

Thinking in Dynamics: How Multimodal Large Language Models Perceive, Track, and Reason Dynamics in Physical 4D World

Supplementary Material

001	This is the supplementary material for the paper: “ <i>Thinking in Dynamics: How Multimodal Large Language Models Perceive, Track, and Reason Dynamics in the Physical 4D World</i> .” It provides additional implementation details, visualizations, and prompt templates to complement the main paper. The contents are organized as follows:	
002		
003		
004		
005		
006		
007	• In Section A , we detail the implementation of our framework, including the data filter strategy for dynamic video curation and the construction of the <i>Spatio-Temporal Textual Cognitive Map (ST-TCM)</i> for object-centric reasoning.	
008		
009		
010		
011		
012	• In Section B , we present qualitative visualizations from Dyn-Bench, including examples across hierarchical levels, representative failure cases, and comparisons highlighting the effects of <i>ST-TCM</i> and <i>Mask-Guided Input</i> .	
013		
014		
015		
016		
017	• In Section C , we describe six prompt templates used in the QA generation stage for creating Visual Question Answering (VQA) and grounding data across dynamic perception and reasoning tasks.	
018		
019		
020		
021	A. More Implementation Details	
022	A.1. Data Filter Strategy	
023	The goal of our video data filtering process is to identify videos that exhibit rich spatio-temporal dynamics, including object and camera motion, inter-object interactions, human activities, and sequential manipulations, while excluding static, artificial, or heavily edited content such as landscapes, cartoons, or composited clips.	
024		
025		
026		
027		
028		
029	To ensure that the curated dataset effectively supports <i>spatio-temporal reasoning</i> and <i>dynamic object grounding</i> , we define three key selection criteria:	
030		
031		
032	S1. High-quality video content. Videos should maintain sufficient visual fidelity in terms of resolution, frame rate, and perspective stability, without severe distortion or over-processing artifacts. Shaky frames or inconsistent frame rates should also be excluded.	
033		
034		
035		
036		
037	S2. Feasibility for motion and geometry reasoning. Videos should be suitable for reliable geometric estimation and motion reasoning. We exclude clips with extreme zooming, abrupt shot transitions, or ambiguous reference frames, such as those captured from moving vehicles. Videos that lack stable visual correspondences, including those with heavily blurred or fully occluded backgrounds, are also removed from the dataset.	
038		
039		
040		
041		
042		
043		
044		
045		
	S3. Dynamic camera and scene motion. Videos should contain non-static cameras or dynamically evolving scenes that allow non-trivial spatio-temporal reasoning. Such videos often capture human-object or inter-object interactions, providing diverse motion for downstream reasoning and grounding tasks.	046 047 048 049 050 051
	To meet these requirements, we design a multi-stage filtering pipeline integrating motion, geometry, dynamic coverage, and multimodal semantics. Each stage progressively refines dataset quality to ensure geometric consistency and temporal richness.	052 053 054 055 056
	S4. Compute motion features. For each video, we extract low-level motion and quality statistics, including blur degree, frame rate (fps), number of I-frames, motion vector magnitude, and motion vector variance, using OpenCV. These features characterize the global motion intensity and temporal smoothness of the video, serving as primary indicators of scene dynamics.	057 058 059 060 061 062 063
	S5. Obtain geometric features. Using VGGT [9] and UniDepth-V2 [7], we estimate per-frame camera intrinsics, extrinsics, and depth maps to evaluate geometric stability across time. We further assess depth continuity, focal-length stability, and camera motion smoothness to ensure temporally coherent geometry. For 2D datasets without ground-truth geometry, these models also reconstruct camera pose and depth to provide metric-scale estimation.	064 065 066 067 068 069 070 071 072
	S6. Dynamic object coverage estimation. We estimate the overall degree of scene dynamics by measuring the proportion and spatial distribution of moving objects across frames, which captures both temporal motion diversity and spatial activity density. For 4D datasets lacking instance segmentation masks, Qwen2.5-VL [1] and Sa2VA [12] are employed to infer or refine missing masks, ensuring complete and consistent instance-level motion representation throughout the sequence.	073 074 075 076 077 078 079 080 081
	S7. Multimodal LLM feature extraction. A multimodal large language model (MLLM), such as Qwen2.5-VL [1], is used to extract semantic and motion-aware features. The model answers 26 structured diagnostic questions (see Fig. 1), capturing motion patterns, interactions, and causal relations among entities.	082 083 084 085 086 087
	S8. Video quality scoring via Random Forest. We aggregate all extracted features, comprising 31 dimensions from the preceding modules, and use them as input to a random forest regressor predicting a continuous <i>video dynamism score</i> between 0 and 5. A total of 3,000	088 089 090 091 092

LLM Feature extraction question list

Quesiton List

Is the camera capturing the scene static? Reply yes or no.
 Is the camera capturing the scene moving? Reply yes or no.
 Is the video capturing a landscape? Reply yes or no.
 Is the video capturing a static scene? Reply yes or no.
 Is the scene captured from a distance? Reply yes or no.
 Is the video captured with a drone? Reply yes or no.
 Is the video computer-generated? Reply yes or no.
 Is the video content abstract? Reply yes or no.
 Is there something moving through the scene? Reply yes or no.
 Is there someone doing something in the video? Reply yes or no.
 Are there several things moving in the video? Reply yes or no.
 Is there an object that is being manipulated? Reply yes or no.
 Are there animals in the video? Reply yes or no.
 Is the scene mostly static? Reply yes or no.
 Are things occluding each other in this video? Reply yes or no.
 Is there something obstructing the view apart from the watermark? Reply yes or no.
 Is there a large number of things in the video? Reply yes or no.
 Are there more than 5 different objects in the video? Reply yes or no.
 Is it hard to keep track of some entities because they are moving so much? Reply yes or no.
 Is someone looking at a phone, a tablet or a computer screen? Reply yes or no.
 Are they looking at a phone, a tablet or a computer screen during the whole video? Reply yes or no.
 Are there several moving persons in this video? Reply yes or no.
 Are there several moving animals in this video? Reply yes or no.
 Are there several objects in this video? Reply yes or no.
 Are there several similar-looking objects in the video? Reply yes or no. Do they look similar? Reply yes or no.

Figure 1. LLM Feature Extraction: the model answers 26 structured diagnostic questions to extract semantic and motion cues.

093 videos are manually annotated for training, where 0
 094 represents static scenes and 5 corresponds to highly dy-
 095 namic, temporally complex motion. The random forest
 096 jointly leverages motion, geometric, and semantic cues
 097 to estimate video-level dynamism.
 098 **S9. VLM-assisted refinement.** After filtering by predicted
 099 scores, we employ a Vision-Language Model (VLM) to
 100 further assess semantic coherence, realism, and motion
 101 validity. Only videos passing both quantitative and se-
 102 mantic evaluations are retained as the final high-quality
 103 dynamic dataset.

104 This hierarchical filtering pipeline ensures that the result-
 105 ing dataset maintains visual fidelity, geometric stability, and
 106 motion diversity, providing robust supervision for *spatio-*
 107 *temporal reasoning* and *dynamic object grounding* in the
 108 proposed Dyn-Bench.

109 A.2. Spatio-Temporal Textual Cognitive Map (ST- 110 TCM)

111 To model fine-grained object motions and interactions in
 112 dynamic scenes, we construct a *Spatio-Temporal Textual*
 113 *Cognitive Map (ST-TCM)* for each filtered video. The ST-
 114 TCM provides a unified representation that bridges geo-
 115 metric perception, temporal reasoning, and linguistic ab-

116 straction. It encodes 3D object trajectories, camera motion,
 117 and inter-object relations into structured textual descrip-
 118 tions, enabling object-centric spatio-temporal understand-
 119 ing for multimodal reasoning. Inspired by recent structured
 120 spatio-temporal representations and cognitive mapping ap-
 121 proaches [3, 6, 11], we design the ST-TCM to capture dy-
 122 namic interactions through both geometric grounding and
 123 textual abstraction, facilitating interpretable reasoning over
 124 temporal evolution in complex scenes.

