Task Preference Optimization: Improving Multimodal Large Language Models with Vision Task Alignment

Supplementary Material

1. Experiment Details

MVbench. We present the detailed performance of MVBench in Table 1, VideoChat-TPO achieves an average score of 66.8, increasing by 6.4 points based on VideoChat2. It gets superior performance among MLLMs with the same number of input frames and LLMs of comparable model scale. In Action Localization, temporal labels in the VideoChat2-Textualized-Task are defined as text. While the model demonstrates strong capabilities in zero-shot temporal grounding, converting the task into a QA problem does not improve performance. However, by optimizing with TPO, the model can benefit from original label supervision, resulting in corresponding performance enhancements. Also, Its superior performance is particularly evident in tasks that require moment-based perception and reasoning, including Action Sequence (AS), Action Localization (AL) and Action Prediction (AP), with scores of 84.0 (+7.5%), 55.0 (+10%), and 69.5 (+13.5%) respectively. This demonstrates the excellent potential of TPO in sophisticated video understanding tasks.

MMIU. The results are shown in Table 2. VideoChat-TPO shows a significant improvement over VideoChat2, achieving an overall score of 40.2 (+5.2%). Compared with VideoChat2, Our model has achieved clear improvements in Causality Reasoning (CR), Visually Grounded Reasoning (VGR), Multiple Image Captioning (MIC), Spot the Difference (STD), General Action Recognition (GAR), Temporal Localization (TL), Video Captioning (VidCap), Multiview Action Recognition (MAR), Image Captioning with Spatial Context (ICSC), and Egocentric Video Question Answering (EVQA), with scores of 73.0 (+26.5%), 69.5 (+15.2%), 83.0 (+19%), 92.5 (+61%), 88.0 (+15%), 94.5 (+13.0%), 73.4 (+35.8%), 48.5 (+12%) and 59.0 (14.5%), respectively. Among them, we suppose the improvement of TL capability comes from the optimization of our temporal head, and the improvement of VGR, STD, MAR and ICSC capabilities comes from the optimization of our region head and mask head. The enhancements observed in captioning, specifically in metrics such as MIC, IC, and VidCap, indicate an improvement of TPO to capture detailed visuals. Meanwhile, we find that the improvement in multi-image capabilities stems from enhanced instruction comprehension. Compared with video assessments, which primarily consist of multiple-choice questions, multi-image evaluations emphasize the accuracy of responses to specific questions. After optimization with TPO, the model has significantly improved its instruction following, leading to a higher success rate.

How Scaling Task Data Affect MLLMs. We perform an ablation experiment on the dataset of stage 2 to evaluate the impact of the task training data on the model performance. Specifically, we reduce the number of temporal grounding datasets from six to one (QVHighlight [18]). As shown in Table 3, using only one dataset leads to slightly worse conversational performance (-0.3%) on MVBench and significantly poorer expert task performance (-5.6%) on Charades-STA R@0.5, when compared to employing multiple temporal grounding datasets for training the temporal task head. Notably, this approach remains more effective than training after textualizing the task data in QA tasks like MVBench. This finding indicates that scaling task data gives notable performance improvements in both multimodal and specific vision tasks. Various datasets are necessary for effectively enhancing TPO's dialogue capabilities and achieving zero-shot generalization to fine-grained visual tasks.

LLaVA-OV-TPO Performance on Video Benchmarks. According to Table 4, TPO method demonstrates performance improvements on LLaVA-OV [19] across multiple video benchmarks as it does in VideoChat [22] model. Since TPO uses extra visual cues to guide MLLM, LLaVA-OV-TPO achieves an average score of 64.8(+8.1%) on MVBench [22] and 64.0(+6.9%) on PerceptionTest [33]. The notable improvement clearly demonstrates the model's greatly enhanced ability to perceive visual details. Moreover, LLaVA-OV-TPO achieves a 3.1% performance improvement on VideoMME [12] and shows that the model has also made progress in knowledge modeling and understanding long videos. These results suggest TPO method is effective across various models and is particularly beneficial for fine-grained perception tasks.

2. Training and Data Details

Table 5 and 6 lists the detailed training configurations and data of VideoChat-TPO in different stages. In each stage, the model is parametrized from the weights from the previous stage and continues training.

Settings of Stage 1. The LLM is equipped with LoRA [16] for saving computational memory, using a LoRA rank of 16 and an alpha of 32. Only the LoRA is

Model	Avg.	AS	AP	AA	FA	UA	OE	OI	OS	MD	AL	ST	AC	MC	MA	SC	FP	CO	EN	ER	CI
VideoChatGPT [28]	32.7	23.5	26.0	62.0	22.5	26.5	54.0	28.0	40.0	23.0	20.0	31.0	30.5	25.5	39.5	48.5	29.0	33.0	29.5	26.0	35.5
VideoLLaMA [43]	34.1	27.5	25.5	51.0	29.0	39.0	48.0	40.5	38.0	22.5	22.5	43.0	34.0	22.5	32.5	45.5	32.5	40.0	30.0	21.0	37.0
VideoChat [21]	35.5	33.5	26.5	56.0	33.5	40.5	53.0	40.5	30.0	25.5	27.0	48.5	35.0	20.5	42.5	46.0	26.5	41.0	23.5	23.5	36.0
TimeChat [37]	38.5	40.5	36.0	61.0	32.5	53.0	53.5	41.5	29.0	19.5	26.5	66.5	34.0	20.0	43.5	42.0	36.5	36.0	29.0	35.0	35.0
Video-LLaVA [23]	43.0	46.0	42.5	56.5	39.0	53.5	53.0	48.0	41.0	29.0	31.5	82.5	45.0	26.0	53.0	41.5	33.5	41.5	27.5	38.5	31.5
P-LLaVA-7B [40]	46.6	58.0	49.0	55.5	41.0	61.0	56.0	61.0	36.0	23.5	26.0	82.0	39.5	42.0	52.0	45.0	42.0	53.5	30.5	48.0	31.0
ShareGPT4Video [6]	51.2	49.5	39.5	79.5	40.0	54.5	82.5	54.5	32.5	**50.5**	41.5	84.5	35.5	62.5	75.0	51.0	25.5	46.5	28.5	39.0	51.5
ST-LLM [25]	54.9	66.0	53.5	84.0	44.0	58.5	80.5	73.5	38.5	42.5	31.0	86.5	36.5	56.5	78.5	43.0	44.5	46.5	34.5	41.5	58.5
VideoGPT+ [27]	58.7	69.0	60.0	83.0	48.5	66.5	85.5	75.5	36.0	44.0	34.0	**89.5**	39.5	**71.0**	**90.5**	45.0	53.0	50.0	29.5	44.0	60.0
VideoChat2 [22]	60.4	75.5	58.0	83.5	50.5	60.5	87.5	74.5	**45.0**	47.5	44.0	82.5	37.0	64.5	87.5	51.0	66.5	47.0	**35.0**	37.0	**72.5**
VideoChat2-textualized-task	64.8	76.5	56.0	**88.5**	**52.5**	77.0	**92.5**	74.0	41.0	**50.5**	45.0	87.0	47.0	**74.0**	89.0	48.0	85.0	45.0	34.0	58.5	**73.0**
VideoChat-TPO	**66.8**	**84.0**	**69.5**	87.5	52.0	**77.0**	92.0	**81.0**	40.5	42.5	**55.0**	89.0	**47.5**	68.0	89.0	**58.0**	**87.0**	**57.5**	27.0	**60.0**	72.0

