HanDyVQA: A Video QA Benchmark for Fine-Grained Hand-Object Interaction Dynamics

Supplementary Material

Supplementary Material Contents

A. Details on Experimental Results

A.1. Multiple-Choice Questions

Performance comparison by GPT-4o answerability. In multiple-choice questions, the GPT-4o (text/vision) models often declined to answer, seemingly when the model evaluated that there is insufficient information to respond to the question, even though we asked it to answer the question. Reporting results only for the questions that GPT-4o answered risks an unfair comparison, as those questions might also be easy for the other models. To this end, we re-evaluated every model on the subset of questions answered by both the text and vision variants of GPT-4o, reducing the valid items from 9,431 to 6,629 (70.3%).

Table 7 reports these subsampled scores together with their deltas from the original evaluation. Most metrics increase, indicating that GPT-4o tends to skip more difficult questions that other models also frequently fail on. In contrast, human performance remains largely unchanged across categories (within 0.5 points), suggesting that humans can reliably recognize HOI dynamics even in challenging videos where MLLMs struggle.

We divided the questions into four categories according to whether the GPT-4o text and vision variants produced an answer, then evaluated Qwen2.5-VL-72B on each category (Table 8). Accuracy peaked for questions answered by both GPT-4o variants and declined whenever either variant abstained. This pattern suggests that GPT-4o can recognize when textual or visual information is insufficient and refrain from answering, thereby avoiding errors. This also suggests that models specifically designed for videos perform better than the general-purpose GPT-4o model, showing that our benchmark poses a challenging video QA task.

Additional Qualitative Results. Additional qualitative results per category are shown in Figure 7. We observe that current models struggle to accurately recognize objects being manipulated, their spatial relationships with hands or other objects, and their movements. The original video clips can be found in the supplementary materials.

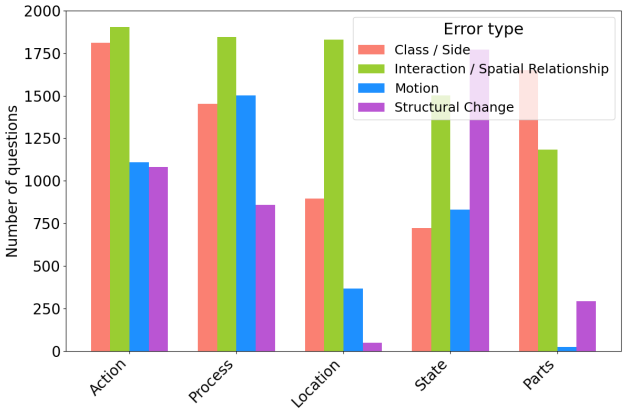

Figure 6. Distribution of error types by question category. Bars indicate the number of questions that contain at least one distractor of each error type.

Error Type Statistics. For each distractor, we annotated the following error types by comparing it with the correct answer:

- **Class / Side Confusion**: The model confuses the hand side (left or right) or the object class.
- **Interaction / Spatial Relationship Error**: The model fails to capture interaction types and spatial relations in hand-object and object-object pairs.
- **Motion Error**: The model fails to capture motion or direction over time.
- **Structural Change Error**: The model fails to perceive state or structural change in hands/objects.

Multiple error types may apply to a single distractor. For example, if the correct answer is "*He is hammering a cylinder with the hammer in his right hand in a downward motion,*" and a distractor is "*He is hammering a cylinder with the hammer in his left hand in a sideways motion,*" the assigned error types would be **Class / Side** and **Motion**.

Figure 6 shows the number of questions that include each error type across all question categories. **Class / Side** and **Interaction / Spatial Relationship** errors are widely distributed across categories. **Motion** errors frequently appear in **Action**, **Process**, and **State**, where hand and object movements play critical roles. **Structural Change** errors appear most often in **State**, followed by **Action** and **Process**, where hand poses are particularly informative. **Location** contains fewer **Motion** errors, as distractors primarily describe positional changes using spatial relations between objects. **Parts** contains fewer **Structural Change** errors because the LLMs