S1. Geometric and motion reconstruction. Each video is
 125 processed at 6 FPS with synchronized RGB-D frames
 126 and instance segmentation masks to ensure balanced
 127 temporal sampling and spatial fidelity. To recover
 128 metric-scale geometry and accurate camera poses, we
 129 employ VIPE [4], which jointly estimates per-frame
 130 depth \hat{D}_t and camera transformation $\hat{T}_t = [\hat{R}_t | \hat{t}_t]$ from
 131 monocular sequences under scale-aligned supervision.
 132 The 3D position of each object o_i is computed by pro-
 133 jecting its instance centroid $\tilde{\mathbf{u}}_t^i$ into world coordinates
 134 as $\mathbf{p}_t^i = \hat{T}_t^{-1} K^{-1} \tilde{\mathbf{u}}_t^i \hat{D}_t(\tilde{\mathbf{u}}_t^i)$, where K denotes the
 135 camera intrinsic matrix. Temporal differencing yields
 136 object velocity $\mathbf{v}_t^i = (\mathbf{p}_t^i - \mathbf{p}_{t-1}^i) / \Delta t$ and acceleration
 137 $\mathbf{a}_t^i = (\mathbf{v}_t^i - \mathbf{v}_{t-1}^i) / \Delta t$. To mitigate frame-level depth
 138 noise and maintain temporal coherence, an exponential
 139

moving average is applied for smoothing. This stage yields geometrically consistent and temporally stable trajectories, serving as the foundation for subsequent motion reasoning and relational inference.

S2. Dynamic relation and spatial reasoning. Building on reconstructed trajectories, we model both inter-object and camera-object relations according to spatial proximity, relative motion, and temporal continuity. For each object pair (o_i, o_j) , we compute their 3D Euclidean distance and evaluate a relative motion rate derived from their velocity vectors and positional difference, which distinguishes *approaching*, *receding*, and *parallel* motion behaviors. Additionally, each object’s spatial configuration relative to the camera is characterized by its azimuth and elevation angles, derived from the relative 3D coordinates, providing directional cues for qualitative reasoning such as *front*, *left*, *right*, or *back*. This unified formulation captures both egocentric and allocentric spatial layouts, enabling consistent modeling of spatial dependencies across frames.

S3. Textual cognitive mapping. All geometric, kinematic, and relational attributes are systematically transformed into structured textual form through a rule-based cognitive mapping module. This process integrates multi-level cues, including object geometry (size and position), motion states (direction, velocity, acceleration), and relational context (distance, orientation, and interaction type), into concise, interpretable frame-level textual representations. These frame-wise textual embeddings are temporally aligned and sequentially aggregated, forming a coherent spatio-temporal narrative that maintains motion continuity and object identity throughout the sequence. The resulting *Spatio-Temporal Textual Cognitive Map (ST-TCM)* provides a unified symbolic-textual representation of scene dynamics, which is subsequently utilized as structured input to *Qwen3-VL-235B* [10] for object-centric reasoning, dynamic scene understanding, and spatio-temporal question answering.

B. More Visual Results

In this section, we present additional qualitative results to further illustrate the effectiveness of our framework. Comprehensive visualizations are provided across the three hierarchical levels and nine sub-tasks defined in *Dyn-Bench*, illustrating the diversity of task settings and dynamic scene types encompassed by the benchmark. We also include representative failure cases to analyze common challenges in dynamic scene understanding. Furthermore, comparative visual examples are shown to evaluate the impact of the proposed *Spatio-Temporal Textual Cognitive Map* and *Mask-Guided Input* on different multimodal large language models, highlighting their respective strengths and limitations.

B.1. Qualitative Examples on *Dyn-Bench*

We provide qualitative examples across the three hierarchical levels of *Dyn-Bench*, including *Dynamic Inter-Object Perception*, *Dynamic Object-Scene Tracking*, and *Dynamic Camera-Object Reasoning* (see Fig. 2). Each level captures distinct dimensions of dynamic scene understanding: the first focuses on relational reasoning between dynamic objects, the second emphasizes object-scene interactions and compositional motion, and the third involves reasoning under varying camera motion and viewpoint changes. Within these levels, multiple sub-tasks are defined, covering nine representative types of dynamic understanding scenarios such as temporal changes, spatial alignment, activity recognition, and cross-object motion dynamics. Together, these examples highlight the diversity and complexity of task settings in *Dyn-Bench*, illustrating how the benchmark provides a comprehensive evaluation framework for assessing visual reasoning under dynamic and multi-object conditions. They also provide qualitative insights into how models interpret temporal cues, preserve semantic coherence, and adapt to scene variations in realistic videos.

B.2. Failure Cases on *Dyn-Bench*

We present failure cases from three representative categories of multimodal large language models (MLLMs), namely *Gemini-2.5 Pro* [2] as a general MLLM, *UniPixel-7B* [5] as a region-level MLLM, and *VST-7B-RL* [8] as a spatial MLLM, evaluated across the nine sub-tasks of *Dyn-Bench*. These cases reveal the typical reasoning and perception errors encountered when handling complex dynamic scenes, such as inaccurate temporal correlation, misalignment in spatial grounding, or failure to infer cross-object motion relationships. In addition, we provide qualitative comparisons using *ST-TCM* and *Mask-Guided Inputs* to demonstrate their effects in enhancing temporal consistency, spatial focus, and reasoning robustness across different MLLM architectures. Representative visual examples are shown in Fig. 3–8, illustrating the limitations of existing MLLMs and the benefits of the proposed enhancements for dynamic reasoning.

C. Prompt Templates for Data Generation

We introduce six prompt templates used in the Question-Answer (QA) generation stage for creating both Visual Question Answering (VQA) and grounding pairs. Each prompt is provided as input to the *Qwen3-VL-23B* model to elicit diverse, context-aware, and semantically consistent responses. The six templates correspond to *Dynamic Inter-Object Perception*, *Dynamic Object-Scene Tracking*, *Dynamic Camera-Object Reasoning*, *Dynamic Inter-Object Grounding*, *Dynamic Object-Scene Grounding*, and *Dynamic Camera-Object Grounding*, as shown in Fig. 9–14.

**Object-Camera: temporal and visual changes**

Question:

During the middle of the video ($t \approx 1.4s$), is the camera above, level with, or below the black and white dog lying on the grass, and is the relative altitude becoming higher, lower, or stable?

- A. Camera is above the black and white dog and becoming higher
- B. Camera is level with the black and white dog and becoming lower
- C. Camera is below the black and white dog and becoming higher
- D. Camera is above the black and white dog and becoming lower

Answer: D

Object-Camera: camera motion and orientation

Question:

Between the start ($t=0s$) and the middle ($t=1.4s$), what is the camera's dominant translation direction?

- A. Rightward and slightly forward
- B. Leftward and slightly backward
- C. Downward and forward
- D. Upward and backward

Answer: B

Object-camera: camera-object interaction

Question:

The black and white dog appears to get smaller in the frame over time. Which explanation best fits this observation?

- A. The camera is zooming out (FOV change)
- B. The camera is translating backward (distance change)
- C. Both zooming out and translating backward are occurring
- D. Neither zooming nor backward translation explains the size change

Answer: B

Inter-object: activity and object description

Question:

How many objects are chased by the dog in the second second of the video?

- A. four
- B. three
- C. two
- D. One

Answer: A

Inter-object: movement and temporal dynamics

Question:

Between the start and the middle of the video, how does the position of the lead goose change relative to the black dog?

- A. It moves from right to left across the dog's path
- B. It moves from behind the dog to in front of it
- C. It moves from in front of the dog to behind it
- D. It moves from left to right across the dog's path

Answer: A

Inter-object: spatial relationships and change

Question:

At the 1.5-second mark, what is the approximate distance between the black dog and the lead goose?

- A. 1.5 meters
- B. 3.0 meters
- C. 5.0 meters
- D. 7.5 meters

Answer: B

Object-scene: movement patterns and trajectories

Question:

How does the dog's position change relative to the ducks as the scene progresses?

- A. The dog starts far away and gradually moves closer to the ducks
- B. The dog remains stationary while the ducks move around it.
- C. The dog initially lies down near the ducks, then rises and moves behind them
- D. The dog chases the ducks across the field, ending up far ahead of them

Answer: C

Object-scene: spatial relationships and composition

Question:

How does the spatial relationship between the white pickup truck and the moving animals change over time?