Table 1. **Results on MVBench Multi-choice Question Answering.**

Model	Overall	CR GuAR	ER GNAP	FD TC	FC VClz	SC VCo	VCor VO	VQA EVQA	VGR HE	FR IQASC	HR ICSC	I2IR ISTE	MIC ITRSC	PR MAR	S2IR MR	STD JPS	STS 3DE	T2IR 3DOD	VR 3DOT	AQA 3DPE	GAR 3DSR	MVU 3DQA	MEV PT	NIP RPM	TL SOT	TO 3DCR	VidCap 3DIR
OpenFlamingo [1]	22.3	25.5 25.0	25.8 21.5	24.6 25.5	21.6 25.0	25.0 14.5	28.2 13.5	34.5 15.5	49.0 27.5	14.5 4.0	19.0 25.5	13.5 23.0	22.5 7.0	17.5 22.1	26.0 3.0	39.0 1.5	49.0 26.5	20.0 22.0	27.5 **35.0**	10.0 17.0	13.5 28.5	16.5 20.5	30.0 23.5	20.0 11.5	18.7 31.0	24.5 25.0	22.5 23.5
XComposer2 [11]	21.9	24.0 55.0	21.0 35.0	10.8 42.5	5.8 22.5	0.0 2.5	0.0 19.0	34.2 20.0	24.0 8.0	14.5 15.5	2.5 45.0	23.0 0.0	63.5 0.0	19.0 20.6	26.0 0.0	14.5 16.5	31.0 0.0	9.5 7.0	28.5 0.0	31.5 4.5	59.5 0.0	44.0 33.5	30.0 63.0	4.5 1.5	15.5 38.5	12.0 42.0	66.0 33.0
Qwen-chat [2]	15.9	20.5 29.0	2.5 23.0	13.3 18.0	2.5 6.0	9.9 6.0	5.9 6.0	31.2 32.0	23.8 9.0	10.5 13.5	19.5 17.0	12.5 15.5	41.0 3.5	5.5 40.2	13.5 15.8	29.5 16.5	45.0 16.5	3.0 22.5	12.0 17.5	10.0 13.0	52.5 14.5	18.5 14.0	16.5 8.0	2.5 3.0	3.6 8.5	5.5 1.5	47.0 0.5
LLaVA-v1.5 [24]	19.2	14.1 24.5	4.2 17.5	13.7 40.0	5.8 15.0	1.9 21.5	6.9 4.0	27.3 26.0	35.0 7.5	6.5 26.5	12.5 17.5	12.5 5.0	53.0 4.5	10.0 25.6	25.5 27.1	66.5 8.5	43.0 8.0	19.0 4.0	3.5 6.0	2.5 6.0	23.5 14.5	36.5 29.5	12.0 66.0	16.5 2.0	6.7 35.0	7.0 34.5	28.0 28.5
ShareGPT4V [5]	18.5	16.4 26.5	5.0 19.0	10.8 42.0	6.2 7.5	9.0 14.0	2.7 7.5	34.2 31.5	28.5 7.0	4.5 29.0	10.5 18.0	3.5 5.0	57.0 1.5	4.0 28.1	12.5 23.3	55.5 9.5	44.5 3.0	13.5 7.0	5.0 6.0	5.0 2.0	26.0 8.0	38.0 27.5	14.0 65.5	15.5 0.0	10.9 44.0	6.0 36.5	25.0 31.0
LLaVA-interleave [20]	32.4	29.5 43.0	24.8 34.0	26.3 49.0	23.2 **29.5**	26.4 **32.0**	25.1 **26.0**	48.8 30.0	49.8 21.5	**23.5** 42.0	25.0 47.5	28.0 22.5	57.0 14.0	**21.5** 23.6	**33.0** 32.3	63.5 17.5	**54.5** **28.5**	25.0 23.0	26.0 17.5	24.0 3.0	27.0 31.0	49.5 36.0	29.0 **79.0**	23.0 **15.0**	25.4 **60.5**	**27.5** 34.5	32.5 42.5
InternVL1.5-chat [7]	37.4	63.7 **90.5**	**31.0** 35.5	22.6 56.5	20.3 23.5	16.3 31.0	28.3 24.5	**63.2** 53.0	38.5 26.0	21.0 40.0	**28.0** **49.0**	26.5 25.5	82.5 15.5	20.5 59.3	31.5 43.6	6.0 19.5	45.5 22.5	26.5 23.5	29.5 15.0	29.5 **33.5**	85.0 28.0	65.0 39.0	32.0 71.0	23.5 9.5	29.0 46.5	18.5 **50.5**	89.0 39.5
VideoChat2 [22]	35.0	46.8 54.0	27.5 **42.0**	31.6 **59.0**	23.6 23.0	25.6 30.5	**28.8** 23.0	45.3 44.5	54.3 **26.5**	20.5 **44.0**	25.5 36.5	25.5 25.0	64.0 **18.0**	21.0 38.6	31.0 **44.4**	31.5 21.0	50.0 26.5	21.0 24.0	**31.0** 13.0	**30.5** 0.0	73.0 **28.5**	51.0 **43.0**	31.5 65.5	23.5 11.5	21.8 58.0	24.0 36.0	81.5 35.0
VideoChat-TPO	**40.2**	**73.3** 59.0	24.3 39.5	**37.0** 56.5	**24.6** 27.5	**26.5** 29.5	26.9 21.0	45.0 **59.0**	**69.5** 25.0	20.5 **44.0**	23.5 48.5	**29.5** **27.5**	**83.0** 14.5	21.0 **73.4**	31.0 **44.4**	**92.5** **23.5**	49.5 27.5	**29.5** **24.5**	30.0 7.5	24.5 0.0	**88.0** 24.0	**67.5** 38.5	**34.5** 67.0	**29.5** 11.5	**36.8** 58.5	24.5 47.0	**94.5** **40.5**