Models (Zero-shot)	Visual Backbone	Resolution	LLM	Action (Acc)	Process (Acc)	Objects (AP)	Location (Acc)	State (Acc)	Parts (Acc)	Avg. (Acc)
Text only models										
GPT-4o (text) [17]	–	–	GPT-4o	37.3 (+0.7)	54.2 (+3.3)	34.4 (+0.1)	36.5 (+2.4)	40.0 (+0.5)	47.5 (+2.0)	43.1 (+1.8)
Open-source dual-encoder video-language models										
LaViLa (TSF-L) [59]	TimeSformer	224x224	–	63.7 (+2.1)	41.7 (+1.6)	68.6 (+0.1)	41.5 (+4.6)	40.6 (+1.7)	37.9 (+2.3)	45.1 (+2.5)
InternVideo2-Stage2 [44]	Original	224x224	–	42.1 (+1.0)	30.7 (+0.5)	37.1 (+0.1)	34.0 (+4.3)	36.2 (+1.3)	31.3 (+0.8)	34.8 (+1.5)
Open source video-language models w/ integrated LLMs										
VideoLLaMA2.1-7B [6]	SigLIP	384x384	Qwen2	44.0 (+2.6)	50.3 (+3.0)	53.2 (+0.3)	39.3 (+4.7)	47.9 (+0.9)	44.8 (+4.3)	45.2 (+3.1)
LLaVa-Video-7B [24]	SigLIP	384x384	LLaVa-7B	61.3 (+4.4)	58.2 (+4.5)	60.7 (+0.3)	55.4 (+4.9)	60.1 (+1.6)	59.6 (+5.0)	58.9 (+4.1)
mPLUG-Owl3-8B [53]	SigLIP	384x384	Qwen2	55.2 (+3.1)	56.1 (+3.0)	61.1 (+0.1)	50.2 (+4.4)	57.5 (+2.8)	52.6 (+4.0)	54.3 (+3.4)
Qwen2.5-VL-7B [42]	Original	384x384	Qwen2.5	64.3 (+3.5)	58.5 (+3.6)	54.1 (+0.2)	53.7 (+5.8)	57.9 (+1.2)	53.1 (+4.5)	57.5 (+3.7)
Qwen2.5-VL-72B [42]	Original	480x854	Qwen2.5	80.5 (+2.5)	77.6 (+4.2)	75.4 (+0.2)	69.9 (+6.7)	74.4 (+2.2)	66.5 (+4.0)	73.8 (+3.9)
Proprietary vision and language models w/ integrated LLMs										
GPT-4o (vision) [17]	Original	480x854	GPT-4o	60.5 (-0.8)	64.6 (+0.2)	64.1 (0.0)	53.9 (+2.4)	59.4 (+0.4)	59.3 (+0.8)	59.6 (+0.7)
Gemini-2.5-Pro [7]	Original	480x854	Gemini-2.5-Pro	81.3 (+2.2)	77.9 (+4.6)	78.9 (+0.1)	73.2 (+5.6)	75.1 (+1.2)	74.7 (+5.4)	76.4 (+3.8)
Human	–	–	–	98.6 (-0.1)	95.9 (0.0)	96.0 (+0.1)	96.6 (-0.2)	95.3 (-0.2)	96.9 (-0.5)	96.6 (-0.3)

Table 7. Comparison of different models on subset of questions answered by both the text and vision versions of GPT-4o. Only 6.6K questions (70.3%) are used for this evaluation. Differences from the full results in Table 3 are indicated in +X.X/-X.X.

Answered by GPT-4o?		Results of Qwen2.5-VL-72B (with number of questions)						
Text	**Vision**	**Action**	**Process**	**Objects**	**Location**	**State**	**Parts**	**Avg. (Sum)**
Yes	Yes	80.5 (955)	77.6 (1042)	75.4 (1493)	69.9 (764)	74.4 (1319)	66.5 (1056)	73.8 (5136)
Yes	No	67.6 (136)	65.5 (521)	54.5 (12)	54.8 (361)	62.6 (211)	55.5 (373)	61.2 (1602)
No	Yes	78.2 (444)	50.0 (8)	– (0)	63.4 (292)	54.3 (35)	50.8 (61)	59.3 (840)
No	No	67.6 (105)	71.4 (7)	– (0)	50.6 (170)	53.8 (13)	47.2 (53)	58.1 (348)

Table 8. Comparison of performance on questions grouped by whether the GPT-4o text/vision models provided an answer. The number in parentheses indicates the number of questions.

used during annotation often interpret a component as a single object, labeling part-confusion errors as **Class / Side** confusion instead.