- A. The truck moves closer to the animals as they advance
- B. The animals move away from the truck, increasing the distance between them
- C. The truck remains stationary while the animals move across the field, changing their relative positions
- D. The truck drives away, leaving the animals alone in the field

Answer: C

Object-scene: scene focus and dynamics


Question: How many objects can be seen in the scene?

- A. four
- B. five
- C. two
- D. three

Answer: B

Figure 2. Qualitative examples of the nine representative dynamic understanding tasks across the three hierarchical levels of Dyn-Bench.

Object-scene: spatial relationships and composition



Example 1: Without the textual map 😞

Prompt: Given the <raw video> 🎥 How does the positioning of the fighters change relative to the edge of the boxing ring as the scene unfolds? (GT: The fighters gradually approached the edge of the boxing ring.)

Gemini-2.5-pro: As the scene unfolds, the fighters are seen moving around within the boxing ring. The crowds are positioned outside the ropes. The fighters do not move towards the boxing ring directly, nor do they retreat to the center away from the ropes. Therefore, the fighters operate within their designated space, maintaining a consistent distance from the edge of the boxing ring.

Answer: It maintains a consistent distance from the edge of the boxing ring. (Wrong) ❌


Example 2: With the textual map 😊

Prompt: Given the <raw video> 🎥 and the <textual map> 🗺️ How does the positioning of the fighters change relative to the edge of the boxing ring as the scene unfolds?(GT: The fighters gradually approached the edge of the boxing ring.)

Gemini-2.5-pro: The fighters' positions change dynamically as they move across the ring's canvas. Initially positioned on the right side, the action shifts as one fighter pressures the other. The fighter in the red headgear is gradually pushed back, this movement brings both combatants much closer to the edge of the boxing ring.

Answer: Getting closer to the edge of the boxing ring. (Correct) ✅

Object-scene: movement patterns and trajectories



Example 1: Without the textual map 😞

Prompt: Given the <raw video> 🎥 What is the estimated average speed of the rider throughout the entire video process? (GT:13m/s)

Gemini-2.5-pro: By analyzing the distance the rider covers relative to the track features between frames, an average speed can be estimated. The rider appears to travel several bike-lengths per frame, maintaining an impressive average speed of approximately 4 meters per second through the sequence.

Answer: 4m/s (wrong) ❌


Example 2: With the textual map 😊

Prompt: Given the <raw video> 🎥 and the <textual map> 🗺️ What is the estimated average speed of the rider throughout the entire video process? (GT:13m/s)

Gemini-2.5-pro: The estimated average speed of the rider throughout the entire video process is approximately 13.3 meters per second. This calculation considers the instantaneous speeds provided for the rider at various points in the video, ranging from stationary to much higher speeds during moments of acceleration or significant movement. By averaging these recorded speeds, we get a general sense of the rider's typical velocity over the duration shown.

Answer: 13.3m/s (Correct) ✅

Object-scene: scene focus and dynamics



Example 1: Without the textual map 😞

Prompt: Given the <raw video> 🎥 How does the silver car's position change relative to the crosswalk as the scene progresses? (GT: The car is moving towards the crosswalk first and then remains stationary.)

Gemini-2.5-pro: Initially, the silver car is positioned just before the pedestrian crosswalk at the corner of the intersection. As the scene progresses, the car accelerates and drives forward, moving directly onto and across the white lines of the crosswalk. It continues making a left turn through the intersection, so that by the end of the sequence, the vehicle has completely passed the crosswalk and is moving away from it.

Answer: Approach first, then move away. (Wrong) ❌

Example 2: With the textual map 😊


Prompt: Given the <raw video> 🎥 and the <textual map> 🗺️ How does the silver car's position change relative to the crosswalk as the scene progresses? (GT: The car is moving towards the crosswalk first and then remains stationary.)

Gemini-2.5-pro: Initially, the silver car is positioned in the distance, approaching the intersection from the right. It moves forward and turns left, gradually getting closer to the crosswalk located on the left side of the frame. The car then slows down and comes to a complete stop at the edge of the intersection, with its front bumper positioned just before the white stripes of the crosswalk, where it remains stationary.

Answer: Approach first and then remain unchanged. (Correct) ✅

Figure 3. Representative failure cases of Gemini-2.5 Pro on Dyn-Bench, showing qualitative comparisons incorporating ST-TCM.

Inter-object: activity and object description



Example 1: Without the textual map 😞

Prompt: Given the <raw video> [video icon] How many objects are chased by the dog in the second second second of the video?
(GT: four ducks)

UniPixel: In the second second of the video, the dog chases three ducks across the grassy field. The sequence shows the dog initially lying down before getting up and moving towards the ducks, which are seen running away from it. This interaction is captured in frames 9077 to 9089, where the dog's pursuit is clearly directed at the three ducks present in the scene.

Answer: Three ducks. (Wrong) ❌

Example 2: With the textual map 😊

Prompt: Given the <raw video> [video icon] and the <textual map> [map icon] How many objects are chased by the dog in the second second second of the video?
(GT: four ducks)

UniPixel: The puppy chases four ducks across the grassy field, as seen in the sequence of frames where the dog moves from a resting position to actively pursuing the birds. The ducks are consistently shown moving away from the puppy, indicating a clear chase scenario involving only these four objects.

Answer: four ducks (Correct) ✅

Inter-object: movement and temporal dynamics



Example 1: Without the textual map 😞

Prompt: Given the <raw video> [video icon] Please describe the object next to the woman making a forward jump in the second second of the video (GT: A black and white dog)

UniPixel: There are two dogs in a grassy field under a cloudy sky. One dog is larger with a fluffy white and grey coat, while the other is smaller with black and white markings. The two dogs often stand on their hind legs, jumping up towards the woman as she gestures.

Answer: Two dogs. (Wrong) ❌


Example 2: With the textual map 😊

Prompt: Given the <raw video> [video icon] and the <textual map> [map icon] Please describe the object next to the woman making a forward jump in the second second of the video (GT: A black and white dog)

UniPixel: In the second second of the video, the object next to the woman making a forward jump is a dog with a black and white coat. The dog appears to be in motion, likely participating in a training or play activity with the woman, as it is positioned close to her and seems to be responding to her actions.

Answer: A dog with a black and white coat. (Correct) ✅

Inter-object: spatial relationships and change



Example 1: Without the textual map 😞

Prompt: Given the <raw video> [video icon] At the start of the video, where is the person in the white shirt relative to the breakdancer?
(GT: To the left of the breakdancer.)

UniPixel: At the start of the video, the person in the white shirt is positioned to the right of the breakdancer, standing slightly behind and to the side, observing the performance. This individual appears to be part of the audience, located near the edge of the crowd that has gathered around the dancer.

Answer: To the right of the breakdancer (Wrong) ❌

Example 2: With the textual map 😊


Prompt: Given the <raw video> [video icon] and the <textual map> [map icon] At the start of the video, where is the person in the white shirt relative to the breakdancer?
(GT: To the left of the breakdancer.)

UniPixel: At the start of the video, the person in the white shirt is positioned to the left of the breakdancer, standing near the edge of the crowd and slightly behind the performer. This individual appears to be observing the dance routine, with their body oriented towards the center where the breakdancer is performing.

Answer: To the left of the breakdancer (Correct) ✅

Figure 4. Representative failure cases of UniPixel-7B on Dyn-Bench, showing qualitative comparisons incorporating ST-TCM.

Object-Camera: temporal and visual changes



Example 1: Without the textual map 😞

Prompt: Given the <raw video> 📹 During the interval from t=0.5s to t=1.5s, is the camera positioned above, at level with, or below the motocross rider?(GT: Below the rider.)

VST: During the interval from t=0.5s to t=1.5s, the motocross rider is level with the camera as they navigate a turn on the dirt track, maintaining a consistent position relative to the viewer's perspective. The rider remains at the same height as the camera, neither above nor below, and continues along the course with precision and control.

Answer: Level with the rider. (Wrong) ❌


Example 2: With the textual map 😊

Prompt: Given the <raw video> 📹 and the <textual map> 🗺️ During the interval from t=0.5s to t=1.5s, is the camera positioned above, at level with, or below the motocross rider?(GT: Below the rider.)