Table 2. **Quantitative results of MMIU [30].** Accuracy is the metric, and the Overall score is computed across all tasks.

Model	Charades-STA [13]				MVBench
	R@0.3	R@0.5	R@0.7	mIoU	AVG
VideoChat-TPO	**58.3**	**40.2**	**18.4**	**38.1**	**66.8**
Only QVHighlight	54.8	34.6	15.1	35.8	66.5

Table 3. **Ablation task datasets.**

Model	MVBench	VideoMME	PerceptionTest
LLaVa-OV	56.7	58.2	57.1
LLaVa-OV-TPO	**64.8**(+8.1)	**61.3**(+3.1)	**64.0**(+6.9)

Table 4. **Performance of LLaVA-OV on Video Benchmarks.**

trained for efficiency. We adopt the AdamW optimizer [26] with the peak learning rate of 2e-5 and use cosine weight decay. The training involves a total batch size of 128 across 32 A100 GPUs. Since the purpose of stage 1 is to make MLLM identify tasks, we only use a small amount of data in this stage and adopt LLM loss so that LLM can generate task-specific tokens. For each task, we train the LLM with 50k examples to recognize the task. For training data, we use DiDeMo [15] and QuerYD [32] for temporal grounding task, RefCOCO [41], RefCOCOg [41] and RefCOCO+ [41] for spatial grounding task, and SAMv2 [35], MeViS [10] for segmentation task.

Settings of Stage 2. In stage 2, we add the task heads (*i.e.* temporal head, region head, and mask head) and learnable task tokens (temporal token, region token, and mask token). The objective of the second training stage is to learn the task head with preliminary functional capabilities. Therefore, we train LLM, task head and task token at this stage, and freeze vision encoder and connector.

In stage 2, the region head and token are trained with a learning rate of 2e-5 using a cosine learning rate scheduler. We use a two-layer MLP as region head to train from scratch and we use MSE loss for region head training. For training data, we use AS-V2 [38], Visual Genome [17], RefCOCO [41], RefCOCOg [41], RefCOCO+ [41] for one epoch with a total batch size of 128 to train region head and token.

We use a learning rate of 1e-4 for the temporal head and 2e-4 for the temporal token in stage 2. The temporal head is the same as CG-DETR [31] in structure, but we use the pre-trained InternVideo2 [39] to extract video features, while query features are extracted using the Chinese-Llama-Alpaca [8]. We use the same loss function in CG-DETR. We train the model on DiDeMo [15], QuerYD [32], HiRest [42], ActivityNet [3], TACoS [36], NLQ [14] for 25 epochs with a total batch size of 64.

For the mask head, we use the pre-trained SAM2 [35] model, replacing the prompt encoder of SAM2 with a single

Config	Stage 1	Stage 2	Stage 3 w/o Con.	Stage 3
Vision Enc. LR	Frozen	Frozen	2e-5	2e-5
Connector LR	Frozen	Frozen	2e-5	2e-5
Temporal Head LR	-	1e-4	2e-5	2e-5
Region Head LR	-	1e-4	2e-5	2e-5
Mask Head LR	-	Frozen	Frozen	Frozen
Mask Adapter LR	-	1e-4	2e-5	2e-5
Temporal Token LR	-	2e-4	2e-5	2e-5
Region Token LR	-	1e-4	2e-5	2e-5
Mask Token LR	-	1e-4	2e-5	2e-5
LLM LoRA LR	2e-5	2e-5	2e-5	2e-5
LR Schedule	Cosine Decay	Cosine Decay	Cosine Decay	Cosine Decay
Optimizer	AdamW [26]	AdamW [26]	AdamW [26]	AdamW [26]
Weight Decay	0.02	0.02	0.02	0.02
Input Resolution	224^2	224^2	224^2	224^2
Input Frames	16	16	16	16
LLM LoRA Rank	16	16	16	16
LLM LoRA Alpha	32	32	32	32
Warmup Ratio	0.2	0.2	0.2	0.2
Total Batch Size	128	64/128/128	256	256
Epoch	1	25/3/1	1	3
Numerical Precision	DeepSpeed bf16 [34]	DeepSpeed bf16 [34]	DeepSpeed bf16 [34]	DeepSpeed bf16 [34]

Table 5. **Training Settings of VideoChat-TPO.** Con. means conversation data and LR means learning rate.

