tool. Existing methods of collecting temporal segment annotations require multiple passes and/or for clips to be pre-labeled [19, 51]. While annotating our dataset would traditionally require three passes (action identification, temporal segmentation, and task labeling), we pre-segment the videos into clips containing a maximum of one step each, allowing us to collect annotations in a single pass. This drastically reduces the number of human worker hours required. The source code for the tool will be publicly released. The protocol is as follows:

Expedition Date/URL	Video Duration
Training set	
October 6, 2019	8:42:06
November 22, 2019	8:58:45
January 15, 2020	9:03:04
January 25, 2020	7:59:43
January 27, 2021	8:25:35
February 1, 2021	8:39:50
June 16, 2021	9:23:13
December 2, 2021	7:46:53
March 23, 2022	8:29:54
December 3, 2022	7:44:35
Validation set	
March 15, 2022	9:14:30
November 15, 2022	10:38:40
Test set	
November 15, 2019	8:25:23
December 2, 2019	7:49:37
June 26, 2020	8:14:13
February 28, 2021	9:28:07
September 12, 2021	8:44:29
June 9, 2023	8:12:25

Table A1. The Spacewalk recordings used in the Spacewalk-18 dataset, and their video durations. The recordings cover the spacewalk missions from 2019 to 2023.

Define the Label Space. In our case, the label space comes from the animation videos. We manually segment the animations into clips containing a single step and label each clip with a short description. We show the step lists of three spacewalk missions in Table A2, A3, and A4, one from each of the training, validation, and test sets.

Split the Videos. To reduce the burden of temporal segmentation, we split the long live streams into sub-clips that each contain at most one step. We find that due to the long nature of the steps and tendency of the camera to switch angles often in spacewalk live streams, a given shot-segmented clip (between camera angle changes) will contain at most one action/step. Thus, to split the multi-hour long spacewalk video into clips for the human workers to annotate, we employ a shot detection algorithm. PySceneDetect's ContentDetector uses changes in color and intensity between individual frames to draw boundaries between shot changes in videos. We find that it errs on the side of over-segmenting

Step ID	Caption
1	EV1 and EV2 exit airlock
2	EV1 heads outward and places safety tether anchors
3	EV2 retrieves foot restraint
4	EV1 goes to carrier with solar array
5	EV1 and EV2 drop of PGT and bags
6	EV2 stows foot restraint and bag
7	EV1 preps iROSA for removal
8	EV2 installs bag and tools on mod kit
9	EV2 sets up cables for future installation
10	EV2 retrieves PGT and goes to EV1
11	EV1 does more prep for iROSA removal
12	EV1 retrieves foot restraint from CETA cart
13	EV1 installs, sets up, and enters foot restraint
14	EV1 removes bolts on iROSA
15	Robotic arm moves EV1 into position
16	EV2 gets into position for iROSA release
17	EV2 prepares iROSA for release
18	EV1 removes iROSA
19	EV2 stows tools and enters foot restraint
20	EV1 carries iROSA on robotic arm to EV2
21	EV1 hands iROSA to EV2
22	EV1 exits foot restraint and goes to EV2
23	EV2 rotates into position
24	EV1 enters foot restraint and receives iROSA
25	EV2 exits foot restraint and gets into position
26	EV1 and EV2 install iROSA
27	EV1 and EV2 swing iROSA into single tube
28	EV1 and EV2 drive mounting bolts
29	EV1 and EV2 clean up and prep for next EVA
30	EV1 and EV2 return to airlock
31	EV1 and EV2 enter airlock

Table A2. Step list of the spacewalk mission on June 16, 2021, which is in the training set.

Step ID	Caption
1	Luca and Drew exit airlock with pump system
2	Luca and Drew take pump system to external support platform 2
3	Luca enter foot restraint on robotic arm
4	Drew hands Luca the pump system
5	Robotic arm takes Luca to AMS
6	Drew move to ELC 2
7	Luca and Drew install pump system
8	Luca and Drew connect power and data cables
9	Robotic arm takes Luca to aft side
10	Luca connect six fluid connections
11	Robotic arm takes Luca to underside of AMS
12	Luca and Drew complete final 2 suages
13	Robotic arm takes Luca to ESP 2
14	Drew bring bags back to airlock

Table A3. Step list of the spacewalk mission on December 2, 2019, which is in the test set.

Step ID	Caption
1	EV1 exit airlock and receive large bag
2	EV2 exit airlock
3	EV1 move to integrated equipment assembly
4	EV1 stow bag and begin prep work
5	EV2 move to phase 1
6	EV2 stow crew bag and retrieve tools
7	EV1 and EV2 assemble upper triangle
8	EV2 move to and enter foot restraint
9	EV1 hand upper triangle to EV2
10	EV2 dock upper triangle on gimbal assembly
11	EV1 stow pistol grip tool (PGT)
12	EV2 exit foot restraint and tilt it to the left side
13	EV1 pass left mid strut to EV2
14	EV1 hands lower strut to EV2
15	EV1 and EV2 install lower strut
16	EV1 and EV2 install mid strut
17	EV2 exit foot restraint and tilt it to the right side
18	EV1 hand right mid strut to EV2
19	EV1 and EV2 install lower strut
20	EV1 and EV2 install mid strut
21	EV1 finish mid strut install
22	EV2 move to bag and stow tools
23	EV2 return to worksite and stow tools on body restraint tether
24	EV1 take pictures of completed mod kit
25	EV1 move to battery charge/discharge unit and begin prep work
26	EV2 translate to CETA cart and stow tools
27	EV2 retreive crew lock bag and move to EV1
28	EV1 and EV2 fold and restrain insulation
29	EV1 and EV2 break torque and reinstall
30	EV1 and EV2 clean up and retrieve crew bag
31	EV2 return to airlock
32	EV1 return to airlock

Table A4. Step list of the spacewalk mission on March 15, 2022, which is in the validation set.

for our spacewalk live stream videos, further ensuring that clips will contain at most one step.

Chunk into Segments. It is unreasonable to expect human workers to annotate a multi-hour video in one sitting. Therefore, we chunk the videos into segments of approximately one hour. We design these segments to contain about one hour of recording content and only contain steps from a continuous subset of the animation. Each video has 5.11 segments on average.

Annotation Interface. To collect annotations, we build the Spacewalk Video Annotation Tool, pictured in Figure A1. The human worker inputs their unique user ID and selects the video date and segment number that they are assigned. The interface then loads the corresponding pre-labeled animation clips and the recording clips for them to label. The platform saves their progress as they annotate clips so they

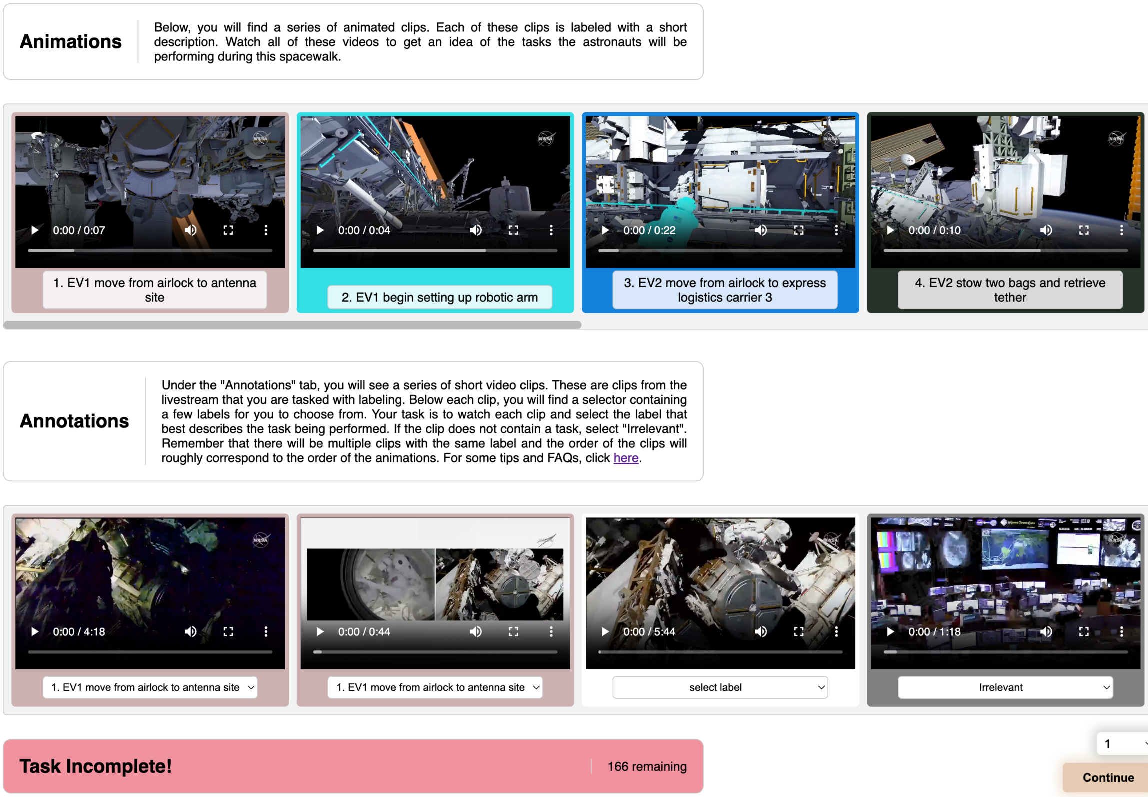


Figure A1. Spacewalk Video Annotation Tool Interface. Row (1) contains a pre-labeled set of steps from the animation video. Row (2) contains a set of live stream clips for the annotator to categorize into steps.

are able to start the task and return to it at any time. Once they complete the task, a completion URL appears.

Source Human Annotators. We use an online platform to source human workers to annotate the dataset. First, participants are sent the training and screening phases where they learn about the task and demonstrate proficiency in being able to accurately label a small sample of spacewalk live stream clips. In the training phase, as workers select labels for live stream clips, they are given feedback about the correctness of their annotations along with some reasoning. They spend as much time as necessary in the training phase before moving on to the screening phase. In this phase, workers are tasked with labeling a set of 10 clips without feedback, and those who achieve an accuracy of 80% or greater are selected for the annotation phase.

Annotate Clips. In the annotation phase, human workers are presented with animation and annotation clips from a single segment of a spacewalk recording (Figure A1). They watch the annotation clips and select a label for each clip from the set of animation clip labels, "Irrelevant", or "Unsure". The "Irrelevant" label is used to categorize any clip that does not contain footage of one of the tasks for the given spacewalk. This includes shots of the mission control

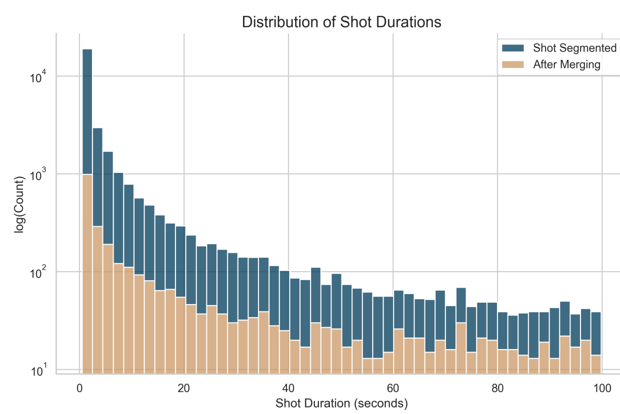
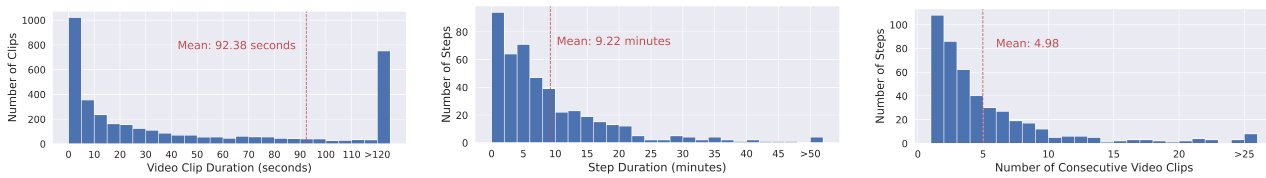


Figure A2. Distributions of video segment durations before and after merging adjacent clips.

center, noisy shots (*e.g.* blue screen), and shots of get-ahead tasks that were not originally planned for the spacewalk. We have three human workers annotate each clip and we choose the most commonly selected label as the true label.