VST: During the interval from t=0.5s to t=1.5s, the camera is positioned below the motocross rider. This is evident from the consistent negative elevation angles reported in the data, which indicate that the rider is above the camera's line of sight. The camera's position remains relatively low and stable throughout this period, while the rider ascends and descends over the terrain, maintaining a higher elevation relative to the camera.

Answer: Below the motocross rider (Correct) ✔️

Object-camera: camera motion and orientation



Example 1: Without the textual map 😞

Prompt: Given the <raw video> 📹 Which interval shows the smoothest camera motion with the lowest acceleration variance? (GT: D)

A. 0.0s to 0.5s
B. 0.5s to 1.0s
C. 1.0s to 1.5s
D. 1.5s to 2.3s

VST: The interval from 0.0 to 0.5 seconds shows the smoothest camera motion with the lowest acceleration variance, as the camera position changes minimally and consistently during this period, indicating a steady and stable movement without abrupt accelerations or decelerations.

Answer: A (Wrong) ❌

Example 2: With the textual map 😊


Prompt: Given the <raw video> 📹 and the <textual map> 🗺️ Which interval shows the smoothest camera motion with the lowest acceleration variance? (GT: D)

A. 0.0s to 0.5s
B. 0.5s to 1.0s
C. 1.0s to 1.5s
D. 1.5s to 2.3s

VST: The interval from 1.5s to 2.3s shows the smoothest camera motion with the lowest acceleration variance, as the camera position changes more gradually and consistently during this period compared to the earlier interval. The data indicates that while both intervals exhibit acceleration, the motion in the latter is more stable, with fewer abrupt changes in direction or speed.

Answer: D (Correct) ✔️

Object-camera: camera-object interaction



Example 1: Without the textual map 😞

Prompt: Given the <raw video> 📹 From the beginning of the video to around 1.0s, does the camera's position relative to the person on the bicycle transition from front to right, left to back, or remain in the same sector?(GT: Remains in the front sector.)

VST: Initially, the cyclist is positioned directly ahead of the camera at a slight angle. As time progresses, the azimuth shifts gradually from positive values (front-left) to negative values (right side). This transition is evident in the changing azimuth angles from approximately 6° to -23°, indicating a shift from front to right.

Answer: From front to right (Wrong) ❌

Example 2: With the textual map 😊

Prompt: Given the <raw video> 📹 and the <textual map> 🗺️ the beginning of the video to around 1.0s, does the camera's position relative to the person on the bicycle transition from front to right, left to back, or remain in the same sector?(GT: Remains in the front sector.)

VST: From the beginning of the video to around 1.0 seconds, the camera's position relative to the person on the bicycle remains in the same sector, as the azimuth angle changes only slightly from 6° to -23°, indicating a gradual shift in direction rather than a transition from behind to front.

Answer: Remains in the front sector. (Correct) ✔️

Figure 5. Representative failure cases of VST-7B-RL on Dyn-Bench, showing qualitative comparisons incorporating ST-TCM.

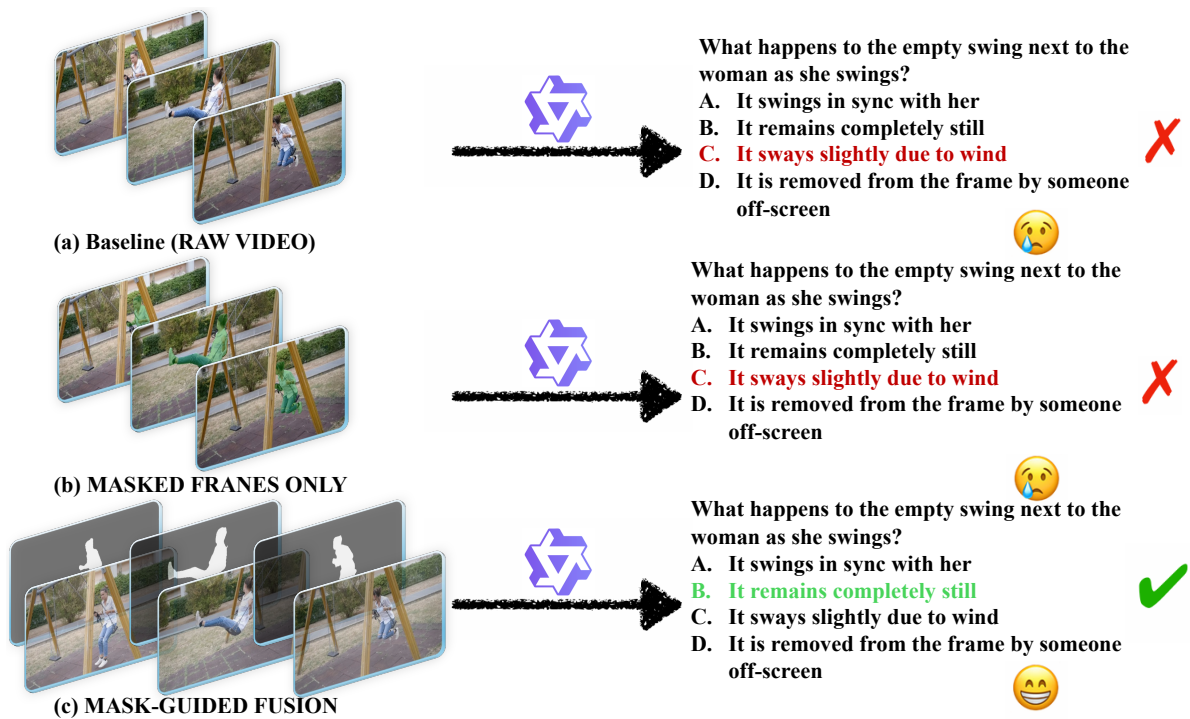


Figure 6. More visual comparison results on *Mask-Guided Input*.

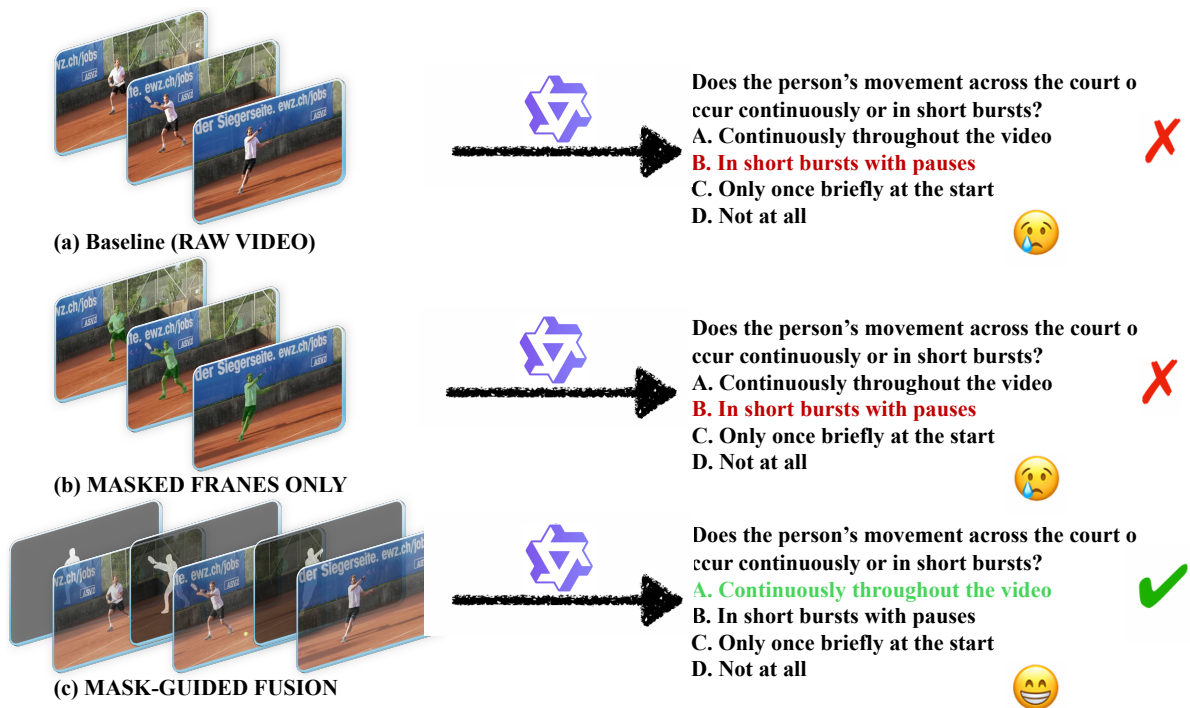


Figure 7. More visual comparison results on *Mask-Guided Input*.