Stage	Task	Samples	Datasets
Stage 1	Temporal Grounding	50K	DiDeMo [15], QuerYD [32]
	Spatial Grounding	50K	RefCOCO [41], RefCOCOg [41], RefCOCO+ [29]
	Segmentation	50K	SAMv2 [35], MeViS [10]
Stage 2	Temporal Grounding	116.5K	DiDeMo [15], QuerYD [32], HiRest [42], ActivityNet [3], TACoS [36], NLQ [14]
	Spatial Grounding	540.0K	AS-V2 [38], Visual Genome [17], RefCOCO [41], RefCOCO+ [41], RefCOCOg [29]
	Segmentation	114.6K	SAMv2 [35], MeViS [10]
Stage 3	Temporal Grounding	7.5K	QVHighlight [18]
	Spatial Grounding	400K	AS-V2 [38], Visual Genome [17], RefCOCO [41], RefCOCO+ [41], RefCOCOg [29]
	Segmentation	116.5K	MeViS [10], SAMv2 [35]
	Temporal Reasoning	40K	YouCook2 [9], ActivityNet [3]
	Conversation	3M	VideoChat2-IT [22], ShareGPT-4o [7], LLaVA-Hound-DPO [44], ShareGPT4V [4]

Table 6. **Datasets Used at Three Training Stages.** The temporal grounding task includes two subtasks: moment retrieval and highlight detection.

MLP layer called the mask adapter. During training, the mask token and adapter are trained with a learning rate of 2e-5, and the rest of SAM2 is frozen. We use MeViS [10] , SAMv2 [35] for three epochs in this stage with a total batch size of 128. We supplement the training data by expanding the ASv2 [38] image dataset into videos and adding it to this stage.

Settings of Stage 3. The third training stage aims to strengthen the model's conversational ability using TPO. This stage is divided into two parts. The first part involves training on a combined dataset of all tasks. The second part uses a dataset combining both task and conversation data. For conversatation data, we use VideoChat2-IT [22], ShareGPT-4o [7], LLaVA-Hound-DPO [44], ShareGPT4V [5] for instruction finetuning. We adopt a peak learning rate of 2e-5 for all the model in this stage and use a total batch size of 128.

Model	GPU	Stage1	Stage2	Stage3
VideoChat-TPO	64	0.5h	11h	52h
textualized task data	64	0.5h	10h	50h
only conversation data	64	-	-	42h

Table 7. **Training Cost of Three Stages on VideoChat.** Textualized task data means converting task data into conversation form.

TPO Additional Training and Inference Cost. From the data perspective, as can be seen from Table **??** in the Appendix, we have very little training data in the first (around 0.15M) and second stages (around 0.7M), most of the data (around 3.5M) is used in third phase of the experiment. Among the data in the third stage, most of it is conversation data for fine-tuning MLLM. Therefore, TPO introduces little new data. Concerning training cost, according Table 7, when using the same amount of data, the training time of our TPO method and the autoregressive method is almost the same, and compared with the version without

visual task, the TPO method increases the training cost by about 25%.

The Temporal Head and the Mask Head contains additional encoders. In training phase, the additional encoders are frozen, and we use the features extracted by the encoder for training. In inference phase, the additional encoders are only used when the task head is activated. When only performing conversation tasks, no additional inference cost is incurred.

Template Details. To support the proper invocation of task-specific decoders, we construct a series of instruction templates for different tasks and use them as instruction tuning data for MLLM. We comprehensively list all the instruction templates below, in Table 8, 9, 10, and 11.

3. Qualitative Results

We evaluate VideoChat-TPO on various visual perception tasks and display the visualizations from Figure 1 to Figure Figure 4. In addition, we also show the results of multimodal video understanding in Figure 5.

Spatial Grounding. In Figure 1, we show the spatial grounding visualizations. VideoChat-TPO can infer the target object from the description of natural language and locate it. Our VideoChat-TPO can accurately locate the target among multiple similar objects. Even if the target object is occluded or in the background area, it can still be accurately located.

Referring Segmentation. We show the visualizations of the referring segmentation in Figure 2. VideoChat-TPO can delinear the target object in the video according to user input in complex scenes. Furthermore, VideoChat-TPO can separate the target object from multiple objects of the same kind according to the description of appearance or action characteristics indicated by the user.

Tracking. The tracking visualizations are shown in Figure 3. The user needs to include the bounding box coordinate information of the first frame of the tracked target in the video in the input. The visualizations show that when the target object is partially occluded in the video, it can still be tracked. Even if the target object is out of the camera's view, our VideoChat-TPO can still track it when it appears in subsequent frames.

Moment Retrieval and Highlight Detection. The visualizations of the moment retrieval and highlight detection are given in Figure 4. Our VideoChat-TPO can infer the results and target events based on the user's questions, and perform moment retrieval and highlight detection on the target events.

Multimodal Video Understanding. The multimodal video understanding visualizations are shown in Figure 5. Our VideoChat-TPO achieve decent results in fine-grained action description, spatial description, and video captioning.

CVPR
#2574

1. Localize the visual content described by the given textual query $\langle query \rangle$ in the video, and output the start and end timestamps in seconds.
2. Detect and report the start and end timestamps of the video segment that semantically matches the given textual query $\langle query \rangle$.
3. Locate and describe the visual content mentioned in the text query $\langle query \rangle$ within the video, including timestamps.
4. The given natural language query $\langle query \rangle$ is semantically aligned with a video moment, please give the start time and end time of the video moment.
5. Find the video segment that corresponds to the given textual query $\langle query \rangle$ and determine its start and end seconds.

Table 8. **Instructions for Temporal Grounding.**

1. Track the object in the video using a box with initial coordinates $\langle track\_box \rangle$.
2. Use a bounding box with coordinates $\langle track\_box \rangle$ to follow the movement of the moving object in the visual input.
3. Given an initial bounding box with coordinates $\langle track\_box \rangle$, track the motion of the target object in the sequence of frames.
4. Starting from the box defined by the coordinates $\langle track\_box \rangle$, monitor the movement of the object in the video.
5. Utilizing the initial box specified by the coordinates $\langle track\_box \rangle$, continuously track and update the location of the object in the video stream.
6. Given a video with an object of interest enclosed in a bounding box with coordinates $\langle track\_box \rangle$, generate a sequence of bounding boxes that track the object's movement.
7. With an initial box defined by $\langle track\_box \rangle$, trace the object's trajectory by generating a sequence of bounding boxes that follow the object's movement in the visual input.
8. Apply an object tracking algorithm to a video, starting with a bounding box defined by $\langle track\_box \rangle$.
9. Given a video and an initial bounding box defined by $\langle track\_box \rangle$, track the movement of the object within the video.
10. Starting from an initial box defined by $\langle track\_box \rangle$, track the movement of the object in the visual input.