Merge Adjacent Clips. We intentionally over-segment the long spacewalk recordings before collecting annotations, to



(a) Merged Video Clip Duration. (b) Total Step Duration. (c) Video Clip Number per Step.

Figure A3. Distributions of (a) Merged video clip duration, (b) Total step duration, and (c) Video clip number per step.

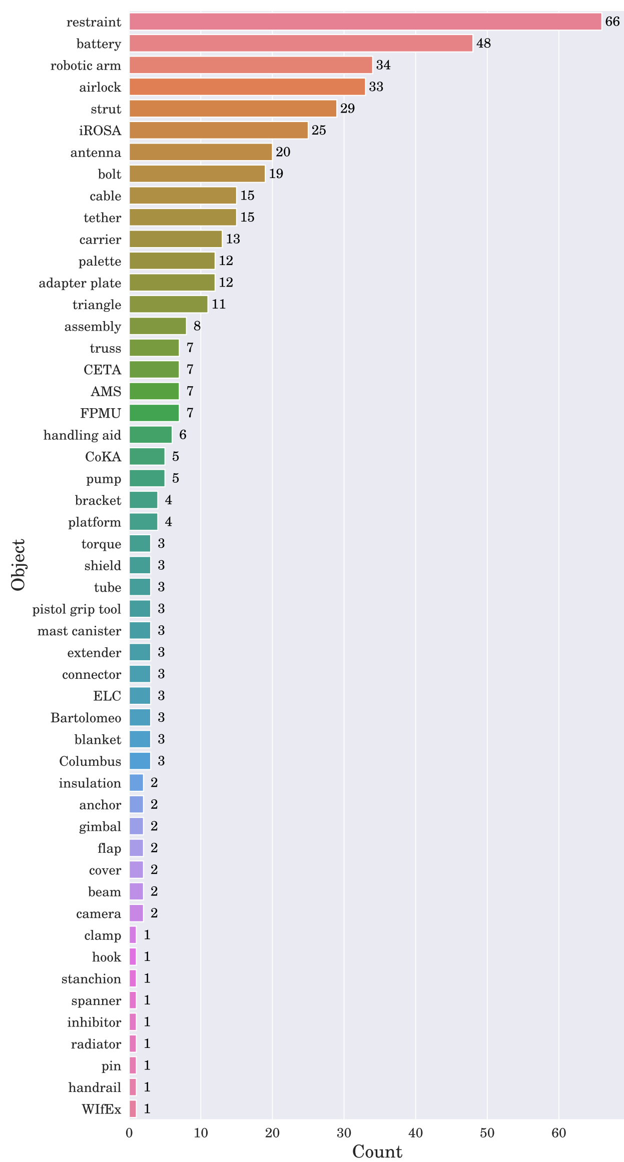


Figure A4. Distribution of objects. It counts how many steps contains each object in their captions.

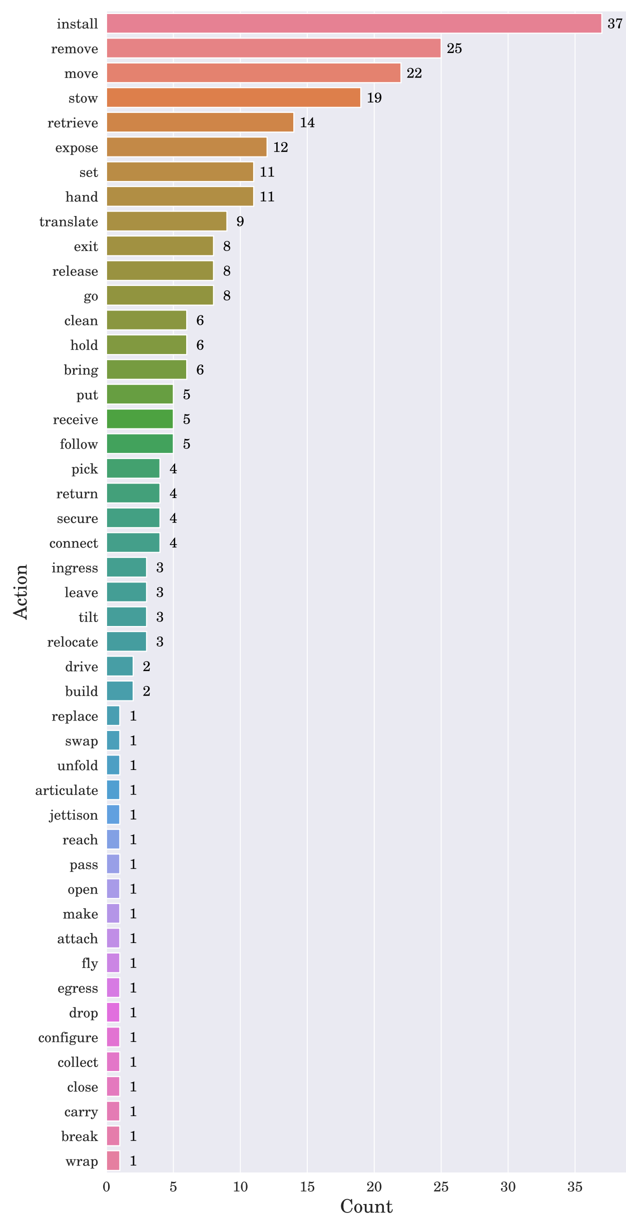


Figure A5. Distribution of actions. It counts how many steps contains each action in their captions.

avoid the same clip spanning across multiple steps. We thus include a final step to merge adjacent clips with the same label. Rather than the traditional method of having human workers provide temporal boundaries, the over-segmentation allows us to break down the challenging tasks of temporal segmentation and action recognition into a series of easier, smaller ones. We then use the collected annotations to obtain true temporal boundaries by concatenating all adjacent clips with identical labels. Figure A2 illustrates how this balances the distribution of clip durations in the dataset.

A.2. Illustrations of Annotated Data

Figure A7, A8, and A9 illustrate three examples of annotations collected from human annotators.

A.3. Statistics

After merging adjacent clips with the same labels, we obtain in total 3,753 clips with annotated spacewalk steps. We show several distributions about the clips in Figure A3, including the durations of the video clips, total durations of the steps, and numbers of clips per step. On average, each merged clip has a length of 92 seconds, each step spans 9 minutes, and each step is composed of 5 clips.

While each spacewalk video has a list of 25 steps on average, a small portion of the steps do not necessarily occur following their order in the list. After manual examination, we find that around 84% of the adjacent steps are logically non-interchangable in order. For example, "installing the battery" must happen after "taking the battery to the worksite". The remaining interchangable steps are mostly due to the parallel work of the astronauts. For example, if "EV 2 retrieves foot restraint" happens while "EV 1 goes to the carrier", there is no guarantee about which step will occur first.

To analyze the diversity of the dataset, we filter the nouns and verbs from the step captions to obtain a list of objects and actions. Figure A4 and Figure A5 illustrate the occurrence of each object and action in the step captions. In total, we observe 51 objects and 47 actions across the dataset.

A.4. Question Types of Question Answering Task

As described in Section 3.3, our question answering task includes 376 questions. 349 of them are generated by templates using our step annotations, while the other 27 miscellaneous questions are manually created. Although the manually created questions are fewer, they are more video-specific than those generated by templates. For the template-generated questions, we further categorize them into four types: task before/at/after location, task before/after task, when task, and task order. Examples of the different question types are listed in Table A6. The distribution of the question types can be found in Figure A6.

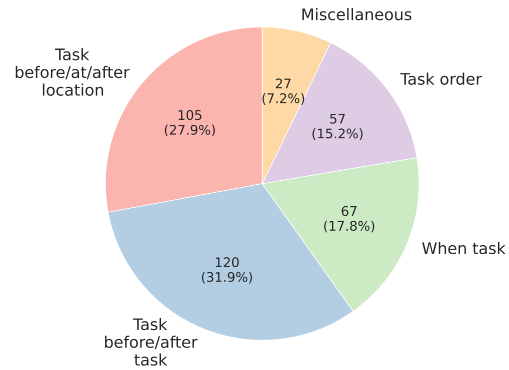


Figure A6. Distribution of the question types of the question answering task.

A.5. Temporal Certificate

Table A5 provides the numerical values we use to plot the temporal certificate figure in Figure 3. It is extended from [31] to include Spacewalk-18.

B. Evaluation Metrics of Step Recognition

We use accuracy, mAP, and IoU to evaluate a model's performance on step recognition. To eliminate the impact of uneven numbers of task steps in each spacewalk video, we first compute the following metrics per video and then average them across videos.

Accuracy. It is the percentage of timestamps whose corresponding steps are correctly recognized.

Mean Average Precision (mAP). For methods that can give a confidence score for each task step, we employ mAP as an evaluation metric. We first compute the average precision of each task step except "Irrelevant" and then take the mean.

Intersection over Union (IoU). After the models predict the step label for each sampled timestamp, we merge adjacent timestamps with the same predictions into consecutive temporal intervals. They form a segmentation of each step across the entire spacewalk video. For each step, we calculate the IoU with the ground truth segmentation of the step. Finally, we take the average over all the steps as the IoU measurement.

Dataset	Temporal Certificate	Average Clip Length	Test Duration
Kinetics	1.931	10	240000
AVA	0.25	1	117900
HVU Concept	0.77	10	650000
HVU Action	1.65375	10	650000
UCF 101	1.81	6.66	9457
Something Something	1.28	3	81471
LVU Relationship	17.12	210	5364
NextQA	2.7	44	47872
EgoSchema	100	180	90000
Youtube8M Segment	0.1	5	2520000000
MSRVTT	0.7	13	38870
IVQA	0.3	18	36000
AGQA	3.7	30	57600
How2QA	1.5	18	16263
ActivityNetQA	2.4	124.58	144000
Spacewalk-18	140	89	119664

Table A5. The numerical values of the temporal certificates. We extend this plot from [31] to include Spacewalk-18. The units for all three columns are seconds. "Test Duration" is the total duration of each test set.

Question Type	Example
Task before/at/after location	Q: Which of the following tasks happens before the astronaut arrives at the external pallet? (A) Chris sets up tools and prepares worksite. (B) Bob moves to IEA. (C) Chris & Bob retrieve battery from slot 1. (D) Chris sets up tether.
Task before/after task	Q: Which of the following tasks happens after EV1 move back inboard? (A) EV1 retreive portable foot restraint with extension. (B) EV1 & EV2 move to P6. (C) EV1 & EV2 install respective bags on worksites. (D) EV2 move to P1 and install anchor hooks for safety tether.
When task	Q: In which part of the video does the task that EV1 & EV2 install mid strut happen? (A) The first third of the video. (B) The middle third of the video. (C) The task does not happen in the video. (D) The last third of the video.
Task order	Q: In which order do the tasks happen in the video? (A) (1) Robotic arm takes Luca to aft side. (2) Drew move to ELC 2. (3) Robotic arm takes Luca to AMS. (B) (1) Drew move to ELC 2. (2) Robotic arm takes Luca to aft side. (3) Robotic arm takes Luca to AMS. (C) (1) Robotic arm takes Luca to AMS. (2) Drew move to ELC 2. (3) Robotic arm takes Luca to aft side. (D) (1) Drew move to ELC 2. (2) Robotic arm takes Luca to AMS. (3) Robotic arm takes Luca to aft side.
Miscellaneous	Q: What type of equipment does the astronaut retrieve first, and how is it utilized during the mission? (A) Bags containing structure to assemble modification kit. (B) Articulating portable foot restraint to later attach to the Canada arm. (C) Pump to be installed on the Alpha Magnetic Spectrometer. (D) Bags containing structure to support new solar arrays.

Table A6. Question types and examples in the question answering task.