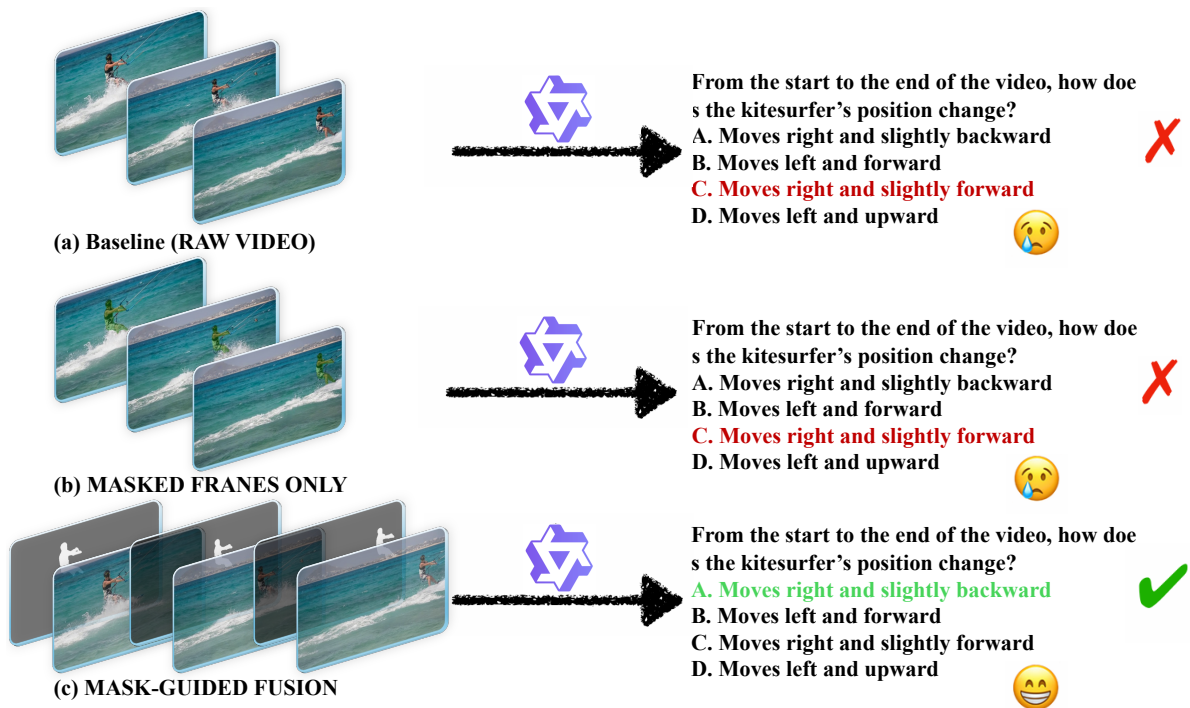


Figure 8. More visual comparison results on *Mask-Guided Input*.

Dynamic Inter-Object Perception

You are given a set of video keyframes and structured scene context data describing dynamic interactions among objects.

Scene Context JSON includes:

- video_id: video identifier
- frames: array of frames
 - * each frame has frame_id, timestamp, and a list of objects
 - * each object includes:
 - object_id: identifier
 - world_info: 3D location and size in world coordinates
 - description: natural-language details about size, state, nearby objects, and relative positions

World coordinate system:

- x-axis: left-right direction (positive = right, negative = left)
- y-axis: up-down direction (positive = down, negative = up)
- z-axis: forward-backward direction (positive = forward, farther from camera. negative = backward, closer to the camera)

Interpretation examples:

- If an object's x increases, it moves to the right.
- If y increases, it moves downward.
- If z increases, it moves forward, away from the camera.
- Never confuse "forward" with "up" or "right." These are independent axes.

Naming & Referencing Policy (VERY IMPORTANT)

- Always refer to items by real-world category names and natural descriptors from description (e.g., "the red mug," "the parked car," "the person in a blue jacket").
- Never use placeholder tokens such as "object," "obj," "obj1/obj2".
- If multiple similar items exist, disambiguate naturally (e.g., "the taller box," "the box with a sticker," "the metal chair") using attributes from description.
- Do not quote or surface object_id in the text. Use only real-world categories and descriptors.

Rules:

- Refer to entities naturally (e.g., "the small cube," "the larger box"), not as "obj1," "obj2."
- For spatial questions, all reasoning must follow the defined 3D coordinate meanings above (especially the z-axis as forward/backward).
- Use fluent, human-like phrasing.
- Avoid numeric coordinates in the questions and options

Task

Generate exactly 11 multiple-choice question-answer pairs, one per category below. Each question must have four options labeled A-D with only one correct answer. Keep wording concise and natural.

1. Video description (scene summary) — Describe the overall activity in the video using natural language.
2. Object counting (multiple objects) — Count how many distinct objects appear.
3. Object stationary detection (single object) — Does an object remain still or move during the video.
4. Average speed (single object) — Ask for an object's average speed between its first and last appearances; provide four plausible numeric choices with the correct unit.
5. Spatial relation at one moment (multiple objects) — Where is one object relative to another (e.g., "left of").
6. Position change (single object) — From moment A to moment B, how does an object's position change (e.g., "moves right and forward," "moves left and upward")?
7. Change in distance over time (multiple objects) — Do two objects get closer, farther apart, or stay about the same over the video.
8. Relative position change (multiple objects) - Between two specified moments A and B (e.g., two timestamps or event cues), how does the relative position of one object to another change?
9. Distance at a specific moment (multiple objects) — At a specified timestamp or event cue, what is the distance between two clearly identified objects?
10. Event order (temporal precedence) — Which happens first in the video (e.g., "Does the person pick up the cup before the door opens, or after?").
11. Action continuity — Does an action occur continuously, intermittently (start-stop), only once briefly, or not at all (e.g., "Does the small drone hover steadily or in short bursts?").

Output format

Output only a single JSON array [...] with 11 items. Each item must follow:

```
{
  "question": "string",
  "options": ["A. ...", "B. ...", "C. ...", "D. ..."],
  "answer": "A|B|C|D"
}
```

Style and Language

- Refer to objects by real-world categories (e.g., "the car", "the person"), not IDs like "obj1"
- Use fluent, human-like phrasing for both questions and answers.
- Avoid numeric coordinates in the question/answer.
- Ensure all spatial terms respect the axis definitions above.

Example

```
[
  {
    "question": "What best describes the main activity in the video?",
    "options": ["A. A car parks beside a curb", "B. A person jogs past a bench", "C. A drone hovers in place", "D. A ball rolls down stairs"],
    "answer": "B"
  }
]
```

Figure 9. Prompt template for *Dynamic Inter-Object Perception* in Dyn-Bench.

Dynamic Object-Scene Tracking

You are given video keyframes and a cognitive map JSON containing object tracking data.
Focus on the overall scene dynamics — how the main moving objects interact, change positions, and transform the spatial composition of the scene over time.
Generate 6 simple multiple-choice questions that reflect scene-level understanding, not isolated object actions.
Each question should capture the global motion patterns, object interactions, or changes in scene layout as time progresses.

Task:

Generate exactly 6 multiple-choice question-answer pairs. Each question must have four options labeled A–D with only one correct answer. Keep wording concise and natural. Each question should highlight scene-level dynamics. You can use these types of questions as examples or adapt them naturally:

1. How does the overall spatial arrangement of elements change throughout the scene?
2. What happens to the person/vehicle/animal performing action XXX in relation to the rest of the scene?
3. How does the movement of one entity influence others or alter the scene composition?
4. Describe the appearance or spatial position of the person/vehicle/animal performing action XXX within the evolving scene.

Questions should use real-world concepts and entities, not abstract placeholders like “object” or “key object”.