Table 9. **Instructions for Tracking.**

1. Where is $\langle expr \rangle$?
2. Can you find $\langle expr \rangle$?
3. Can you detect $\langle expr \rangle$?
4. Can you locate $\langle expr \rangle$?
5. Please find $\langle expr \rangle$.
6. Please detect $\langle expr \rangle$?
7. Please locate $\langle expr \rangle$?
8. Find $\langle expr \rangle$.
9. Detect $\langle expr \rangle$?
10. Locate $\langle expr \rangle$?

Table 10. **Instructions for Spatial Grounding.**

1. Please give the motion path of $\langle obj \rangle$ in the video over time.
2. Show the tracking trajectory of $\langle obj \rangle$'s movement through the scene in the video.
3. Please generate a motion path of $\langle obj \rangle$'s movement in the video, highlighting its tracking trajectory.
4. Show the tracking trajectory of $\langle obj \rangle$.
5. Generate $\langle obj \rangle$'s tracking trajectory.
6. Visualize the tracking trajectory of $\langle obj \rangle$ in the video.
7. Please generate a visual representation of $\langle obj \rangle$'s movement in the video, highlighting its tracking trajectory.

Table 11. **Instructions for Referring Segmentation.**

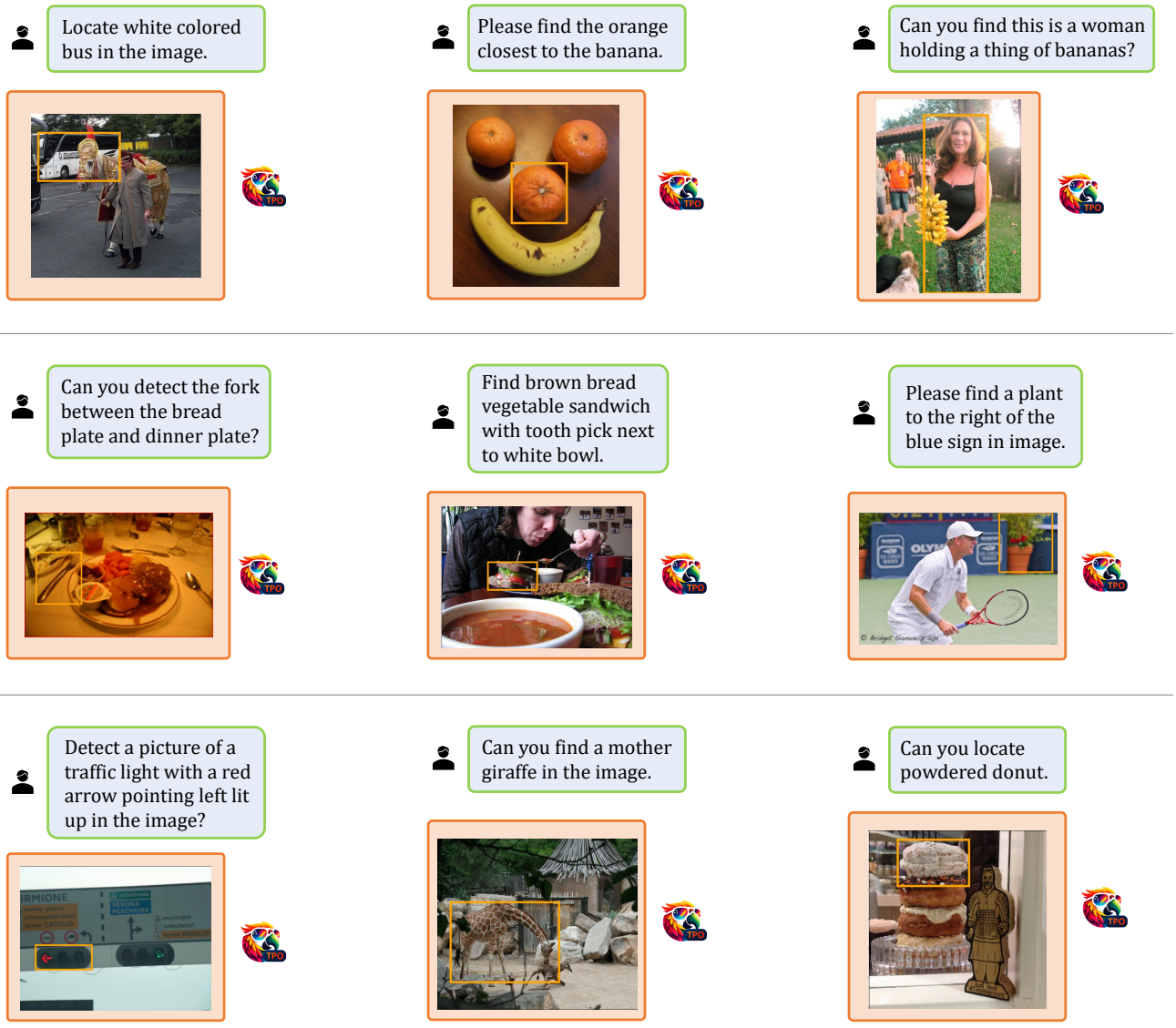

Figure 1. **Qualitative Results of Spatial Grounding.**

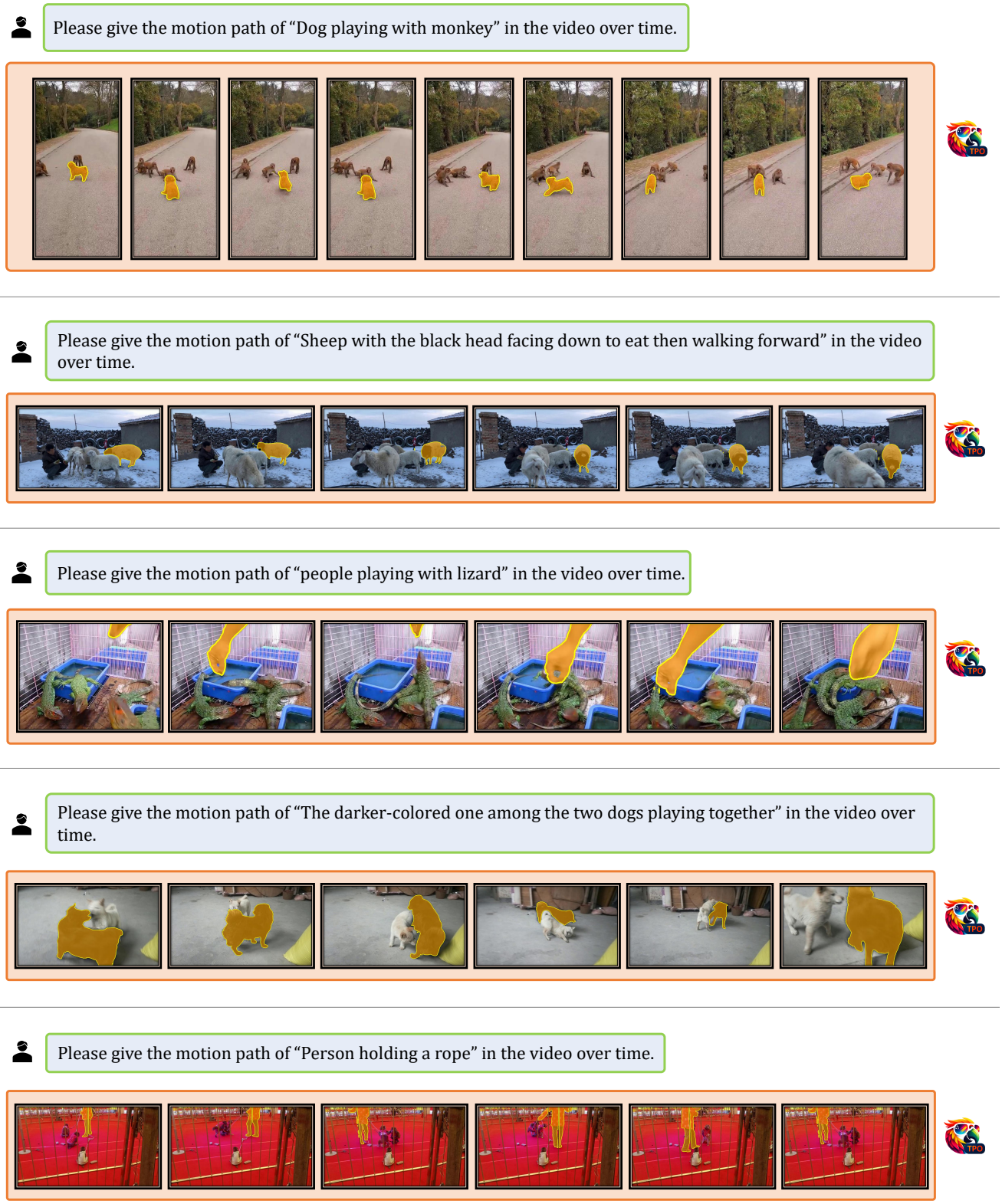

Figure 2. **Qualitative Results of Referring Segmentation.**

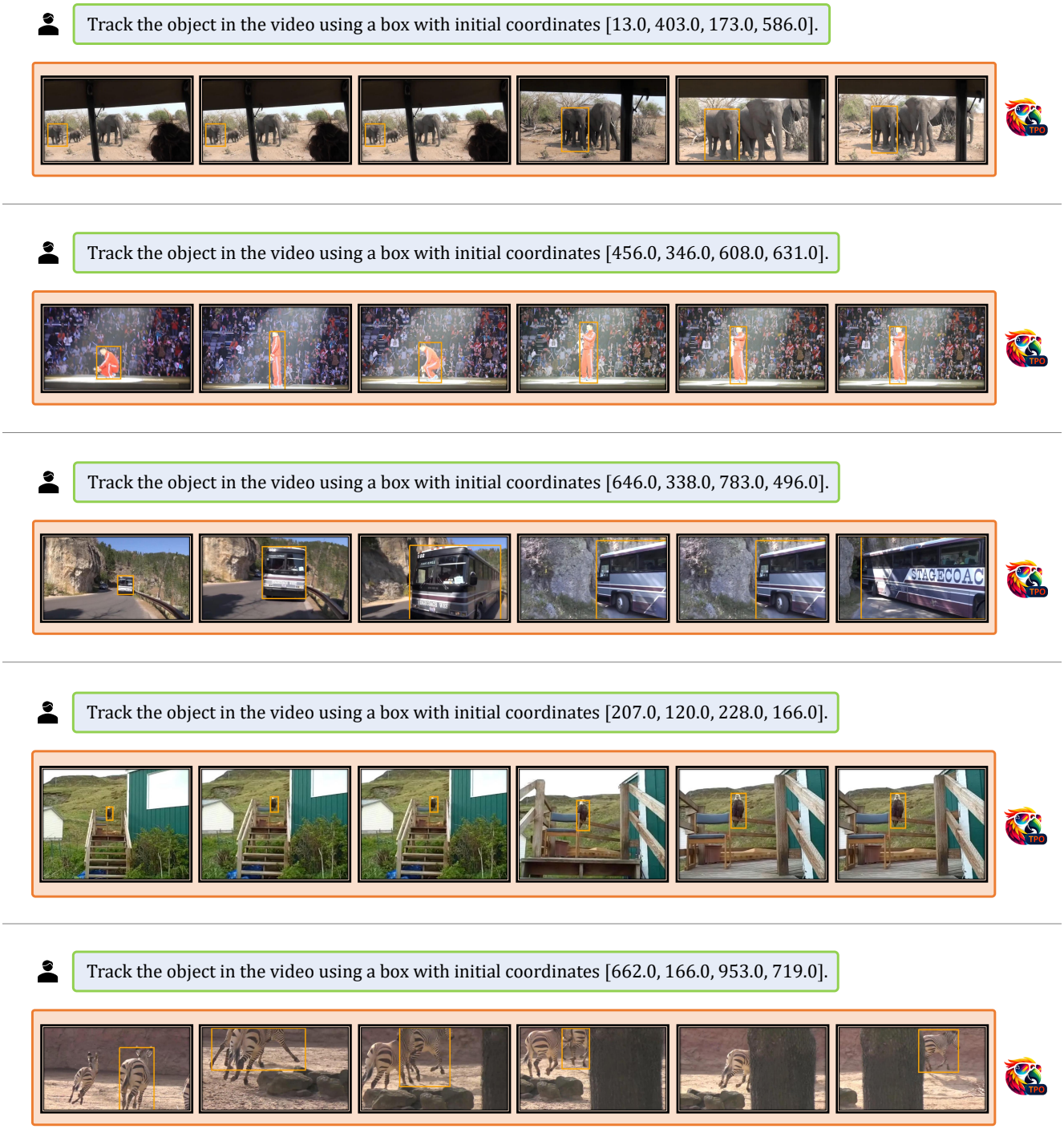

Figure 3. **Qualitative Results of Tracking.**

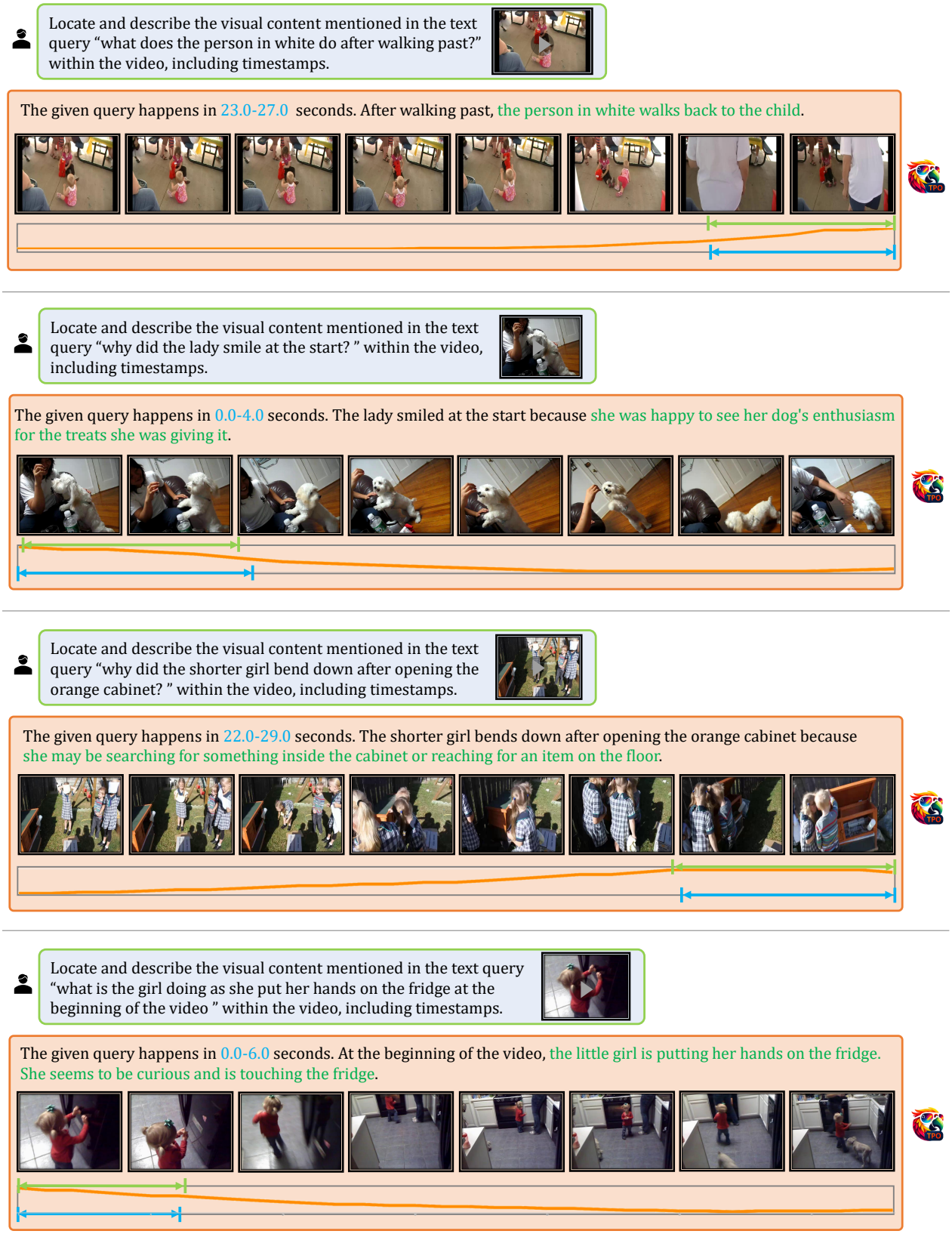

Figure 4. **Qualitative Results of Moment Retrieval and Highlight Detection.** The orange curve represents saliency score, the blue interval represents the time interval predicted by the model, and the green interval represents the ground truth.

Describe the cat's actions in the video. (fine-grain action)

The video shows a cat walking down a street in the rain. The cat is seen walking on the wet ground and passing by various objects such as a blue trash can, a brown cardboard box, and a pile of garbage. The cat is also seen walking past a pile of garbage bags and a pile of garbage. The video captures the cat's movements and the surrounding environment.

Describe the video in detail. (caption)

The video shows a bird's eye view of a city with a large body of water in the background. The camera pans over the city, showing various buildings and structures, including a large building with a dome, a tall building with a clock tower, and a large building with a dome. The camera also captures a view of the ocean and a rocky cliff. The video provides a detailed view of the city's architecture and the surrounding natural beauty.

Which cup does the lid go under, the left one, the middle one, or the right one? (spatial perception)

The lid goes under the middle cup.