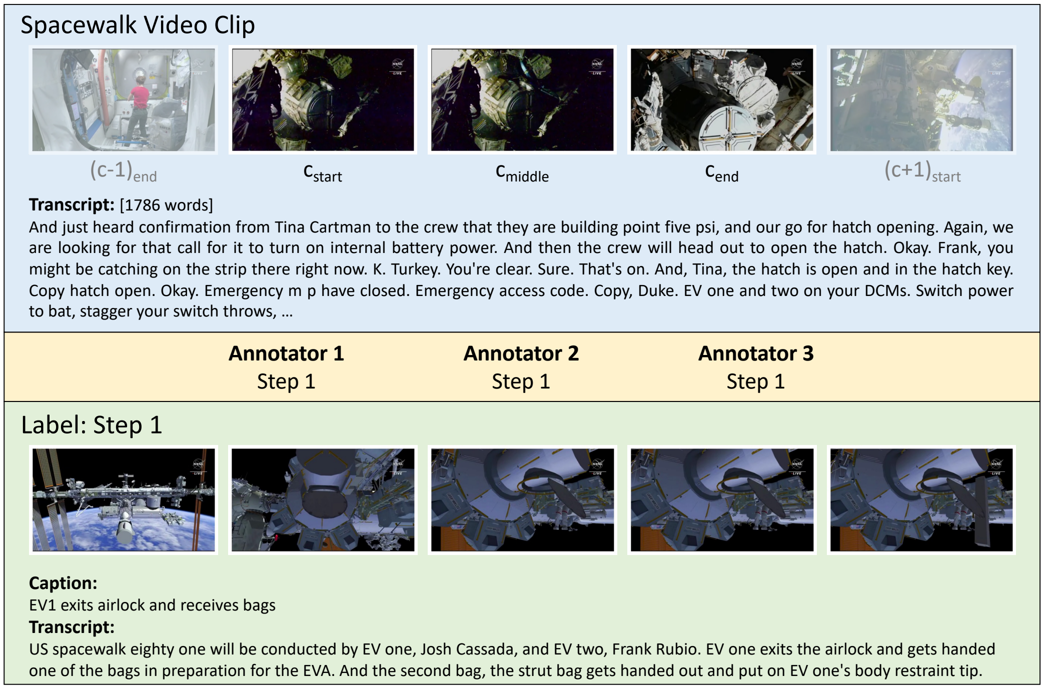


Figure A7. Annotation example 1 from the spacewalk recording from Nov. 15, 2022 (training set). All three annotators agree on the label for the video clip.

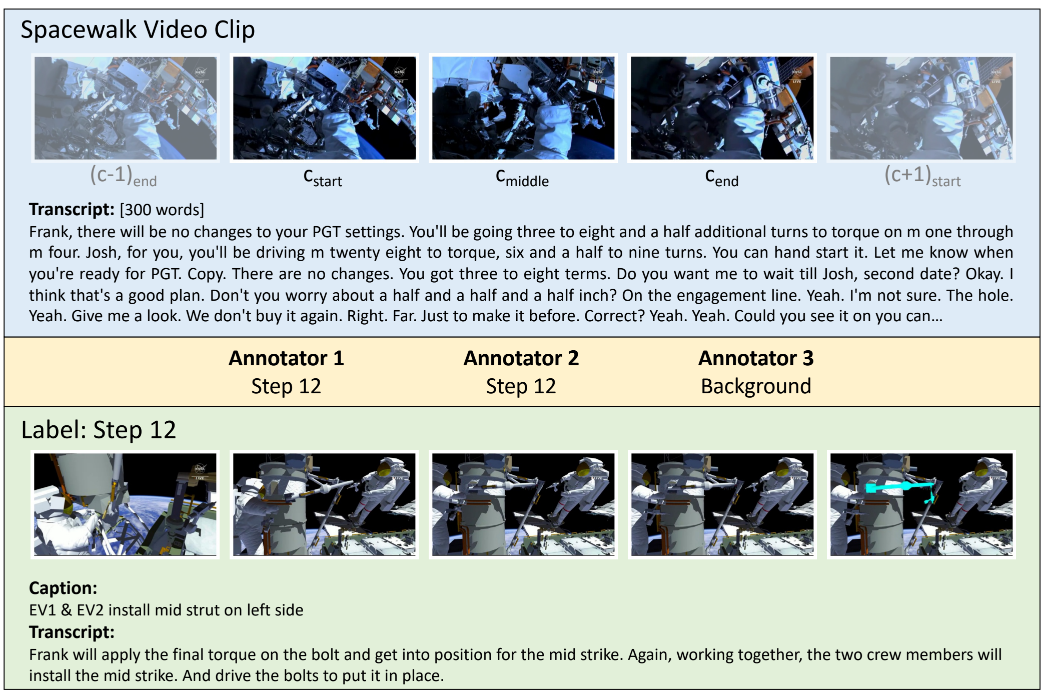


Figure A8. Annotation example 2 from the spacewalk recording from Nov. 15, 2022 (training set). One annotator disagrees with the other two but since the majority of annotators selected step 12, the clip is labeled as step 12. The clips on either side appear to be of the same step, which illustrates the effect of over-segmenting the spacewalk recordings and the necessity of merging adjacent clips after collecting annotations.

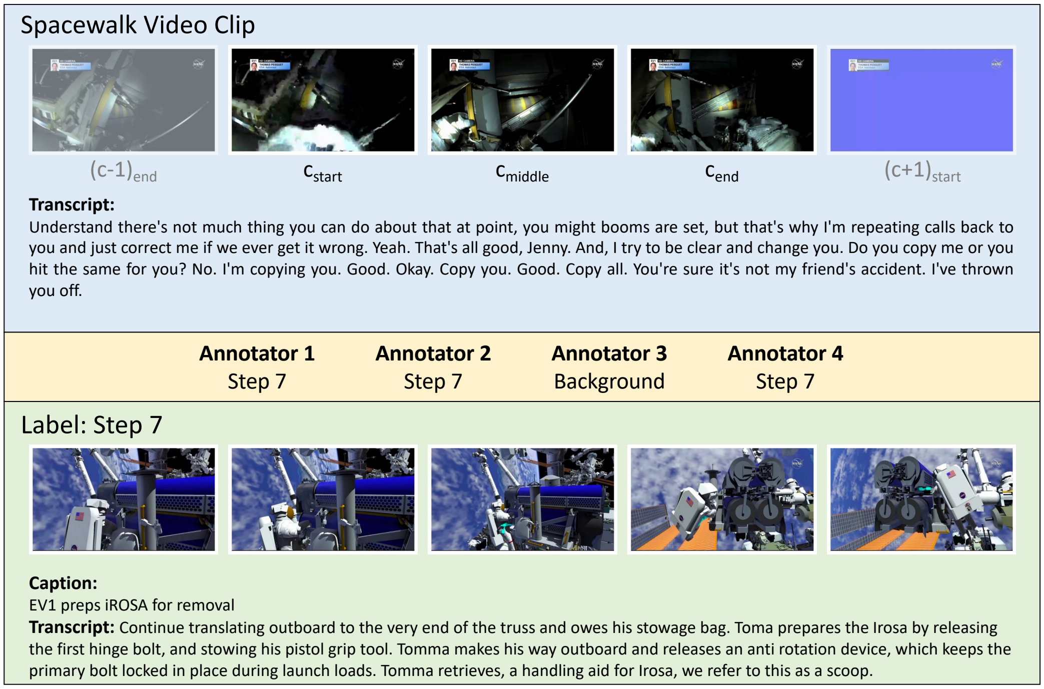


Figure A9. Annotation example 1 from the spacewalk recording from Jun. 16, 2021 (validation set). One annotator disagrees with the other three but since the majority of annotators selected step 7, the clip is labeled as step 7. The clip immediately following this one demonstrates an example of a blue screen that would be classified as "Background".

C. Implementation Details

C.1. Evaluated Models

EgoVLP [28] is an egocentric video-language model trained on EgoClip dataset. In our step recognition experiments, we sample 4 frames from a video for feature extraction.

VideoCLIP [60] is a video-language model trained with video-text contrastive learning. It uses a Transformer to integrate S3D [59] video features and align it with text feature. 150 frames are sampled from a video to extract feature in the step recognition experiments.

InternVideo [55] is a video-language model pre-trained on a large corpus of video-language datasets including HowTo100M [32] and WebVid10M [4]. It achieves state-of-the-art performances across 39 video datasets from extensive tasks. In our step recognition experiments, we use the checkpoint further fine-tuned for video-text retrieval on MSRVTT [61]. 12 frames are sampled from each video clip in the experiments.

LLaVA-Next-Video [66] is a video large language model trained on a large video/image-language corpus. We use the checkpoint `LLaVA-NeXT-Video-34B-DPO` and uniformly sample 32 frames from each video as visual inputs on the step recognition task.

VideoLLaMA2 [8] is a multimodal large language model capable of understanding video, audio, and language. It employs Spatial-Temporal Convolution connector to capture the intricate spatial and temporal dynamics in the video input. We use the `VideoLLaMA2-7B` checkpoint and sample 8 frames from each video on the step recognition task.

GPT-4o is a proprietary multimodal large language model API. As it can receive multiple images as input but not a video file, we uniformly sample a few frames from a video and feed them to the API following the temporal order. The used checkpoint on both of our tasks is `gpt-4o-2024-05-13`. The number of sampled frames on the step recognition task is 8.

Caption-augmented LLM. Following [65], on both of our two tasks, we use `LLaVA-1.5-13B` [29] to caption 60 uniformly sampled video frames with prompt "Describe the image in 30 words". The generated captions are delivered into GPT-4o (`gpt-4o-2024-05-13`) to answer the given question.

C.2. VLLM Prompts

To evaluate VLLMs on the step recognition task, we format the task into multi-choice video question answering. Given the video content and transcript,we ask the model to choose a step index from a given list. Specifically, the VLLM prompt for the step recognition task is as following:

```
<video>
You are given a spacewalk video, where the spacewalk
mission can be divided into <number of steps> steps.
The transcript of the video speech is:
<transcript>.
Please provide a single-number answer (from 0 to
<number of steps>) to the following multiple-choice
question, and your answer must be one of the numbers
from 0 to <number of steps>. You must not provide
any other response or explanation. If you are not
sure, answer with the most likely answer.
Here is the question: Which step does the frame in
the middle of this video belong to?
Here are the choices:
(0) Irrelevant: The mission control center, noisy
shots (e.g. blue screen), or tasks not planned for
the spacewalk.
(1) <step 1 caption>: <step 1 transcript>.
(2) <step 2 caption>: <step 2 transcript>.
... ...
(N) <step N caption>: <step N transcript>.
```

For our task 2, the question answering task, we use the following prompt:

```
<video>
Please provide a single-letter answer (from A to D)
to the following multiple-choice question, and your
answer must be one of the letters from A to D. You
must not provide any other response or explanation.
If you are not sure, answer with the most likely
answer.
Here is the question: <question>
Here are the choices:
(A) <option_1>
(B) <option_2>
(C) <option_3>
(D) <option_4>
```

On both tasks, we set the all the VLLMs' temperature to be 0 to disable sampling during the answer generation.

C.3. Last-layer Fine-tuning Contrastive VLMs

When last-layer fine-tuning the contrastive VLMs, we use Adam [23] optimizer with a learning rate of $1e-4$ and a batch size of 2048. All models are trained for 20 epochs and the checkpoint with the best validation loss is picked.

C.4. All-layer Fine-tuning Contrastive VLMs

We follow the code base of InternVideo [55] to fine-tune its entire model backbone on our step recognition task. We use Adam [23] optimizer with a learning rate of $4e-6$. The learning rate warms up linearly in the first 10% training steps, after which cosine annealing is adapted. Besides, a weight decay of 0.2 is used. We fine-tune the model for 1 epoch with a batch size of 16. We find that training for more than 1 epoch always leads to over-fitting.

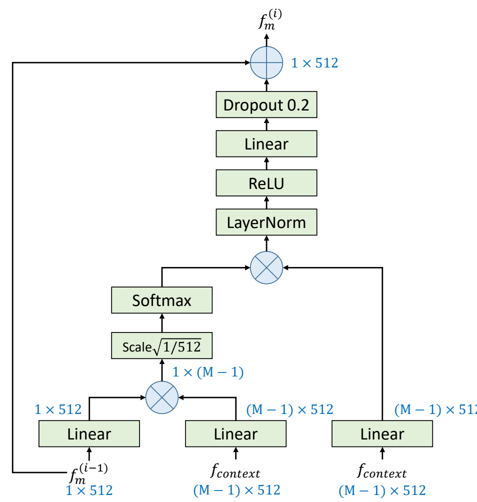


Figure A10. Architecture of a non-local block Non-local$(f_m^{(i-1)}, f_{\text{context}})$ borrowed from [57]. It employs attention mechanism between the query feature $f_m^{(i-1)}$ and context feature f_{context}, and produces an updated query feature $f_m^{(i)}$. When stacking multiple non-local blocks, the query feature is iteratively updated while the context feature remains unchanged.