Output format:

Output only a single JSON array with 6 items, e.g.:

```
[
  {
    "question": "string",
    "options": ["A. ...", "B. ...", "C. ...", "D. ..."],
    "answer": "A|B|C|D"
  },
  ...
]
```

Description Rules:

1. Scene Initialization – Identify key dynamic objects and their initial layout within the environment.
2. Scene Evolution – Describe how spatial relations, movements, or object interactions develop over time. Use temporal connectives such as: initially, then, next, meanwhile, as the scene progresses, finally.
3. Scene Transformation – Explain how the scene composition changes by the end (e.g., objects converge in the center, one leaves the frame while others remain).

Style Guidelines:

- Always refer to entities by real-world categories or descriptions, not generic terms like “object”, “key object”, or IDs like “obj1”.
- Use fluent, human-like phrasing for both questions and answers.
- Questions should be simple and easy to understand.
- Avoid camera-related or numerical terms (e.g., “frame”, “meters”, “timestamp”).
- Focus on scene-level context, not camera movement or technical metrics.
- Use natural spatial references (“near the center”, “across the street”).
- Emphasize relationships between objects and overall scene transitions.
- Write in a concise, natural, human-observer style.
- Critical: Output must be pure JSON starting and ending with [and] — no markdown, comments, or extra text.

Figure 10. Prompt template for *Dynamic Object-Scene Tracking* in Dyn-Bench.

Dynamic Camera-Object Reasoning

You are given video keyframes and structured scene context that include per-frame CAMERA pose and OBJECTS. Your goal is to generate **exactly 11** multiple-choice questions (A–D, single correct) that assess **camera-centric** understanding: the camera’s motion in the world frame and the camera’s motion **relative to** clearly named real-world objects. Output ONLY a single JSON array with 11 items; no extra text.

INPUT

Scene Context JSON (example fields):

- video_id
- frames: array
 - * frame_id, timestamp
 - * camera: position: (x,y,z)
 - * objects: list of visible entities, each with:
 - world_info: 3D location/size in world coordinates
 - description: natural language attributes (category, color, size, accessories, state, nearby items)

CAMERA COORDINATE SYSTEM

This scene uses a standard **camera coordinate frame** consistent with COLMAP/OpenCV:

- +x: right (image right)
- +y: down (image down)
- +z: forward (into the scene, away from camera origin)

BEARING (OBJECT RELATIVE TO CAMERA)

When the scene context provides bearing as (azimuth, elevation) for an object **relative to the camera**:

- Azimuth: horizontal angle around the camera; sign maps to **left/right**.
 - Elevation: 0° at camera’s level; **elevation < 0** ⇒ object is below camera (viewer must look downward); **elevation > 0** ⇒ object is above camera (viewer must look upward).
- Use bearing only for **qualitative** judgments (left/right/front/back, above/below, nearer/farther). **Do not expose raw numeric angles** in questions/options.

REFERENCE FRAMES

- [WORLD]: camera motion w.r.t. the world frame (default reference is the first frame unless a time range is specified).
 - [RELATIVE]: camera’s position/trajectory **around a named real-world object** (object-centric top view: front/right/back/left; distance change; orbital CW/CCW).
- Note: WORLD yaw direction and RELATIVE orbital direction may differ; evaluate independently.

NAMING & REFERENCING

- Refer to entities **ONLY** by natural descriptors present in ‘description’: e.g., “the larger brown goat”, “the black goat wearing a blue cloth”.
- NEVER use placeholders/IDs or pronouns: no “object/obj”, “Object 1/2”, “it”, “its”, or “the subject”.
- **Descriptor Lock (critical):** once you name an entity, use the **EXACT SAME** descriptor verbatim across all 11 questions and options. Do not shorten/replace with “the goat”, “it”, etc.
- If multiple similar entities exist, compose a unique descriptor (color + size + accessory + pose). If uniqueness is not possible, **do not** create that question.
- Do NOT invent unseen attributes (logos, text, brands, numbers) not present in JSON/descriptions.

OPTION DESIGN & UNIQUENESS

- Exactly four options A–D; options must be **mutually exclusive**; exactly **one** correct.
- Numeric concepts (distance/angle/speed) must be expressed as **non-overlapping ranges** or qualitative tiers (closer/farther/=constant/fluctuates-ends-similar; negligible/small/medium/large). Do not print raw numbers/angles.
- Make the dominant aspect explicit when asked (e.g., “dominant orientation change = yaw clockwise (medium)”).

TASK: 11 CAMERA-CENTRIC CATEGORIES

- 1) [WORLD] Camera motion summary — Natural-language summary of overall CAMERA motion (e.g., “curved rightward with gradual clockwise pan and slight descent”).
- 2) [WORLD] Camera stationary detection — Does the CAMERA remain still or move (whole video or a specified interval)?
- 3) [WORLD] Translation direction — Between time A and B, what is the CAMERA’s **dominant** translation (e.g., right & forward / left & upward)?
- 4) [WORLD] Orientation change — Between time A and B, what is the CAMERA’s **dominant** orientation change? (axis + direction + qualitative magnitude: yaw CW/CCW, pitch up/down, roll right/left; tiers: negligible/small/medium/large)
- 5) [RELATIVE] Distance trend to a named object — Over an interval, does the CAMERA get closer to <named object>, farther, about the same, or fluctuate but end similar?
- 6) [RELATIVE] Orbital bearing change — Around <named object> (top view), is the CAMERA’s net orbit clockwise, counterclockwise, negligible, or reversing?
- 7) [RELATIVE] Lateral sector transition — From Early→Late (or A→B), does CAMERA move relative to <named object> from left/right/front/back to another sector (e.g., front→right)?
- 8) [RELATIVE] Relative altitude & trend — Is CAMERA above/below <named object> during a period, and is it becoming higher/lower/stable?
- 9) [WORLD] Motion smoothness/stability — Which phase/interval is smoothest (lowest acceleration variance) or shows handheld-like micro-shake?
- 10) [WORLD vs RELATIVE] Zoom vs Dolly cue — If <named object> apparent scale changes, which explanation fits: FOV/zoom change, forward/backward translation (distance change), both, or neither?
- 11) [RELATIVE] Event order (camera vs a named object) — Which happens first: CAMERA starts a specific motion (e.g., rightward pan/truck) vs <named object> enters/exits/changes side? Options must be mutually exclusive (A first / B first / same frame / only one occurs).

QUALITY SELF-CHECK BEFORE RETURNING

- Exactly 11 items, covering the 11 categories **in order**.
- Every item is CAMERA-centric; named objects are reference anchors with consistent descriptors (**Descriptor Lock enforced; no pronouns**).
- Options are mutually exclusive; exactly one correct.
- No raw coordinates/angles/object_ids in the text.
- Timestamps are within bounds; all referenced entities are visible at the cited times.
- All direction terms are **screen-aligned** (viewer phrasing), not coordinate signs.

OUTPUT FORMAT

Return ONLY a JSON array with 11 items. Each item:

```
{
  "question": "string",
  "options": ["A. ...", "B. ...", "C. ...", "D. ..."],
  "answer": "A|B|C|D"
}
```

Figure 11. Prompt template for *Dynamic Camera-Object Reasoning* in Dyn-Bench.

Dynamic Inter-Object Grounding

You are given video keyframes and an object JSON file describing detected objects. Please generate exactly 3 expression pairs (3 statements and 3 questions with consistent content) for DYNAMIC (moving) objects ONLY.

The expressions MUST describe the target object's spatiotemporal relationship with **another dynamic object** in the scene.

OUTPUT FORMAT:

Generate the following fields for each dynamic object:

- crop_caption: Detailed description of the object's appearance in a single frame (focus on visual features, colors, shapes, textures)
- crop_category: Brief category/type identification of the object (e.g., 'A dog', 'A person', 'A vehicle')
- formatted: Comprehensive description of the object's appearance, position, and movement throughout the video (combine all aspects)
- expression_1: Relative spatial position statement (e.g., 'The person in blue is to the left of the person in red')
- question_1: Relative spatial position question (e.g., 'What object is to the left of the person in red?')
- expression_2: Interaction statement (e.g., 'The person in blue is handing a bag to the person in red')
- question_2: Interaction question (e.g., 'What object is handing a bag to the person in red?')
- expression_3: Relative movement pattern statement (e.g., 'The white car is moving towards the black car')
- question_3: Relative movement pattern question (e.g., 'What object is moving towards the black car?')