A.2. Referring Video Object Segmentation

Additional qualitative results. Additional qualitative examples are shown in Figure 8. VideoLISA often fails to detect the target objects even when the ground-truth options are provided as prompts, likely due to domain shift from its training data to egocentric video [23]. It also tends to segment hands instead of the intended object regions in **Parts** questions.

Sa2VA generally follows the prompts more faithfully in both the **Objects** and **Parts** categories. However, when applied in a frame-wise manner, it sometimes segments visually similar but unmanipulated objects, or loses track of the manipulated object because it lacks temporal context (*e.g.*, missing the paint tube in the bottom-left case, or producing a false-positive segmentation of the paintbrush in the bottom-right example of Figure 8). When the ground-truth option is provided, Sa2VA often segments objects of the same class that are not being manipulated. This is likely because the ground-truth description does not precisely specify which instance is being interacted with. The original video clips can be found in the supplementary materials.

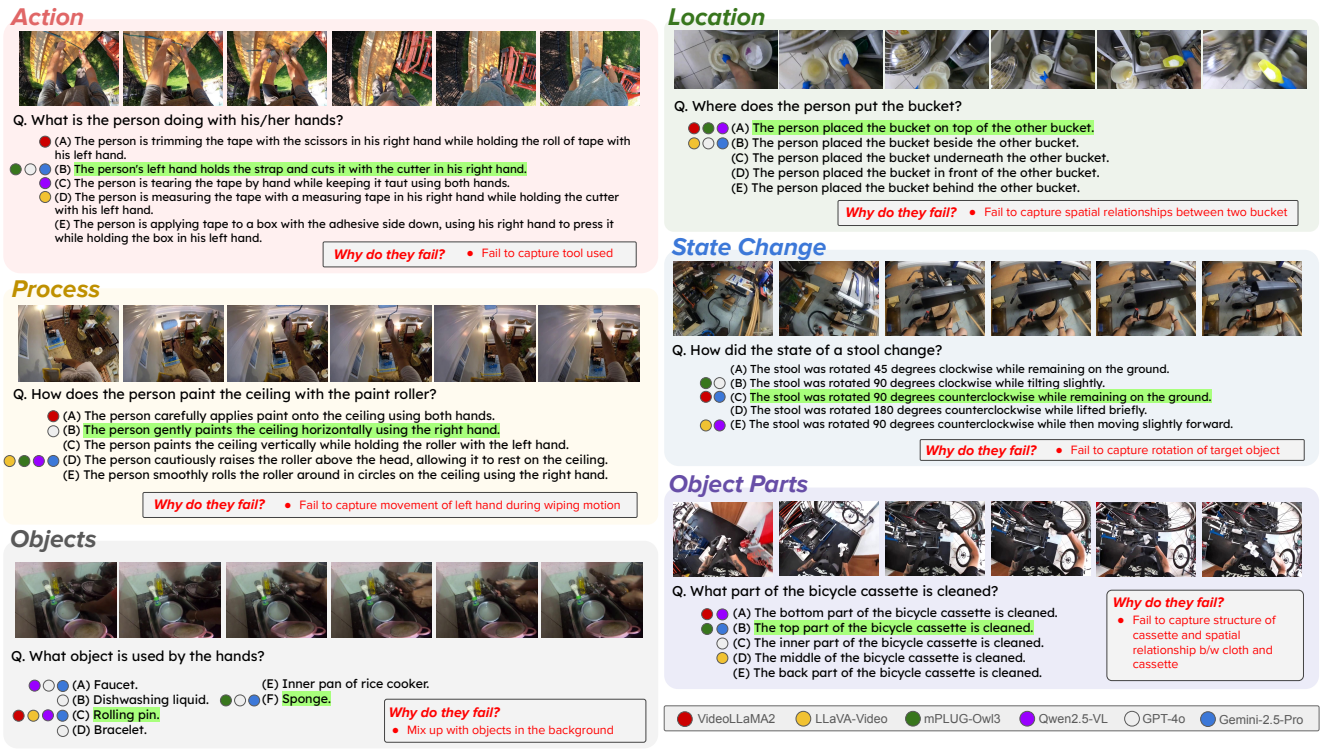

Figure 7. Additional qualitative results for multiple-choice questions. Green highlights denote correct answers.