C.5. Long-term Feature Bank

To employ Long-term Feature Bank (LFB) [57] to solve the step recognition task, we first divide the spacewalk recording clip $V_{t,w}$ into k-second-long segments and extracts their InternVideo features with a frame sampling rate of 1 FPS, respectively. Since InternVideo is pre-trained with 12 frames per video, we set $k = 12$. There are in total $M = 60 \times w/12$ segments for a w-minute-long video and the query timestamp falls in segment $m = M/2$. Denote the feature of the i-th segment as f'_i. LFB aims to incorporate the history feature $f'_H = f'_{1:m-1}$ and the future feature $f'_F = f'_{m+1:M}$ into the query timestamp feature f'_m. Based off [57], we design four context incorporation mechanisms – **LFB Avg**, **LFB Cat**, **LFB NL**, and **LFB TF**. Each of these methods integrate f'_m, f'_H, and f'_F into a feature f for the entire video clip $V_{t,w}$. We substitute f for f_s in Equation (1) to train the linear layer $G_\theta(\cdot)$ and the parameters of NL/TF blocks jointly. The pre-trained video-language encoders are always frozen in this process. Unless otherwise specified, the training hyperparameters are the same as those of last-layer fine-tuning in Appendix C.3.

LFB Avg. We pool the query, history, and future features together to form a single feature, *i.e.*,

$$f = \text{Average}([f'_H, f'_m, f'_F]) = \text{Average}(f'_{1:M}). \quad (4)$$

LFB Cat. We first pool the history and future features, respectively, and then concatenate them with the query feature, *i.e.*,

$$f = [\text{Average}(f'_H), f'_m, \text{Average}(f'_F)]\,. \tag{5}$$

LFB NL. We stack two non-local blocks [54] to learn the interaction between the query feature and the context features via attention mechanism. The architecture of a non-local block is shown in Figure A10. Before feeding the features into non-local blocks, we first add positional encoding to them and use a linear layer to project them into a 512-dimensional space. The output of the non-local blocks is concatenated with the query feature to form the final clip feature. Formally,

$$f = [f'_m, \text{Non-local}\left(L(f'_m + P_m), L([f'_H + P_H, f'_F + P_F])\right)]\,, \tag{6}$$

where $L(\cdot)$ is a linear projection and P_m, P_H, and P_F are positional encoding for the query, history, and future timestamps. We use a learning rate of $1e-3$ and a batch size of 512 during training.

LFB TF. We employ a two-layer Transformer encoder upon the concatenation of the history, query, and future features, *i.e.*,

$$f = \text{Transformer}([f'_H, f'_m, f'_F]{+}P) = \text{Transformer}(f'_{1:M}{+}P), \tag{7}$$

where P is positional encoding. The hidden size is the same as the feature dimensions, which is 2304. We use a learning rate of $1e-4$ and a batch size of 512 to train the model.

D. Additional Experiments

D.1. Accuracy of Each Question Answering Type

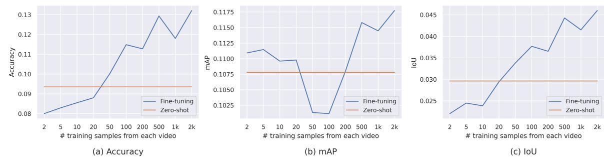
As discussed in Appendix A.4, our question answering task includes five different types of questions. In Table A7, we decompose the model performance in Table 3 into the five question types. We find that most of models perform the best on the task before/after task (TT) questions, while the miscellaneous (MI) questions are generally difficult to the VLLMs. However, given the transcript of the animation video, the oracle model makes great improvement on the miscellaneous questions, further highlighting the incapability of the VLLMs of abstracting the tasks in spacewalk videos. We notice that the oracle model performs poorly on the when task (WT) questions. This is expected because the time period options of these questions are based on the segmented one-hour video clip, while the oracle information given to the model covers the entire spacewalk mission.

D.2. Ablation Study on Fine-tuning Data Size

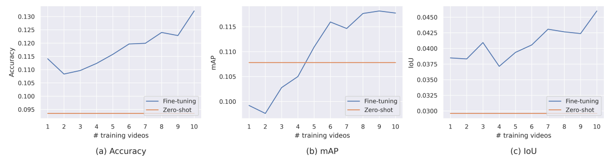
We conduct experiments to explore the impact of data size on all-layer fine-tuning InternVideo on the step recognition task. In Figure A11a, we ablate the number of samples drawn from each video in the training set. The model accuracy and IoU first rise as the data size grows. However, when more than 500 samples are drawn from each video (5k samples in total), the performance becomes saturated (less than 0.3% gain from 500 samples to 2k samples). This indicates that the number of training samples per video is not the bottleneck for our fine-tuning approach.

In Figure A11b, we vary the number of training videos, while 2k training samples are drawn from each video. The accuracy and IoU curves show significant upward trends when the videos get more, while mAP is saturated when more than 6 videos are used. This shows that fine-tuning on more spacewalk videos might better adapt the model the novel domain. However, the number of spacewalk videos is naturally limited by the number of real spacewalk missions, which disables large-scale data collection. So we urge the need for domain adaptation method with higher data efficiency.

All these experiments are conducted under a context window length of 1 minute. For different training data sizes, we keep the number of fine-tuning epochs the same. Therefore, the less training data, the smaller number of fine-tuning steps. However, we find that naively enlarging the number of training steps always leads to over-fitting.



(a) Number of training samples per video



(b) Number of training videos

Figure A11. Ablation studies on training data size. (a) Varying number of training samples per video (10 training videos in total). (b) Varying number of training videos (2000 training samples from each video). We full-layer fine-tune InternVideo with a context length of 1 minute. The blue curves are the fine-tuning performances while the orange straight lines are the zero-shot performances on the corresponding metrics.

D.3. Temporal Context Incorporation

In Section 5.5, we test models under varying context window lengths to investigate their capability to incorporate temporal context. The numerical results used in Figure 4 are listed in Table A8. Those used in Figure 5 are in Table A9.

Method	#Frames	TL	TT	WT	TO	MI	Acc
Random	-	25.00	25.00	25.00	25.00	25.00	25.00
LLaVA-Next-Video-7B	32	34.29	20.82	19.40	22.81	33.33	25.53
LLaVA-Next-Video-34B	32	26.67	33.33	25.37	29.82	33.33	29.52
VideoLLaMA2-7B	8	16.19	35.83	29.85	26.32	25.93	27.13
VideoLLaMA2-7B	16	25.71	40.83	20.90	36.84	29.63	31.65
Caption-enhanced LLM	60	33.33	35.00	26.87	15.79	25.93	30.37
GPT-4o	8	33.33	32.50	34.33	33.33	11.11	31.65
GPT-4o	16	27.62	36.67	29.85	24.56	25.92	30.32
GPT-4o	32	33.33	33.33	32.84	31.58	25.93	32.45
GPT-4o	60	30.48	34.17	37.31	28.07	22.22	31.91

Table A7. Accuracy on different question types of the question answering task. TL: task before/at/after location; TT: task before/after task; WT: when task; TO: task order; MI: miscellaneous.

Method	w = 1 min			w = 2 min			w = 3 min			w = 5 min		
	Acc.	mAP	IoU	Acc.	mAP	IoU	Acc.	mAP	IoU	Acc.	mAP	IoU
EgoVLP	6.34	8.92	2.17	7.73	9.83	2.59	9.68	10.66	3.03	10.21	10.86	3.18
VideoCLIP	8.40	9.80	3.21	9.01	10.40	3.28	9.98	11.14	3.71	8.87	10.39	3.46
InternVideo	10.12	**12.17**	**4.02**	11.22	12.30	4.22	11.13	**12.68**	4.04	10.08	**12.53**	4.04
LLaVA-Next-Video	**10.35**	-	3.77	12.03	-	**4.53**	13.71	-	5.07	13.82	-	4.92
VideoLLaMA2	9.34	-	2.88	**12.43**	-	4.32	**14.37**	-	**5.45**	**17.32**	-	**6.28**

Method	w = 10 min			w = 15 min			w = 20 min		
	Acc.	mAP	IoU	Acc.	mAP	IoU	Acc.	mAP	IoU
EgoVLP	8.63	10.13	2.65	7.42	9.37	2.12	8.33	9.61	2.48
VideoCLIP	8.08	10.44	3.36	8.47	9.79	**3.79**	7.23	8.82	3.20
InternVideo	10.26	**12.36**	4.18	9.66	**11.28**	3.54	9.94	**10.53**	3.87
LLaVA-Next-Video	11.85	-	4.58	11.80	-	-	12.05	-	4.00
VideoLLaMA2	**20.98**	-	**8.03**	**19.72**	-	-	**17.49**	-	**6.69**

Table A8. Last-layer fine-tuning performances of contrastive VLMs and zero-shot performances of VLLMs on the step recognition task under varying context window lengths. They are used to plot Figure 4.

D.4. Qualitative Results

In this section, we demonstrate qualitative results on the step recognition task and the question answering task. While we provide a few key frames for each video in our figures.

We illustrate examples on the step recognition task with a context window length w = 5min from Figure A12 to Figure A17. The model makes predictions mainly based off keywords shared by the transcript of the spacewalk video and the caption/transcript of the step animation. Besides, it is also aware of the objects (*e.g.*, adapater plate in Figure A12) and actions (*e.g.*, moving around in A13) appearing in both the spacewalk and animation videos. While the model correctly recognizes some steps, these cues can also lead to failure cases. Sharing keywords in the transcripts does not necessarily mean that the spacewalk clip belongs to the step. The models need to better utilize the semantic details in the spacewalk video and transcript to determine the corresponding step. Moreover, all-layer fine-tuning corrects InternVideo's predictions in some examples. The model learns the visual concept of airlock in Figure A14 and swages in Figure A15. Finally, there are still difficult examples for the evaluated models (Figure A16 and A17), demonstrating the room for model improvement.

In Figures A18 to A20, we show qualitative results on the question answering task. Here, LLaVA-Next-Video refers to LLaVA-Next-Video-34B. VideoLLaMA2 refers to VideoLLaMA2-7B with 16 input frames. And GPT-4o refers to GPT-4o with 32 input frames. In Figure A18, most of the models recognize that astronauts are connecting the cables after leaving from the robotic arm. However, In Figure A19, only the GPT4o with the overall spacewalk mission animation as oracle correctly identifies the temporal relations between tasks. Some of the models choose "(D) Luca and Drew remove debris shield", which indeed happens before they hand off the debris shield. In Figure A20, the models have difficulty localizing the spacewalk task of interest.

E. Additional Task – Intra-video Retrieval

In our step recognition task, the list of steps in the corresponding spacewalk mission is provided. However, the predefined steps are not always available for all the spacewalk mission. In this case, the segmentation of spacewalk steps

Method	w = 1 min			w = 2 min			w = 3 min			w = 5 min		
	Acc.	mAP	IoU	Acc.	mAP	IoU	Acc.	mAP	IoU	Acc.	mAP	IoU
Sparse Frame Sampling	10.12	12.17	4.02	11.22	12.30	4.22	11.13	12.68	4.04	10.08	12.53	4.04
Dense Frame Sampling	10.28	12.54	4.20	11.98	12.98	4.60	11.75	13.51	4.13	11.14	13.19	4.43
LFB Avg	11.47	**13.92**	4.39	11.98	**15.15**	4.49	11.71	**16.14**	4.29	12.37	17.25	3.73
LFB Cat	**12.22**	13.43	**4.49**	**12.76**	14.25	**4.82**	**12.98**	14.83	**4.95**	**13.12**	15.46	**4.76**
LFB NL	9.31	13.23	3.61	10.22	13.96	3.48	9.50	14.98	3.26	10.70	**17.28**	3.01
LFB TF	11.31	13.08	3.77	11.67	14.83	3.71	12.45	15.68	4.11	11.38	16.10	3.04

Method	w = 10 min			w = 15 min			w = 20 min		
	Acc.	mAP	IoU	Acc.	mAP	IoU	Acc.	mAP	IoU
Sparse Frame Sampling	10.26	12.36	4.18	9.66	11.28	3.54	9.94	10.53	3.87
Dense Frame Sampling	10.72	13.23	4.20	-	-	-	-	-	-
LFB Avg	10.86	**18.11**	3.35	11.39	**18.65**	3.03	10.77	18.29	2.68
LFB Cat	**13.20**	16.69	4.89	11.98	16.43	**4.52**	**12.78**	16.41	**4.84**
LFB NL	12.78	16.81	4.20	10.57	17.55	3.64	10.08	17.41	3.39
LFB TF	13.18	16.96	**5.00**	**12.98**	17.06	4.42	11.17	**18.79**	3.35

Table A9. Performances of different temporal context incorporation methods on the step recognition task. These methods are built upon frozen InternVideo. They are used to plot Figure 5.

must be done in a unsupervised manner. Current approaches to unsupervised video segmentation rely on clustering video features [14, 25]. This requires models to form distinguishable embeddings across steps. With this in mind, we explore an additional task – intra-video retrieval. This task evaluates the models capability of retrieving video clips that are from the same step as a query video, without any description about the step.