CRITICAL REQUIREMENT:

You must generate expressions ONLY for objects that show movement patterns (not stationary objects).

DYNAMIC OBJECT IDENTIFICATION: Look for objects that have changing positions, speeds, or movement patterns across the video frames.

SKIP stationary objects that remain in the same position throughout the video.

MOVEMENT VERIFICATION: Before generating expressions for any object, verify that it actually moves or changes position in the video.

If an object remains stationary (like roads, walls, furniture, static lines, or fixed background elements), DO NOT generate expressions for it.

DYNAMIC OBJECT EXAMPLES: People walking, animals moving, vehicles driving, objects being carried, etc.

STATIC OBJECT EXAMPLES: Roads, sidewalks, walls, furniture, painted lines, buildings, trees (if not swaying), etc.

Establish a one-to-one mapping between JSON object IDs and output object IDs.

If the JSON contains obj1, obj2, obj3, then you MUST generate object_1, object_2, object_3.

Mapping rule: obj1 -> object_1, obj2 -> object_2, obj3 -> object_3, etc.

Level 1 Expression Generation Strategy - Exactly 3 Pairs per Object:

For each object, generate exactly 3 expression pairs covering these DISTINCT spatiotemporal aspects. All expressions must relate the target object to ANOTHER dynamic object.

1. **Relative Spatial Position (Obj-Obj):**

* Statement: Describe ONLY where the object is located relative to **another dynamic object** (e.g., 'The person in blue is standing to the right of the person in red').

* Question: Ask about what object is in this specific relative position (e.g., 'What object is standing to the right of the person in red?').

* CRITICAL: Must involve two dynamic objects.

2. **Dynamic Interaction (Obj-Obj):**

* Statement: Describe a direct spatiotemporal interaction between the object and **another dynamic object** (e.g., 'The person in blue is handing a bag to the person in red', 'The dog is chasing the cat').

* Question: Ask about what object is performing this interaction (e.g., 'What object is handing a bag to the person in red?').

* CRITICAL: Must describe an active, dynamic interaction, not just proximity.

3. **Relative Movement Pattern (Obj-Obj):**

* Statement: Describe ONLY the object's motion relative to **another dynamic object's** motion (e.g., 'The white car is moving away from the black car', 'The person in blue is following the person in red').

* Question: Ask about what object is performing this relative motion (e.g., 'What object is moving away from the black car?').

* CRITICAL: Must describe movement, not a static position.

Requirements:

- Generate exactly 3 expression pairs (3 statements + 3 questions) for DYNAMIC (moving) objects ONLY.

- You MUST identify and process ONLY objects that show movement patterns.

- SKIP stationary objects.

- Each statement/question should be a single natural sentence in English.

- Focus on qualitative descriptions and relationships.

- The final output must be a JSON object:

```
{
  "object_1": {
    "crop_caption": "...",
    "crop_category": "...",
    "formatted": "...",
    "expression_1": "Relative spatial position statement",
    "question_1": "Relative spatial position question",
    "expression_2": "Interaction statement",
    "question_2": "Interaction question",
    "expression_3": "Relative movement pattern statement",
    "question_3": "Relative movement pattern question"
  },
  "object_2": {
    ...
  },
  ...
}
```

- CRITICAL: Include ONLY dynamic (moving) objects from the JSON data.

- MANDATORY: Generate expressions for ALL dynamic objects following the mapping rule.

- STATIC OBJECT CHECK: Double-check that NO static objects are included.

- Do not include any explanations or text outside the JSON object.

Figure 12. Prompt template for *Dynamic Inter-Object Grounding* in Dyn-Bench.

Dynamic Object-Scene Grounding

You are given video keyframes and an object JSON file describing detected objects. Please generate exactly 3 expression pairs (3 statements and 3 questions with consistent content) for DYNAMIC (moving) objects ONLY.

The expressions MUST describe the target object's spatiotemporal relationship with the **overall scene** or **static background elements**.

OUTPUT FORMAT:

Generate the following fields for each dynamic object:

- crop_caption: Detailed description of the object's appearance in a single frame (focus on visual features, colors, shapes, textures)
- crop_category: Brief category/type identification of the object (e.g., 'A dog', 'A person', 'A vehicle')
- formatted: Comprehensive description of the object's appearance, position, and movement throughout the video (combine all aspects)
- expression_1: Scene-relative spatial position statement (e.g., 'The person in blue is standing in the middle of the intersection')
- question_1: Scene-relative spatial position question (e.g., 'What object is standing in the middle of the intersection?')
- expression_2: Spatiotemporal path statement (e.g., 'The person in blue walks from the sidewalk into the store')
- question_2: Spatiotemporal path question (e.g., 'What object walks from the sidewalk into the store?')
- expression_3: Scene-contextualized action statement (e.g., 'The dog is jumping over the puddle on the grass')
- question_3: Scene-contextualized action question (e.g., 'What object is jumping over the puddle on the grass?')

CRITICAL REQUIREMENT:

You must generate expressions ONLY for objects that show movement patterns (not stationary objects).

DYNAMIC OBJECT IDENTIFICATION: Look for objects that have changing positions, speeds, or movement patterns across the video frames.

SKIP stationary objects that remain in the same position throughout the video.

MOVEMENT VERIFICATION: Before generating expressions for any object, verify that it actually moves or changes position in the video.

If an object remains stationary (like roads, walls, furniture, static lines, or fixed background elements), **DO NOT** generate expressions for it.

DYNAMIC OBJECT EXAMPLES: People walking, animals moving, vehicles driving, objects being carried, etc.

STATIC OBJECT EXAMPLES: Roads, sidewalks, walls, furniture, painted lines, buildings, trees (if not swaying), etc.

Establish a one-to-one mapping between JSON object IDs and output object IDs.

If the JSON contains obj1, obj2, obj3, then you **MUST** generate object_1, object_2, object_3.

Mapping rule: obj1 -> object_1, obj2 -> object_2, obj3 -> object_3, etc.

Level 2 Expression Generation Strategy - Exactly 3 Pairs per Object:

For each object, generate exactly 3 expression pairs covering these **DISTINCT** spatiotemporal aspects. All expressions must relate the target object to the **STATIC SCENE**.

1. **Scene-Relative Spatial Position (Obj-Scene):**

* Statement: Describe **ONLY** where the object is located relative to the **overall scene** or a major static element (e.g., 'The person in blue is standing near the fountain', 'The car is in the background of the shot').

* Question: Ask about what object is in this specific scene-relative position (e.g., 'What object is standing near the fountain?').

* **CRITICAL:** Must relate to the static scene, not another dynamic object.

2. **Spatiotemporal Path (Obj-Scene):**

* Statement: Describe the object's **path** of movement across the scene by referencing static elements (e.g., 'The person walks from the left side of the screen to the right side', 'The ball rolls across the grass and stops on the path').

* Question: Ask about what object is performing this path of movement (e.g., 'What object walks from the left side of the screen to the right side?').

* **CRITICAL:** Must describe a trajectory or path, not a static position.

3. **Scene-Contextualized Action (Obj-Scene):**

* Statement: Describe a specific **dynamic action** the object performs **within** the scene context (e.g., 'The person is climbing the stairs', 'The dog is digging a hole under the tree').

* Question: Ask about what object is performing this scene-contextualized action (e.g., 'What object is climbing the stairs?').

* **CRITICAL:** Must be a specific action (e.g., 'climbing', 'jumping', 'digging'), not just 'moving'.

Requirements:

- Generate exactly 3 expression pairs (3 statements + 3 questions) for DYNAMIC (moving) objects ONLY.

- You **MUST** identify and process **ONLY** objects that show movement patterns.

- **SKIP** stationary objects.

- Each statement/question should be a single natural sentence in English.

- Focus on qualitative descriptions and relationships.

- The final output must be a JSON object:

```
{
  "object_1": {
    "crop_caption": "...",
    "crop_category": "...",
    "formatted": "...",
    "expression_1": "Scene-relative spatial position statement",
    "question_1": "Scene-relative spatial position question",
    "expression_2": "Spatiotemporal path statement",
    "question_2": "Spatiotemporal path question",
    "expression_3": "Scene-contextualized action statement",
    "question_3": "Scene-contextualized action question"
  },
  "object_2": {
    ...
  },
  ...
}
```

- **CRITICAL:** Include **ONLY** dynamic (moving) objects from the JSON data.