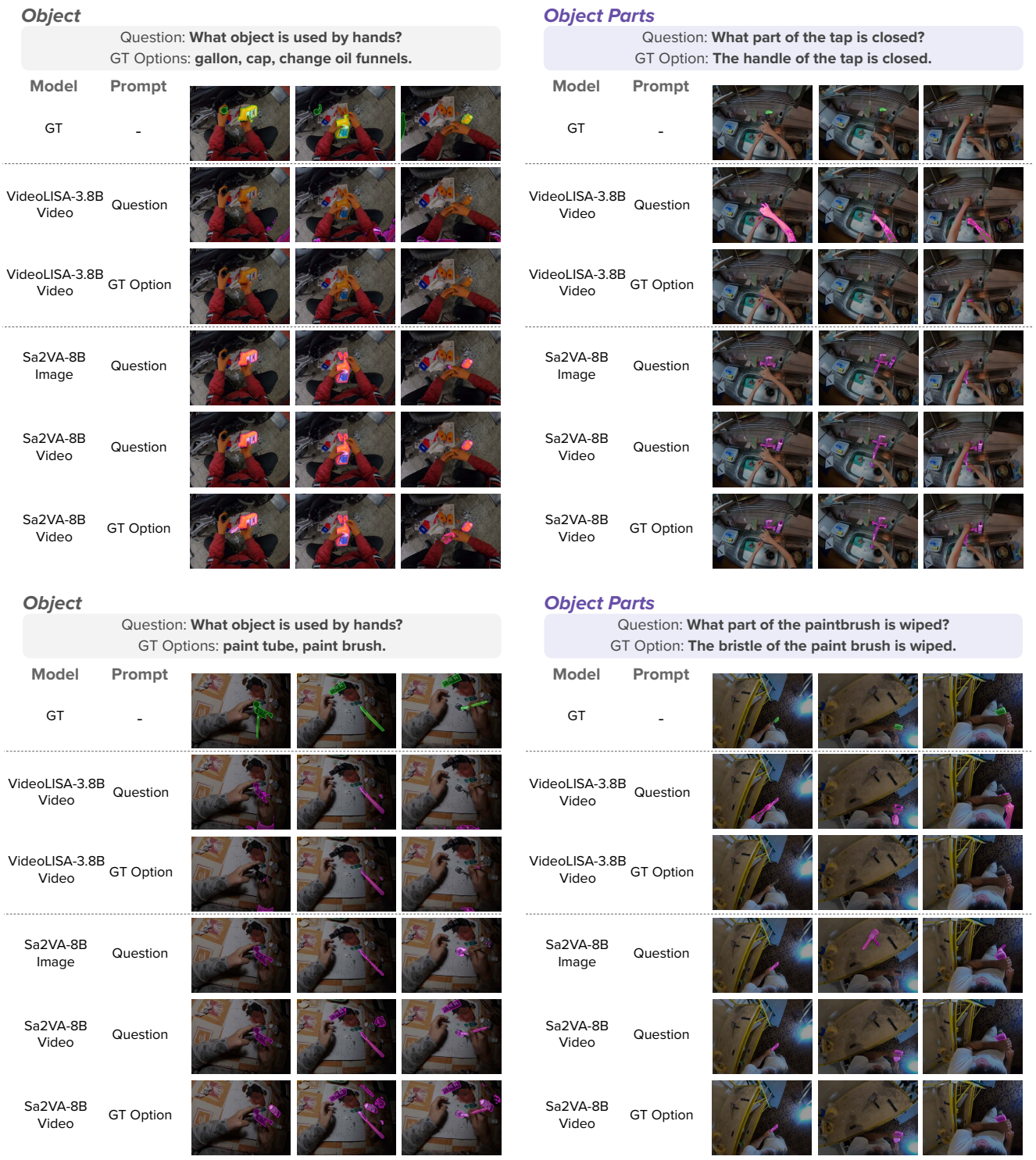

Figure 8. Qualitative results on ReasoningVOS. GT means ground-truth. GT masks are shown in green in first row, while predictions of each model are shown in magenta.