E.1. Task Definition

In a spacewalk video, given a query timestamp t_q and two candidate timestamps t_{c1} and t_{c2}, the task is to determine which of t_{c1} and t_{c2} is of the same step as t_q. Note that closer timestamps are more likely to be in the same step. Therefore, to avoid this shortcut, we involve only two equidistant candidates satisfying $|t_q - t_{c1}| = |t_q - t_{c_2}|$, where one of them to a different step and serves as a hard negative. To make them distinguishable, we ensure that the candidates are at least 30 seconds away from the query. Moreover, both the query and candidates should be at least 15 seconds away from their corresponding step boundaries to avoid ambiguity. Similar to the step recognition task, we set a context window length w and the model can access the video and transcript in the w-long windows centered at t_q, t_{c1}, and t_{c2}.

We construct 2000 samples from each spacewalk video for this task. For each annotated spacewalk clip, we first uniformly sample several query timestamps t_q's. Then, for each t_q, we randomly sample a distance d such that only one of $t_q - d$ and $t_q + d$ is in the same step as t_q and they follow the restrictions in the task definition. Finally, we randomly choose one of $t_q - d$ and $t_q + d$ as t_{c1} and the other as t_{c2}. We adjust the number of samples from each spacewalk clip to ensure roughly uniform numbers of queries from different spacewalk steps.

For performance measurement, we calculate the retrieval accuracy as the evaluation metric. Random chance is 50% accuracy on this task.

E.2. Evaluation Methods

E.2.1. Intra-video Retrieval by VLLMs

Zero-shot. To evaluate VLLMs on the intra-video retrieval task, we format it as multi-choice video question answering by asking "*Which candidate shows the same step as the query?*" after providing the query and candidate videos. Specifically, the prompt is as following:

```
You are given three spacewalk video clips from a
spacewalk mission, one query clip and two candidate
clips. The mission can be divided into multiple
steps. Please provide a single-number answer (1
or 2) to the following question, and your answer
must be either 1 or 2. You must not provide any
other response or explanation. If you are not sure,
answer with the most likely answer.
Here is the question: Which candidate shows the
same step as the query?
Query video clip:
Eight uniformly sampled video frames: <query video
frame 1> ... ... <query video frame 8>
Video speech transcript: <query video transcript>
Candidate 1 video clip:
Eight uniformly sampled video frames: <candidate 1
video frame 1> ... ... <candidate 1 video frame 8>
Video speech transcript: <candidate 1 video
transcript>
Candidate 2 video clip:
Eight uniformly sampled video frames: <candidate 2
video frame 1> ... ... <candidate 2 video frame 8>
Video speech transcript: <candidate 2 video
transcript>
```

Figure A12. In this example of step recognition, the spacewalk video transcript mentions that the astronaut is installing an adapter plate, and the video also shows them tightening the screws on the adapter plate. All models except LLaVA-Next-Video match this video with the step of installing adapter plate.

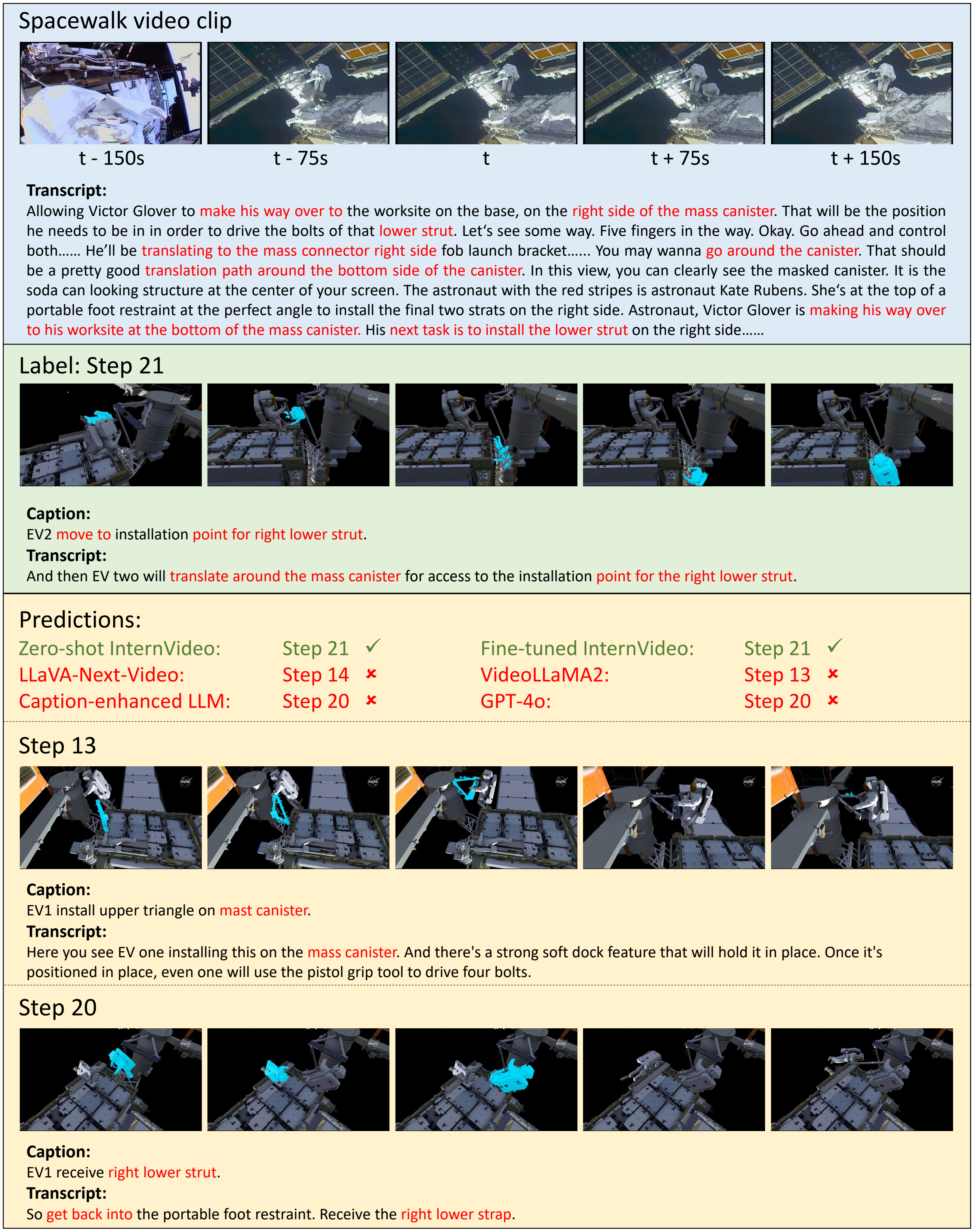


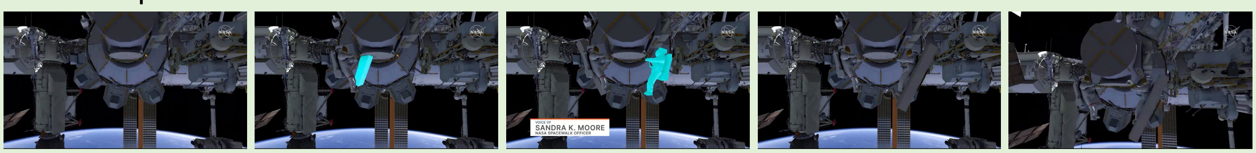
Figure A13. In this example of step recognition, both the spacewalk video and transcript indicate that the astronaut is translating around the mass canister, which aligns with the transcript of step 21. However, only InternVideo models make the correct prediction. The four VLLMs mismatch it, probably based on keywords "mast canister" (step 13) and "lower strut" (step 20).

Spacewalk video clip

t - 150s | t - 75s | t | t + 75s | t + 150s

Transcript:
His poles have been blocked. I'm going for the log. Copy. EV one anchor…… Floting in the airlock, you can see the strut bag. That is a specially designed bag that will that is carrying the pieces for the modification kit that will be built today. E v one, then I see a green light. Okay. To my I am in position, so we The the the camera as it's exposed camera is on the forward camera side. K. Eight. Are you ready for the bag? Yeah…… That strap bag carries the parts that will be used to assemble the modification kit this morning. Okay. I think you can look right. Straight back to your lock. You have a light on it. And here it comes the back…… I have control. Okay. Right now. Uh-huh. Go ahead. I'm gonna stay on the outside. What are you with that? I think I can disconnect my waist tether from the dial up during extender issue. Yes. You have a go for that. Yeah. We concur as well. Alright. And, Akie, while you're, waiting for Samantha egress……

Label: Step 1

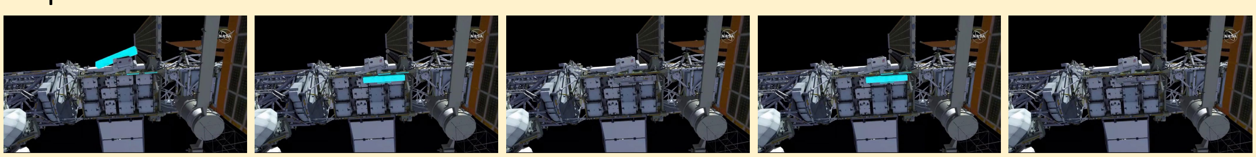


Caption:
EV1 & EV2 exit airlock.
Transcript:
US EVA seventy seven will begin at the Quest Airlock. Jackson astronaut, Aki Hochde, EV one, noted by the red stripes, will egress first and receive and hold the eight foot mod kid's strut bag. European astronaut to Ma Pes Gay will egress second with the full white suit.

Predictions:

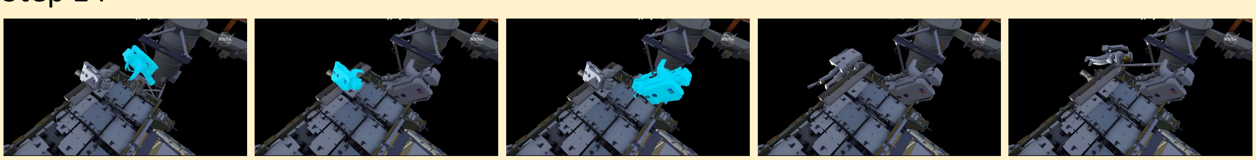
Zero-shot InternVideo:	Step 14 ✗	Fine-tuned InternVideo:	Step 1 ✓
LLaVA-Next-Video:	Step 1 ✓	VideoLLaMA2:	Step 1 ✓
Caption-enhanced LLM:	Step 4 ✗	GPT-4o:	Step 1 ✓

Step 4



Caption:
EV2 stow strut bag.
Transcript:
While tomorrow is doing the very large strip bag at p four, Aki will complete his translation out to the p four beta Gimbal assembly where he will install and set up his quick fix.

Step 14



Caption:
EV2 collect tools and prepare for right side.
Transcript:
Small then gather tools and reset for the right hand side. Rubbering your tools back to the bag while arching egresses and reposition the foot restraint to bias it to the right side.