- **MANDATORY:** Generate expressions for **ALL** dynamic objects following the mapping rule.

- **STATIC OBJECT CHECK:** Double-check that **NO** static objects are included.

- Do not include any explanations or text outside the JSON object.

Figure 13. Prompt template for *Dynamic Object-Scene Grounding* in Dyn-Bench.

Dynamic Camera-Object Grounding

You are given video keyframes and an object JSON file describing detected objects. Please generate exactly 3 expression pairs (3 statements and 3 questions with consistent content) for DYNAMIC (moving) objects ONLY.

The expressions MUST describe the target object's spatiotemporal relationship with the **camera's perspective** (e.g., depth, scale, movement relative to the camera).

OUTPUT FORMAT:

Generate the following fields for each dynamic object:

- crop_caption: Detailed description of the object's appearance in a single frame (focus on visual features, colors, shapes, textures)
- crop_category: Brief category/type identification of the object (e.g., 'A dog', 'A person', 'A vehicle')
- formatted: Comprehensive description of the object's appearance, position, and movement throughout the video (combine all aspects)
- expression_1: Relative depth/proximity statement (e.g., 'The person in blue is the object closest to the camera')
- question_1: Relative depth/proximity question (e.g., 'What object is closest to the camera?')
- expression_2: Movement relative to camera statement (e.g., 'The person in blue is walking towards the camera')
- question_2: Movement relative to camera question (e.g., 'What object is walking towards the camera?')
- expression_3: Apparent scale/frame position statement (e.g., 'The large object occupying the center of the frame')
- question_3: Apparent scale/frame position question (e.g., 'What object is large and occupying the center of the frame?')

CRITICAL REQUIREMENT:

You must generate expressions ONLY for objects that show movement patterns (not stationary objects).

DYNAMIC OBJECT IDENTIFICATION: Look for objects that have changing positions, speeds, or movement patterns across the video frames.

SKIP stationary objects that remain in the same position throughout the video.

MOVEMENT VERIFICATION: Before generating expressions for any object, verify that it actually moves or changes position in the video.

If an object remains stationary (like roads, walls, furniture, static lines, or fixed background elements), **DO NOT** generate expressions for it.

DYNAMIC OBJECT EXAMPLES: People walking, animals moving, vehicles driving, objects being carried, etc.

STATIC OBJECT EXAMPLES: Roads, sidewalks, walls, furniture, painted lines, buildings, trees (if not swaying), etc.

Establish a one-to-one mapping between JSON object IDs and output object IDs.

If the JSON contains obj1, obj2, obj3, then you **MUST** generate object_1, object_2, object_3.

Mapping rule: obj1 -> object_1, obj2 -> object_2, obj3 -> object_3, etc.

Level 3 Expression Generation Strategy - Exactly 3 Pairs per Object:

For each object, generate exactly 3 expression pairs covering these **DISTINCT** spatiotemporal aspects. All expressions must relate the target object to the **CAMERA'S PERSPECTIVE**.

1. **Relative Depth / Proximity (Obj-Camera):**

* Statement: Describe the object's proximity to the camera, often relative to other objects (e.g., 'The person in blue is in the foreground', 'The red car is the object farthest from the camera').

* Question: Ask about what object is at this specific depth (e.g., 'What object is in the foreground?').

* **CRITICAL:** Must describe depth (foreground, background, closest, farthest).

2. **Movement Relative to Camera (Obj-Camera):**

* Statement: Describe the object's **movement trajectory** relative to the camera's viewpoint (e.g., 'The person is walking **towards** the camera', 'The bird is flying **away** from the camera', 'The car is moving from left to right across the frame').

* Question: Ask about what object is performing this camera-relative motion (e.g., 'What object is walking towards the camera?').

* **CRITICAL:** Must describe movement relative to the camera, indicating a change in depth or position **in the frame**.

3. **Apparent Scale / Frame Position (Obj-Camera):**

* Statement: Describe the object's **apparent size or dominant position in the frame** as an indicator of its relationship to the camera (e.g., 'The object that appears largest in the frame', 'The small object in the distance', 'The object in the center of the camera's view').

* Question: Ask about what object fits this scale or frame description (e.g., 'What object appears largest in the frame?').

* **CRITICAL:** Relates visual scale or frame position directly to camera perspective.

Requirements:

- Generate exactly 3 expression pairs (3 statements + 3 questions) for DYNAMIC (moving) objects ONLY.

- You **MUST** identify and process **ONLY** objects that show movement patterns.

- **SKIP** stationary objects.

- Each statement/question should be a single natural sentence in English.

- Focus on qualitative descriptions and relationships.

- The final output must be a JSON object:

```
{
  "object_1": {
    "crop_caption": "...",
    "crop_category": "...",
    "formatted": "...",
    "expression_1": "Relative depth/proximity statement",
    "question_1": "Relative depth/proximity question",
    "expression_2": "Movement relative to camera statement",
    "question_2": "Movement relative to camera question",
    "expression_3": "Apparent scale/frame position statement",
    "question_3": "Apparent scale/frame position question"
  },
  "object_2": {
    ...
  },
  ...
}
```

- **CRITICAL:** Include **ONLY** dynamic (moving) objects from the JSON data.

- **MANDATORY:** Generate expressions for **ALL** dynamic objects following the mapping rule.

- **STATIC OBJECT CHECK:** Double-check that **NO** static objects are included.

- Do not include any explanations or text outside the JSON object.

Figure 14. Prompt template for *Dynamic Camera-Object Grounding* in Dyn-Bench.

242

References

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

- [1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhao-hai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 1
- [2] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasapat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blisstein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 3
- [3] Mohsen Gholami, Ahmad Rezaei, Zhou Weimin, Yong Zhang, and Mohammad Akbari. Spatial reasoning with vision-language models in ego-centric multi-view scenes. *arXiv preprint arXiv:2509.06266*, 2025. 2
- [4] Jiahui Huang, Qunjie Zhou, Hesam Rabeti, Aleksandr Korovko, Huan Ling, Xuanchi Ren, Tianchang Shen, Jun Gao, Dmitry Slepichev, Chen-Hsuan Lin, Jiawei Ren, Kevin Xie, Joydeep Biswas, Laura Leal-Taixe, and Sanja Fidler. Vipe: Video pose engine for 3d geometric perception. In *NVIDIA Research Whitepapers arXiv:2508.10934*, 2025. 2
- [5] Ye Liu, Zongyang Ma, Junfu Pu, Zhongang Qi, Yang Wu, Ying Shan, and Chang Wen Chen. Unipixel: Unified object referring and segmentation for pixel-level visual reasoning. *arXiv preprint arXiv:2509.18094*, 2025. 3
- [6] Yifan Liu, Fangneng Zhan, Kaichen Zhou, Yilun Du, Paul Pu Liang, and Hanspeter Pfister. Abstract 3d perception for spatial intelligence in vision-language models, 2025. 2
- [7] Luigi Piccinelli, Christos Sakaridis, Yung-Hsu Yang, Mattia Segu, Siyuan Li, Wim Abbeels, and Luc Van Gool. UniDepthV2: Universal monocular metric depth estimation made simpler, 2025. 1
- [8] Yanwei Li, Jingjia Huang, Shen Yan, Siyuan Zhou, Zhe Liu, Xiangtai Li, Shuangye Li, Wenqian Wang, Yi Lin, Hengshuang Zhao, Rui Yang, Ziyu Zhu. Visual spatial tuning. *arXiv preprint arXiv:2511.05491*, 2025. 3
- [9] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 1
- [10] Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfu Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu, Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*, 2025. 3
- [11] Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How mul-

timodal large language models see, remember, and recall 299

spaces. In *Proceedings of the Computer Vision and Pattern* 300*Recognition Conference*, pages 10632–10643, 2025. 2 301

- [12] Haobo Yuan, Xiangtai Li, Tao Zhang, Zilong Huang, Shilin 302

Xu, Shunping Ji, Yunhai Tong, Lu Qi, Jiashi Feng, and 303

Ming-Hsuan Yang. Sa2va: Marrying sam2 with llava for 304

dense grounded understanding of images and videos. *arXiv* 305*preprint arXiv:2501.04001*, 2025. 1 306