B. Details on Data Collection

Details on video clip filtering. First, all the narration from stereo videos is filtered out. If a narration is duplicated, it is counted as a single entry. A narration is also filtered out if it does not mention either "right hand" or "left hand." Additionally, if the narration contains the word "unsure", indicating an unknown object name, it is filtered out. We input each narration into large language models (LLMs) to infer the contact objects and secondary objects for each hand (The prompt is shown in Table 9). If we can confirm that the camera wearer is manipulating at least one object, we retain the narration for use. However, if the extracted contact objects include any of the wearer's body parts or the camera itself, the narration is discarded, as such cases are likely to obscure the visibility of hand-object interactions.

System Prompt:
You are a helpful assistant who understands interactions between human hands and objects.

User Prompt:
Please analyze the narration and answer the contact object and the secondary object of each right/left hand. "contact object" means the object that a hand is contacting (manipulating). "secondary object" means the object that the contact object is contacting (the object that is manipulated by the contact object). If the information is not specified in the narration, fill with None. The answer format should be json:
{"Contact object of right hand": <obj/None>, "Secondary object of right hand": <obj/None>, "Contact object of left hand": <obj/None>, "Secondary object of left hand": <obj/None>}

Narration: {narration text}

Table 9. Prompt to extract hand-object interaction information from narration. We replace {narration text} with narration from Ego4D.

Sampling HOI narrations. To ensure diverse narration samples for each question, we first sort the remaining narrations chronologically. Then, for each question, we offset the starting index and select every sixth narration. This staggered sampling prevents identical questions from being generated for temporally adjacent HOI segments.

Next, we filter out unsuitable narrations specifically for **Location** and **Object Parts**. For **Location** questions, we select videos where the narration contains verbs indicating object movement. For **Object Parts** questions, we determine whether the object is partially affected and retain only those where partial impact is evident. This filtering process is done automatically using LLMs based on the narration (The prompt is shown in Table 10).

Finally, to ensure diversity in HOI samples, we extract verbs from the narrations and randomly select 2,000 samples while ensuring that no single verb appears more than N times. N is determined for each question type to maintain diversity while ensuring at least 2,000 candidate samples are available.

System Prompt:
You are a helpful assistant who understands interactions between human hands and objects.

User Prompt:
Create a question for VQA about Hand-Object Interaction. Specifically, the question should follow the format "What part of [Object] is [Verb]?"
(e.g., "What part of the bicycle is replaced?").

Given a narration describing a Hand-Object Interaction:
1. First, determine if it's an appropriate scene to create a question.
2. An appropriate scene should include the following conditions:
- The person is interacting with the object(s).
- The action only affects a limited area of the object (e.g., replacing the tire of a bicycle or folding the left top corner of the paper) rather than the entire object (e.g., moving the speaker).

If the scene is appropriate for creating the question, create the question accordingly. If it's inappropriate to create a question, reply with None.
If possible, create the corresponding answer to the question. If it's difficult to create the answer, reply with Ambiguous.

Here is the narration:
{narration text}

Write reasoning and then output the json answer like this: <reasoning about whether it is appropriate to create a question>

```json
{"question": <created question or None>, "answer": <created answer or Ambiguous>}
```

Table 10. Prompt to extract hand-object interaction information from narration. We replace {narration text} with narration from Ego4D.

Human QA annotation. For the automatically generated questions, annotators perform the following tasks while referencing the video: (i) verifying the validity of the question, (ii) creating the correct answer, and (iii) generating incorrect answer choices.

If an automatically generated question does not match the video, annotators either revise the question or reject the sample. Next, they create the correct answer, ensuring that it provides sufficient detail for the question to be understandable without watching the video.

Once the question and correct answer are prepared, an initial set of plausible incorrect answer choices is generated using LLMs (The prompts are shown in Table 11– 16). Annotators then review these choices, filtering out inappropriate ones, such as those that overlap with the correct answer. They refine and add challenging distractors that effectively assess comprehension of the question.

Overall, human annotators verify all questions, correct answers, and incorrect answer choices, ensuring that the dataset remains sufficiently challenging while still solvable by humans. The instructions provided to the annotators are shown in Table 17– 22 for each category. The screenshot of the annotation tool interface is provided in Figure 9.