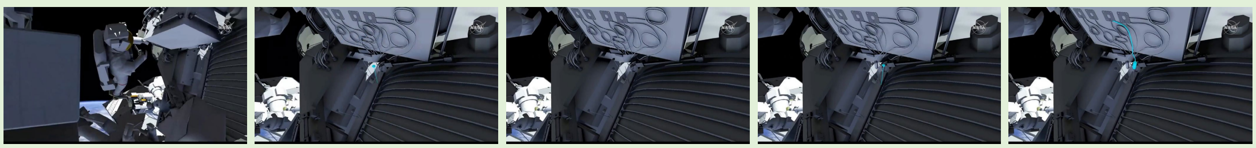
Figure A14. In this example of step recognition, the spacewalk video shows the astronauts exiting the airlock. While zero-shot InternVideo makes an incorrect prediction, fine-tuned InternVideo learns the concept of airlock and gives the true answer. The mistakes made by zero-shot InternVideo and caption-enhanced LLM are probably due to keywords "strut bag" (step 4) and "egress" (step 14).

Spacewalk video clip

t - 150s | t - 75s | t | t + 75s | t + 150s

Transcript:
Hey, Drew. Next, I can get the straightener. Okay. In hand. It looks like this card is unlocked. K. There we go. Back to me, administrator. The right length's been on the straightener...... Luca. Oh, okay. K. I have the tape installed. Check me down the gauge. Okay. Copy that. No. I have to retake it. It's moved to auto doing it. Okay. Copy that. Able to get it off completely. That it's not it's not completely, but I won't be able to reboot it. Certainly not representative of Major's work, amazing work on this project..... For a good swage, the tube has to be put into the alternate fitting, a very specific length marked off by the Kapton tape, measured by the tube straightener, both in the hands of Luca Parmitano right now. Those precise measurements, required for a good swage to make sure it's pinched at just the right point. This one's on. Primatano just working to make sure that that, measurement is precise......

Label: Step 10

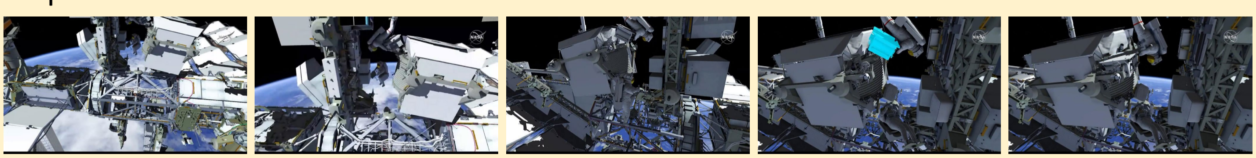


Caption:
Luca connect six fluid connections.
Transcript:
There'll be six fluid connections completed here at this VSP work site using, commercial off the shelf swages. Those swages have any encased in a, custom fitting that the engineering team to develop to allow the astronauts pressurized gloves to operate the small ferrals of the swage.

Predictions:

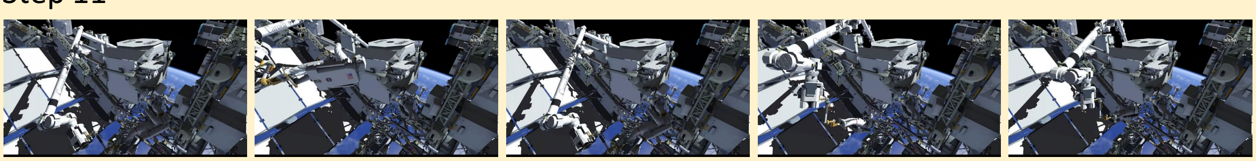
Zero-shot InternVideo:	Step 7 ✗	Fine-tuned InternVideo:	Step 10 ✓
LLaVA-Next-Video:	Step 14 ✗	VideoLLaMA2:	Step 1 ✗
Caption-enhanced LLM:	Step 10 ✓	GPT-4o:	Step 11 ✗

Step 7



Caption:
Luca & Drew install pump system.
Transcript:
Drew will provide eyes on to assist Luca in aligning system and install it onto AMS into the mechanical attachment device.

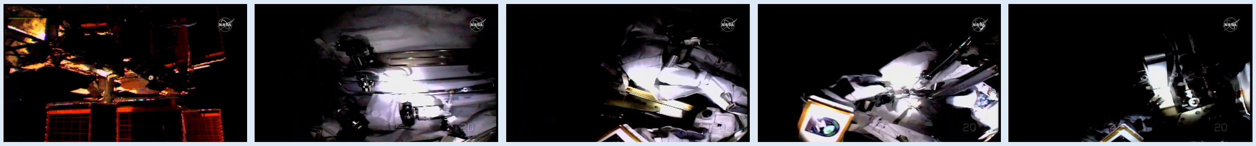
Step 11



Caption:
Robotic arm takes Luca to underside of AMS.
Transcript:
Once those six swages are completed, the SSRMS will take Luca to the underside of AMS for two final swages.

Figure A15. In this example of step recognition, the spacewalk videos shows the astronaut connecting swages, while its transcript also mentions them. However, only fine-tuned InternVideo and caption-enhanced LLM make the correct predictions. The mismatch to step 11 by GPT-4o is possibly because of the keyword “swage”.

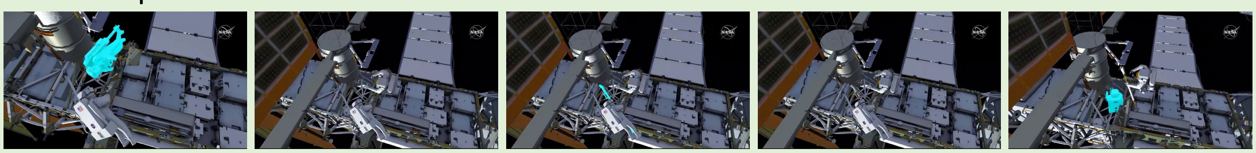
Spacewalk video clip



t - 150s | t - 75s | t | t + 75s | t + 150s

Transcript:
Am I having something? Oh, okay. Good…… I'll just do the five pound. Good words. Hey, Reddit. Kindly missed your ret. Nope. You have three left. Okay. Oh, where's the refi? I can do it. Yeah. Yeah. Okay. I'm already turned out for sure. And, as in some IPs, I have the car. Okay. Toma, you'll be retrieving the Okay. Okay. Next tomorrow, you'll be retrieving the left lower stretch under straps five and six. And handing that off. That's all the slots five and six coming up. I mean, I got It's red on my BRT with the rest. It's rattles out. Copy. Good config, Okey. Pesk now working to treat the left lower strut from the strap bag. Once he has that, he'll pass it along to Hoshide, who is in the AP far or articulating portable foot restraint? Once Hoshide has the lower strut, he will install it to the mounting bracket using the PGT or the pistol grip tool, which has been used quite so far this morning. And tomorrow, you'll be handing the clevis bolt side……

Label: Step 11



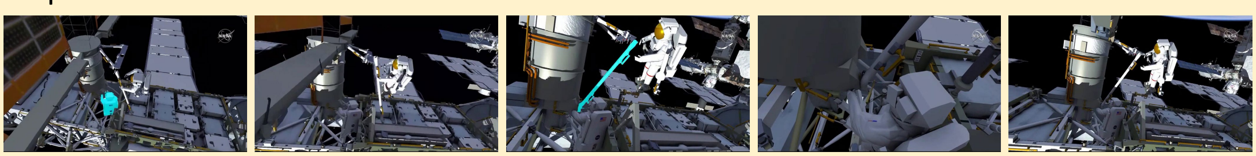
Caption:
EV1 enter foot restraint and receive strut.

Transcript:
Once complete with the upper triangle, Aki will egress the foot restraint, bias it to the left hand side while Tamal prepares the hand a left mid strut for BRT stow. Aki will ingress the APFR and hold the lower strut while Tama reposition to the solar array blanket box for saab bearing to install the left lower struts.

Predictions:

Zero-shot InternVideo: Step 28 ×
LLaVA-Next-Video: Step 1 ×
Caption-enhanced LLM: Step 12 ×
Fine-tuned InternVideo: Step 28 ×
VideoLLaMA2: Step 11 ✓
GPT-4o: Step 11 ✓

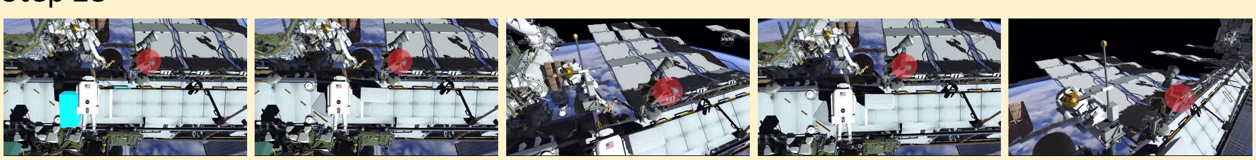
Step 12



Caption:
EV1 & EV2 install lower left strut.

Transcript:
Tomah will begin driving this bolt by hand four turns while Aki aligns his end and drives his to the mounting bracket two turns. Tomaw will finish his bolt by driving it with a pistol grip tool, followed by high torque with the EVA torque wrench. Once the bolt is deemed good, off you will be given a go to drive his bolt of torque using the crystal grip tool. This will complete our minimum config for the mod kit.

Step 28



Caption:
EV1 stow spare FPMU.

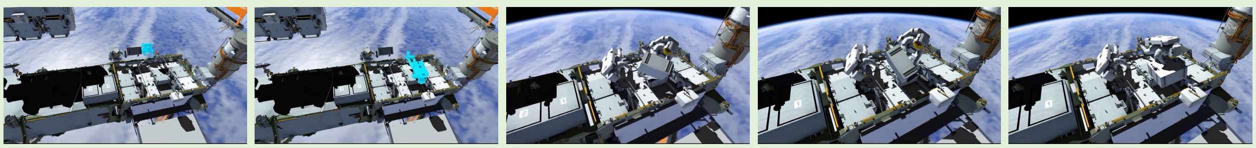
Transcript:
And meet Tamah at the FPMU at p one. He will stow the spare in a location optimal for hand off.

Figure A16. In this example of step recognition, the spacewalk video is dark but the transcript includes a command that one astronaut would hand off a strut to the other. As this example is complicated to understand, only VideoLLaMA2 and GPT-4o predict the step label correctly.

Spacewalk video clip

Transcript:
The alpha seven clockwise two. You're going to h two. Confirm socket tape line flush. Looking for sixteen to seventeen turns. Alpha seven, a price two on h two. And if they find this flush okay. Matures moving. Are you appears to be going ahead? I agree. This one seems snugger up against the wall than the other guy...... You'll confirm the tape line flush looking for four to five and a half turns. Copy. You can remove it? It's twelve thousand indicator is not quite to a block nine point two And it looks like we had about that four turns. Copy. Four turns. And torque was nine point two, but the status indicator doesn't quite look locked. Status indicator on battery is maybe a quarter inch for a block. Okay. We're good with that. You can hand that PGT back to Chris to Stone a swing arm and release that RET. That far. Plus with the JSON patterns, everything looks straight.....

Label: Step 31

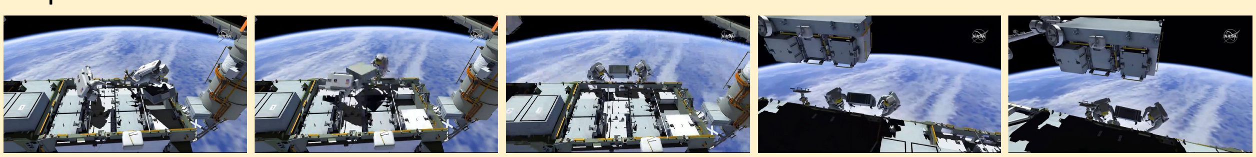


Caption:
Chris & Bob install new battery in empty slot 3.
Transcript:
They repeat the steps to release the bolts, translate back to the truss, And together, install an empty slot number three.

Predictions:

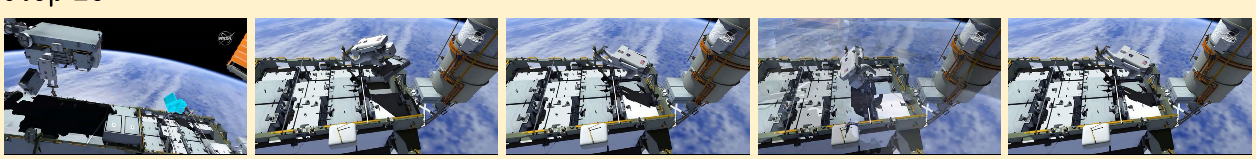
Zero-shot InternVideo:	Step 23 ✗	Fine-tuned InternVideo:	Step 25 ✗
LLaVA-Next-Video:	Step 14 ✗	VideoLLaMA2:	Step 18 ✗
Caption-enhanced LLM:	Step 6 ✗	GPT-4o:	Step 16 ✗

Step 18



Caption:
Chris & Bob move battery to EP.
Transcript:
Next, the battery in slot two is removed and translated over to the pallet.