Describe the video in detail. (caption)

The video shows a person riding a bike down a dirt path in the woods. The camera is mounted on the bike, and we can see the rider's legs and feet as they pedal. The path is surrounded by trees, and we can see the shadows of the trees on the ground. The rider is wearing a helmet, and we can see the reflection of the trees in the helmet. The video is shot during the day, and we can see the sunlight shining through the trees. The rider is wearing a colorful shirt, and we can see the colors of the shirt as they ride. Overall, the video shows a peaceful and scenic bike ride through the woods.

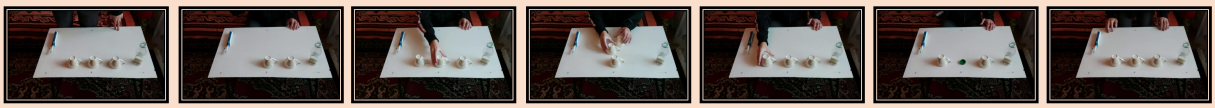

Figure 5. **Qualitative Results of Multimodal Video Understanding.**

References

[1] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023. 2

[2] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 2

[3] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, pages 961–970, 2015. 2, 3

[4] Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*, 2023. 3

[5] Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. In *ECCV*, 2024. 2, 3

[6] Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, Li Yuan, Yu Qiao, Dahua Lin, Feng Zhao, and Jiaqi Wang. Sharegpt4video: Improving video understanding and generation with better captions. *arXiv preprint arXiv:2406.04325*, 2024. 2

[7] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *CVPR*, pages 24185–24198, 2024. 2, 3

[8] Yiming Cui, Ziqing Yang, and Xin Yao. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*, 2023. 2

[9] Pradipto Das, Chenliang Xu, Richard F Doell, and Jason J Corso. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In *CVPR*, pages 2634–2641, 2013. 3

[10] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, and Chen Change Loy. MeViS: A large-scale benchmark for video segmentation with motion expressions. In *ICCV*, 2023. 2, 3

[11] Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Xilin Wei, Songyang Zhang, Haodong Duan, Maosong Cao, et al. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *arXiv preprint arXiv:2401.16420*, 2024. 2

[12] Chaoyou Fu, Yuhan Dai, Yondong 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. 1

[13] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, pages 5267–5275, 2017. 2

[14] 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*, pages 18995–19012, 2022. 2, 3

[15] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017. 2, 3

[16] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022. 1

[17] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123:32–73, 2017. 2, 3

[18] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. *NeurIPS*, 34:11846–11858, 2021. 1, 3

[19] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *CoRR*, abs/2408.03326, 2024. 1

[20] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024. 2

[21] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 2

[22] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *CVPR*, pages 22195–22206, 2024. 1, 2, 3

[23] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023. 2

[24] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2024. 2

[25] Ruyang Liu, Chen Li, Haoran Tang, Yixiao Ge, Ying Shan, and Ge Li. St-llm: Large language models are effective temporal learners. In *ECCV*, 2024. 2

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2, 3

[27] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Videogpt+: Integrating image and video encoders for enhanced video understanding. *arXiv preprint arXiv:2406.09418*, 2024. 2

[28] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video

understanding via large vision and language models. In *ACL*, 2024. 2

[29] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, pages 11–20, 2016. 3

[30] Fanqing Meng, Jin Wang, Chuanhao Li, Quanfeng Lu, Hao Tian, Jiaqi Liao, Xizhou Zhu, Jifeng Dai, Yu Qiao, Ping Luo, et al. Mmiu: Multimodal multi-image understanding for evaluating large vision-language models. *arXiv preprint arXiv:2408.02718*, 2024. 2

[31] WonJun Moon, Sangeek Hyun, SuBeen Lee, and Jae-Pil Heo. Correlation-guided query-dependency calibration in video representation learning for temporal grounding. *arXiv preprint arXiv:2311.08835*, 2023. 2

[32] Andreea-Maria Oncescu, Joao F Henriques, Yang Liu, Andrew Zisserman, and Samuel Albanie. Queryd: A video dataset with high-quality text and audio narrations. In *ICASSP*, pages 2265–2269. IEEE, 2021. 2, 3

[33] Viorica Patraucean, Lucas Smaira, Ankush Gupta, Adria Recasens, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Mateusz Malinowski, Yi Yang, Carl Doersch, et al. Perception test: A diagnostic benchmark for multimodal video models. 2024. 1

[34] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *SIGKDD*, pages 3505–3506, 2020. 3

[35] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 2, 3

[36] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics*, 1:25–36, 2013. 2, 3

[37] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. *CVPR*, abs/2312.02051, 2024. 2

[38] Weiyun Wang, Yiming Ren, Haowen Luo, Tiantong Li, Chenxiang Yan, Zhe Chen, Wenhai Wang, Qingyun Li, Lewei Lu, Xizhou Zhu, et al. The all-seeing project v2: Towards general relation comprehension of the open world. *arXiv preprint arXiv:2402.19474*, 2024. 2, 3

[39] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Jilan Xu, Zun Wang, et al. Internvideo2: Scaling video foundation models for multimodal video understanding. In *ECCV*, 2024. 2

[40] Lin Xu, Yilin Zhao, Daquan Zhou, Zhijie Lin, See Kiong Ng, and Jiashi Feng. Pllava: Parameter-free llava extension from images to videos for video dense captioning. *arXiv preprint arXiv:2404.16994*, 2024. 2

[41] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *ECCV*, pages 69–85. Springer, 2016. 2, 3

[42] Abhay Zala, Jaemin Cho, Satwik Kottur, Xilun Chen, Barlas Oguz, Yashar Mehdad, and Mohit Bansal. Hierarchical video-moment retrieval and step-captioning. In *CVPR*, pages 23056–23065, 2023. 2, 3

[43] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 2

[44] Ruohong Zhang, Liangke Gui, Zhiqing Sun, Yihao Feng, Keyang Xu, Yuanhan Zhang, Di Fu, Chunyuan Li, Alexander Hauptmann, Yonatan Bisk, et al. Direct preference optimization of video large multimodal models from language model reward. *arXiv preprint arXiv:2404.01258*, 2024. 3