User Prompt:
You will be given a pair of a question and an answer about a hand-object interaction.
Create 4 sentences that describe similar but incorrect answers, as per the examples.
Examples:
- Question: What is the person doing with his/her hands?
- Correct Answer: The person is cutting an apple on the chopping board using the knife in his right hand while holding the apple with his left hand.
- Incorrect Answer (object name or situation is wrong): The person is slicing an apple on the table using an apple slicer in his right hand while holding the apple with his left hand.
- Incorrect Answer (action is different): The person is removing the skin of an apple on the chopping board using the knife in his right hand while grasping the apple with his left hand.

The sentences should:
- Contain either different object names or situations (as in the first incorrect example)
- Or, contain the same object names but describe different actions (as in the second incorrect example)
- Maintain the same level of detail and be of a similar length to the correct answer
- Avoid overlapping with each other

The answer format should be json:
{"1": "<sentence1>", "2": "<sentence2>", "3": "<sentence3>", ..., "4": "<sentence4>"}

Make sure that you only answer json.

Question: {question}
Correct Answer: {answer}

Table 11. Prompt to generate options for **Action** question. We replace {question} and {answer} with created question and answer, respectively

User Prompt:
You will be given a pair of a question and an answer about a hand-object interaction.
Create 4 sentences that describe similar but incorrect answers, as per the examples.
Examples:
- Question: How does the person loosen the fastener?
- Correct Answer: The person loosens the fastener by holding the piece firmly in one hand while using the wrench in the other hand to turn it.
- Incorrect Answer (hand movement or object handling is slightly different but plausible): The person loosens the fastener by shaking the piece in one hand while holding the wrench in the other hand.

The sentences should:
- Contain same objects but different hand movements or ways of doing the action compared to the correct answer
- Be incompatible with the correct answer
- Maintain the same level of detail and be of a similar length to the correct answer
- Avoid overlapping with each other

The answer format should be json:
{"1": "<sentence1>", "2": "<sentence2>", "3": "<sentence3>", ..., "4": "<sentence4>"}

Make sure that you only answer json.

Question: {question}
Correct Answer: {answer}

Table 12. Prompt to generate options for **Process** question. We replace {question} and {answer} with created question and answer, respectively

User Prompt:
You will be given a pair of a question and an answer about a hand-object interaction.
Create 4 sentences that describe similar but incorrect answers, as per the examples.
Examples:
- Question: What object is used by the hands?
- Correct Answer: knife,apple.
- Incorrect Answer: chopping board.

The sentences should:
- Contain incorrect object names that are likely to be in the same scene as the correct answer (e.g., chopping board)
- Even if the correct answer contains multiple objects, the incorrect answer should contain only one object per option
- Avoid overlapping with each other

The answer format should be json:
{"1": "<sentence1>", "2": "<sentence2>", "3": "<sentence3>", ..., "4": "<sentence4>"}

Make sure that you only answer json.

Question: {question}
Correct Answer: {answer}

Table 13. Prompt to generate options for **Objects** question. We replace {question} and {answer} with created question and answer, respectively

User Prompt:
You will be given a pair of a question and an answer about a hand-object interaction.
Create 4 sentences that describe similar but incorrect answers, as per the examples.
Examples:
- Question: Where does the person place the cup?
- Correct Answer: On the left bottom corner of the table.
- Incorrect Answer: On the right top corner of the table.
- Incorrect Answer: On the chair under the table.

The sentences should:
- Include a specific location indicating where the object is moved to, but it should be a different location from the correct answer
- Be close to but different from the correct location (e.g., "On the right top corner of the table.")
- Maintain the same level of detail and be of a similar length to the correct answer
- Avoid overlapping with each other

The answer format should be json:
{"1": "<sentence1>", "2": "<sentence2>", "3": "<sentence3>", ..., "4": "<sentence4>"}

Make sure that you only answer json.

Question: {question}
Correct Answer: {answer}

Table 14. Prompt to generate options for **Location** question. We replace {question} and {answer} with created question and answer, respectively

User Prompt:
You will be given a pair of a question and an answer about a hand-object interaction.
Create 4 sentences that describe similar but incorrect answers, as per the examples.
Examples:
- Question: How did the state of the screw change?
- Correct Answer: The screw was picked up, put on the hole, and turned clockwise.
- Incorrect Answer (a little bit different state change): The screw that had already been in the hole was turned counter-clockwise.
- Question: How did the state of the camera change?
- Correct Answer: The camera was moved right by the right hand. - Incorrect Answer (a little bit different state change): The camera was moved left and slightly shifted.