Step 23



Caption:
Bob installs adapter plate in open slot 2.
Transcript:
Bob translates back to the truss to install the adapter plate in open slot two.

Figure A17. In this example of step recognition, the video clearly shows an astronaut tightening the screws on a white battery. However, all the models fail to recognize it.

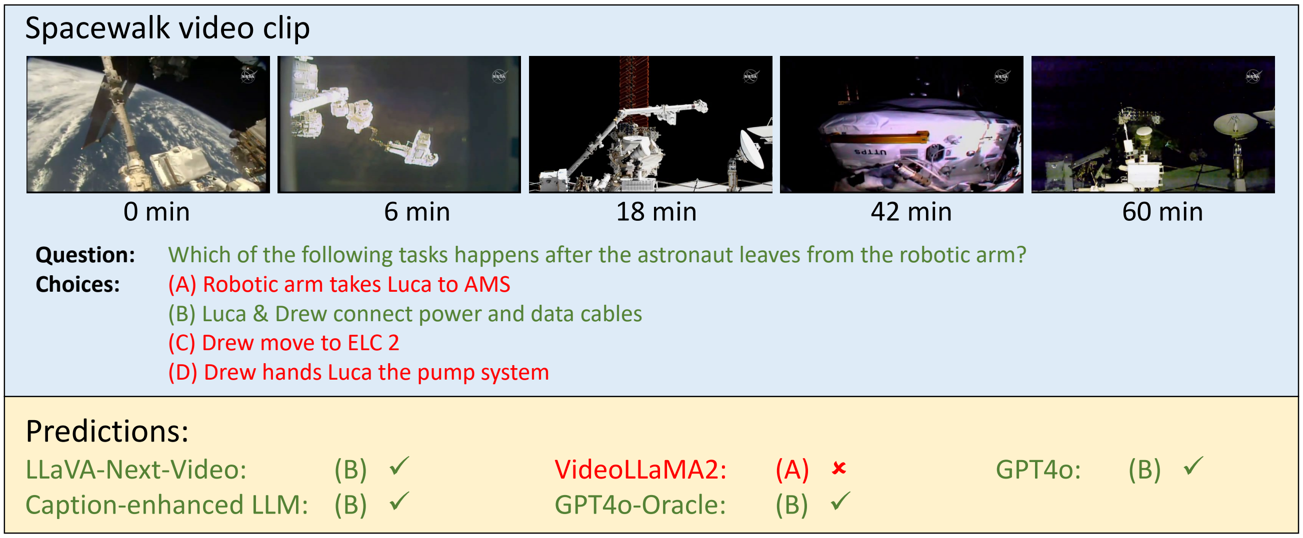


Figure A18. In this example of spacewalk question answering, the astronaut is on the robotic arm in the first 18 minutes. After that, as shown in the 42nd minute, they are connecting cables to an equipment. So the answer to the question is “(B) Luca & Drew connect power and data cables”. All the models except VideoLLaMA2 correctly answer the question.

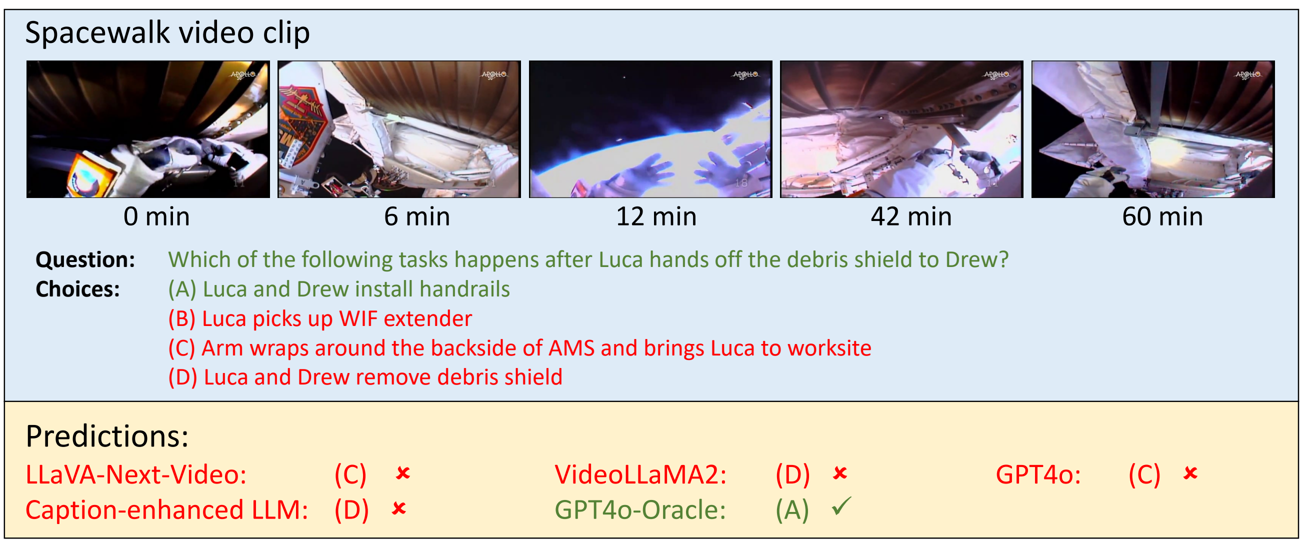


Figure A19. In this example of spacewalk question answering, the astronaut first remove and jettison the debris shield and then install handrails. They remove the debris shield in the 0th minute, hand it off in the 6th minute, and jettison it in the 12th minute. So the only option happened after they hand off the debris shield is “(A) Luca and Drew install handrails”. Only the GPT4o model with the spacewalk mission animation as oracle manages to answer the question.

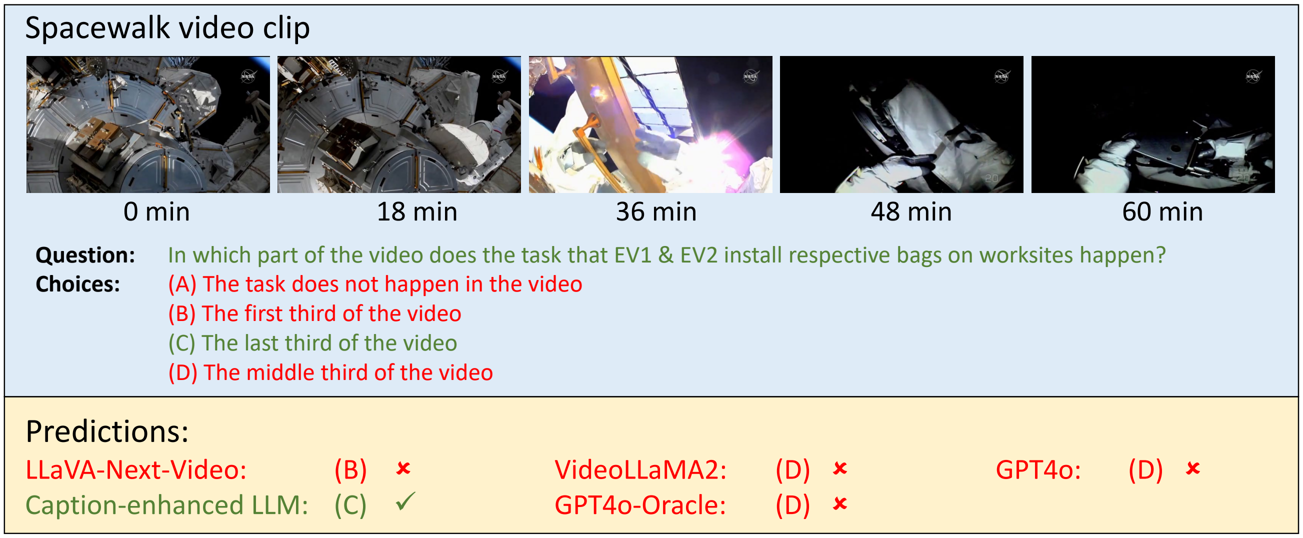


Figure A20. In this example of spacewalk question answering, the astronauts exit the airlock in the first third of the video, move along the handrails in the middle third, and finally install the respective bags in the last third. Therefore, the correct answer is "(C) The last third of the video". Only the caption-enhanced LLM gives the correct answer to the question.

Because this task has three videos as input, we only evaluate VLLMs that can process interleaved videos and texts.

E.2.2. Intra-video Retrieval by Contrastive VLMs

Zero-shot. In this task, we match a query spacewalk clip centered at timestamp t_q with one of two candidate clips centered at t_{c1} and t_{c2} respectively. In contrast to step recognition, we **only use video features** and ignore the transcripts. This proved the best in Section E.3. After calculating the similarities between the query and the candidates, we pick the candidate with higher score as the prediction.

Last-layer Fine-tuning. We freeze the models and train two linear layers to align the query and candidate features. One is $G^q_\theta(\cdot)$ for queries and the other is $G^c_\theta(\cdot)$ for candidates. We optimize cross entropy loss during training. Denoting the query feature as f^q and candidate features as f^c_1 and f^c_2, it is

$$\mathcal{L} = -\log \frac{\exp\left(G^q_\theta(f^q)\cdot G^c_\theta(f^c_y)\right)}{\sum_{1\le i\le 2}\exp\left(G^q_\theta(f^q)\cdot G^c_\theta(f^c_i)\right)}. \tag{8}$$

After training, the candidate with higher similarity to the query, $G^q_\theta(f^q)\cdot G^c_\theta(f^c_i)$, is pick as the prediction.

We use the same hyperparameters as the last-layer fine-tuning for the step recognition task in Appendix C.3.

All-layer Fine-tuning. We fine-tune the backbone of the pre-trained models using the same training objective as that of the last-layer fine-tuning. The models learn to extract features that are critical to distinguish spacewalk steps. Specifically, we fine-tune EgoVLP on the intra-video retrieval task using AdamW [21] optimizer with a learning rate of $1e-6$. The model is trained for 1 epoch with a batch size of 32.

E.2.3. Unsupervised Segmentation Method

TW-FINCH [41] is a clustering-based unsupervised action segmentation method. We employ it to solve the intra-video retrieval task as a baseline. For a spacewalk recording in the test set, we first extract InternVideo features with a sliding window. The window length is 12 seconds and the step is 1 second. We sample video frames at 1 FPS to feed the video encoder, *i.e.,* 12 frames per video clip. Based on these clip features, TW-FINCH clusters the entire spacewalk recording into N_{seg} segments, each of which is regarded as a step. For a retrieval query timestamp t_q with two candidate timestamps t_{c1} and t_{c2}, if exactly one of the candidates falls in the same segment as t_q, we regard it as our prediction. Otherwise, we randomly pick one of them as the prediction. We find that the best configuration is $N_{\text{seg}} = 100$ with only video features.

E.3. Evaluation Results

Table A10 shows the model performances on the intra-video retrieval task. GPT-4o achieves the best performance of 71%, while the 70.93% accuracy of EgoVLP is competitive. Different from the step recognition task, last-layer fine-tuning cannot improve the model performance in most cases. All-layer fine-tuning only increases the model accuracy very slightly. Besides, while the context window length expands, the model performance first increases and then degrades after a peak. This observation is in line with the step recognition results, which reveals their incapability of digesting long video contexts.

Table A11 ablates the modalities on intra-video retrieval in zero-shot setting. The video-only approach achieves the

Method	Accuracy					
	(w=)10s	20s	30s	1min	2min	3min
Random	50.00	50.00	50.00	50.00	50.00	50.00
TW-FINCH*	-	-	-	-	-	64.10
Zero-shot						
EgoVLP	69.04	69.95	**70.92**	65.76	**60.20**	56.43
VideoCLIP	66.91	67.48	67.88	63.52	50.72	52.07
InternVideo	67.60	69.12	69.82	**66.76**	56.46	56.45
GPT-4o	**71.00**	**71.00**	70.83	58.50	53.08	**56.67**
Last-layer Fine-tuning						
EgoVLP	**68.63**	**69.91**	**70.12**	**66.51**	**57.46**	**52.36**
VideoCLIP	61.15	60.53	61.29	59.00	54.08	52.23
InternVideo	65.11	65.33	64.97	62.62	56.02	52.24
All-layer Fine-tuning						
EgoVLP	68.78	70.18	70.93	69.79	62.58	58.80

Table A10. Model performances on the intra-video retrieval task. GPT-4o achieves the highest accuracy while EgoVLP is comparative. $*$: TW-FINCH have unlimited context.

Method	Accuracy					
	w = 20 s			w = 30 s		
	V	T	V+T	V	T	V+T
EgoVLP	69.95	48.73	63.71	**70.92**	47.85	64.95
VideoCLIP	67.48	51.65	56.56	67.88	50.04	57.14
InternVideo	69.12	51.55	63.15	69.82	50.23	63.08
GPT-4o	**71.00**	**52.42**	**69.25**	70.83	**53.67**	**69.92**

Table A11. Ablation about input modality on intra-video retrieval. The models are evaluated in zero-shot scenario. V: video; T: text (captions and transcripts).

highest accuracy across all models, while the text-only one performs poorly. This is different from step recognition where videos contribute less than texts. Because the query and candidates here are the same modality with the same visual distribution, whereas we need to match the real spacewalk videos with their animations in the step recognition task. Moreover, combining video and text does not improve the performance, indicating that transcript is effectively noise for the current method on this task. This motivates the future development of stronger models that can utilize the text signal on this task to facilitate unsupervised segmentation of spacewalk videos.

E.4. Qualitative Results

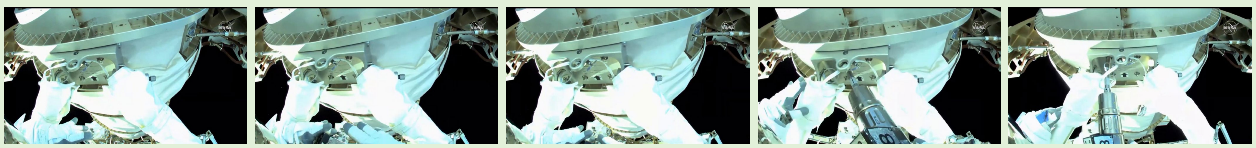
On the intra-video retrieval task, we show two qualitative examples with context window length $w = 30$s in Figures A21 and A22. We find that the models tend to retrieve candidate videos naively based off the visual scene appearance. This helps them do well in simple scenarios (Figure A21). However, there are more difficult test samples. In Figure A22, the query and candidate 1 show the same step but from different views. All the models fail in this example. Better spacewalk video understanding capability is necessary for future models to solve it.

Query

t - 15s | t – 7.5s | t | t + 7.5s | t + 15s

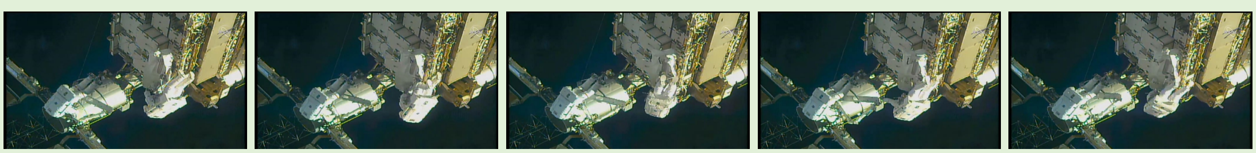
Step label:
Step 13 - EV1 install upper triangle on mast canister
Transcript:
Now it's up. And you'll engage the left side first and then pivot the right side to engage the soft dock. Mike, you'll be working on stowing your short PGT on handrail five three seven two. That's near the APFR wrenches. Handbrake embedded in the soft dock.

Candidate 1



Step label:
Step 13 - EV1 install upper triangle on mast canister
Transcript:
Great views from the high definition camera view of Kate Rubin's helmet camera. Alpha two, clockwise two. She's confirming the pistol grip settings before she goes ahead and starts driving some of the bolt that you see on the center pad there that's currently soft dock to the mass canister. Driving those bolts will secure it in place. Through mic eight to torque. We anticipate eleven and a half turns. Copy. Eleven and a half torque.

Candidate 2



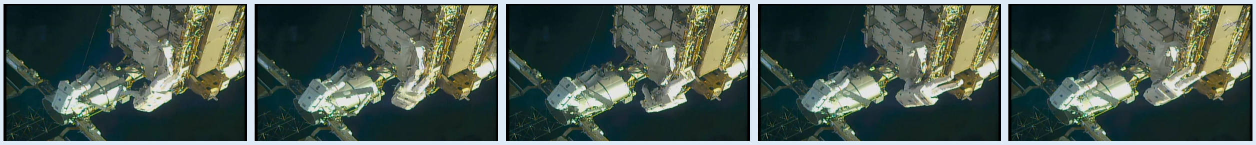
Step label:
Step 12 - EV2 hand upper triangle to EV1
Transcript:
Yeah. We want the RET back in the bag. Sorry. You you don't have to release it from the mounting bracket. Okay. Either you can relocate or you can hook it back this way, and I'll be No. It'll text me. Okay. I'm sending it back to a queue. It's still of control. Again, Rubens has control of that upper bracket. They're just making sure that the tether configuration is okay before she begins the work of installing it to the mass canister.

Predictions:

Zero-shot EgoVLP: Candidate 1 ✓ Fine-tuned EgoVLP: Candidate 1 ✓
GPT-4o: Candidate 1 ✓

Figure A21. In this example, both the query video and candidate 1 show an astronaut installing an upper triangle. They are in the same scene while candidate 2 is different. All the models make the correct predictions.

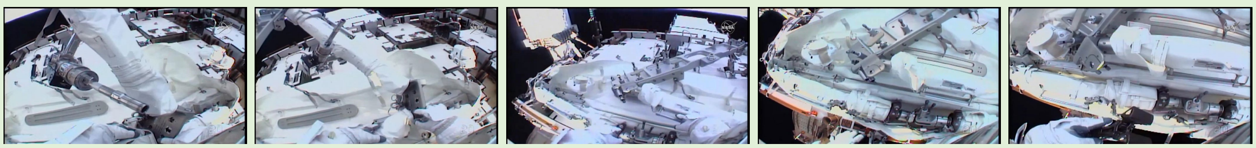
Query



t - 15s | t – 7.5s | t | t + 7.5s | t + 15s

Step label:
Step 12 - EV2 hand upper triangle to EV1
Transcript:
And I have my RET on the right. K. Let me know when you already take control. I already take control. And you guys I I got control. Alright. Glover just handed off the upper triangle to Kate Rubins. Okay. And, like, on case go, you can release the bad RET. And I have control. I have it ready to go. The back. RET. He wants the RET to go back to the back or to the mounting bracket.

Candidate 1



Step label:
Step 12 - EV2 hand upper triangle to EV1
Transcript:
Like, looks like you're doing a great job on MOI. I think they're happy with that config, and you can pass the triangle up to Kate when you're ready. Works.

Candidate 2

Step label:
Step 13 - EV1 install upper triangle on mast canister
Transcript:
And you'll engage the left side first and then pivot the right side to engage the soft dock. Mike, you'll be working on stowing your short PGT on handrail five three seven two. That's near the APFR wrenches. Handbrake embedded in the soft dock. Copy. Soft dock. Thank you. K. On your long PGT, your settings will be alpha two clockwise two.

Predictions:
Zero-shot EgoVLP: Candidate 2 ×
Fine-tuned EgoVLP: Candidate 2 ×
GPT-4o: Candidate 2 ×

Figure A22. In this example, all three videos are visually dissimilar. The query video shows an astronaut handing an upper triangle to the other from a third-person view, while candidate 1 shows it from a first-person view. In candidate 2, they are installing the upper triangle. However, all the models fail to recognize candidate 1 as the same step as the query.

Contributions to Spacewalk-18: Scalable Annotation of Multimodal Long-Form Procedural Videos in Novel Domains

Rohan Myer Krishnan[1]

[1]**Brown University, Department of Computer Science**

ABSTRACT

Spacewalk-18 is a novel benchmark for advancing multimodal, long-form procedural video understanding. It comprises 96 hours of annotated spacewalk recordings from the International Space Station—an entirely new domain for video-language research. A key contribution of this work is the design and implementation of a scalable annotation pipeline and a publicly released annotation interface, the Spacewalk Video Annotation Tool (https://spacewalk-data.org/), which enables efficient collection of high-quality temporal step annotations from non-expert annotators. This report details the dataset curation process, including video and transcript preparation, segmentation, and annotation. The resulting Spacewalk-18 dataset and tooling provide the first large-scale resource for benchmarking multimodal, long-range video-language models on structured procedural tasks in an extreme, real-world environment.

Keywords: video understanding, multimodal learning, temporal segmentation, human annotation, dataset curation

INTRODUCTION

Spacewalk-18 introduces a new frontier for multimodal video understanding: long-form procedural demonstrations recorded in the highly specialized domain of extravehicular activities (spacewalks) outside the International Space Station. Existing video-language benchmarks focus primarily on short, everyday activities in household or internet-sourced settings. In contrast, Spacewalk-18 comprises 18 multi-hour expert demonstrations in a zero-gravity, visually distinct environment, making it the first benchmark of its kind.

A critical enabler of this benchmark is the Spacewalk-18 dataset, a structured collection of over 96 hours of annotated spacewalk video. This report describes the novel data collection and annotation pipeline, including the development of a publicly released annotation interface that made it possible to scale high-quality human-in-the-loop labeling to this extreme domain.

DATASET CONSTRUCTION

Video and Transcript Preparation

We sourced 18 full-length spacewalk recordings from NASA's publicly available YouTube broadcasts, each spanning 7 to 8 hours. Each recording begins with a procedural animation outlining the planned steps of the mission, which served as the basis for constructing the annotation label space.

Audio tracks were processed using Deepgram's Automatic Speech Recognition (ASR) system, selected after comparative evaluation of multiple ASR providers. Recordings were segmented into 10-minute audio chunks to optimize transcription accuracy, producing time-aligned transcripts for the entire dataset.

Clip Segmentation

Given the extreme length of the source videos, we applied automatic shot boundary detection using PySceneDetect to segment recordings based on camera angle changes. This over-segmentation strategy ensured that each clip contained at most a single procedural step, simplifying the annotation task and reducing cognitive load on annotators.

Label Space Definition

We manually segmented the animated mission previews into distinct procedural steps and authored corresponding textual descriptions. These labels formed the step taxonomy for each spacewalk, capturing mission-specific objectives and ensuring consistency in downstream annotation.

ANNOTATION PIPELINE

Novel Annotation Interface

A key contribution of this work is the design and public release of the Spacewalk Video Annotation Tool, available at `https://spacewalk-data.org/`. This purpose-built, web-based interface enabled scalable human annotation of long-form, multimodal procedural videos. The tool provided:

- Interactive access to the full sequence of animated step previews and descriptions.
- A focused video clip to label, with navigable access to surrounding context clips.
- Labeling options including predefined steps, "Irrelevant" for off-task footage, and "Unsure" for ambiguous content.

The interface uniquely allowed annotators to reference the entire spacewalk timeline, balancing local clip-level labeling with global procedural awareness. This design significantly reduced annotation effort compared to traditional boundary-drawing methods, while maintaining high labeling quality.

Quality Assurance and Aggregation

To ensure label reliability, all annotators completed a qualification task requiring 80% accuracy on a held-out validation set. Each clip was labeled by at least three independent annotators, and majority voting was used to resolve the final label. Post-processing merged adjacent clips with identical labels into continuous temporal segments, producing high-quality procedural annotations.

CONTRIBUTION TO BENCHMARK DEVELOPMENT

In addition to leading dataset construction and tool development, I contributed to the writing of the Spacewalk-18 benchmark paper. This included documenting the dataset collection methodology, describing the annotation interface, and analyzing the dataset's novel multimodal and long-form characteristics.

CONCLUSION

Spacewalk-18 represents the first large-scale dataset for procedural video understanding in a domain as specialized and visually distinct as spacewalks. The Spacewalk Video Annotation Tool, publicly available at `https://spacewalk-data.org/`, is a novel contribution that enables scalable human-in-the-loop annotation of long-form multimodal videos. Together, these contributions provide a foundation for advancing research in multimodal reasoning, domain generalization, and long-range video understanding.