The sentences should:
- Describe very similar but different state changes of the object.
- Be incompatible with the correct answer.
- Not change the object (tools) used from the correct answer.
- Maintain the same level of detail and be of a similar length to the correct answer.
- Avoid overlapping with each other.

The answer format should be json:
{"1": "<sentence1>", "2": "<sentence2>", "3": "<sentence3>", ..., "4": "<sentence4>"}

Make sure that you only answer json.

Question: {question}
Correct Answer: {answer}

Table 15. Prompt to generate options for **State** question. We replace {question} and {answer} with created question and answer, respectively

User Prompt:
You will be given a pair of a question and an answer about a hand-object interaction.
Create 4 sentences that describe similar but incorrect answers, as per the examples.
Examples:
- Question: What part of the hammer is being held?
- Correct Answer: The bottom of the hammer handle.
- Incorrect Answer: Close to the head of the hammer.

The sentences should:
- Contain nonexistent parts or incorrect parts of the object (e.g., Close to the head of the hammer)
- Maintain the same level of detail and be of a similar length to the correct answer
- Avoid overlapping with each other

The answer format should be json:
{"1": "<sentence1>", "2": "<sentence2>", "3": "<sentence3>", ..., "4": "<sentence4>"}

Make sure that you only answer json.

Question: {question}
Correct Answer: {answer}

Table 16. Prompt to generate options for **Parts** question. We replace {question} and {answer} with created question and answer, respectively

Purpose	To ask how the person is manipulating objects and to understand the dynamically changing relationships between the hands and the objects.
Required answer information	• Details of manipulation by both hands (left and right, if distinguishable) • All objects involved in the action
Example question	What is the person doing with his/her hands?
Example correct answer	The person is slicing an apple on the chopping board using the knife in his right hand while holding the apple with his left hand.
Examples of incorrect answers	• The person is slicing an apple on the table using an apple slicer in his right hand while holding the apple with his left hand. • The person is removing the skin of an apple on the chopping board using the knife in his right hand while grasping the apple with his left hand.
Notes	Do not omit the subject ("The person").

Table 17. Instruction for annotators to annotate **Action** category

Purpose	To ask about the manner, procedure, technique, or skill involved in a hand action or its interaction with an object.
Required answer information	• Which hand is used • How the hand moves or interacts with the object, including the steps and state changes
Example question	How does the person drop the toy on the table?
Example correct answer	The person drops the action figure gently on the table by holding it with the index finger and thumb in the left hand.
Incorrect example	The person released his grip and let it fall with force.
Notes	Do not omit the subject ("The person") or the verb (action).

Table 18. Instruction for annotators to annotate **Process** category

Purpose	To identify the types and positions of objects being manipulated by the hands.
Required answer information	• A verbal description of all manipulated objects and their positions (including segmentation mask if applicable) • Objects that are merely touched and not clearly manipulated should not be included as correct or incorrect options
Example question	What object is used by the hands?
Correct answer	knife, apple
Incorrect example	The chopping board (present but not manipulated)
Notes	Only include object names in the answer. Separate multiple correct objects with commas.

Table 19. Instruction for annotators to annotate **Objects** category

Purpose	To ask where the manipulated object was moved to, or where it ended up as a result of the action.
Required answer information	A specific description of the location where the object was placed or moved.
Example question	Where does the person place the cup?
Example correct answer	The person places the cup on the left bottom corner of the table.
Incorrect example	The person places the cup on the top right corner of the table.
Notes	Do not omit the subject ("The person"), the verb (action), or the object.

Table 20. Instruction for annotators to annotate **Location** category

Purpose	To ask how the state, structure, composition, or spatial arrangement of the object changed during the video (or remained unchanged).
Required answer information	A description of how the object's state, structure, composition, or placement changed or did not change in the video.
Example question	How did the state of the apple change?
Correct answer	The apple was cut into small slices.
Example question 2	How did the state of the camera change?
Correct answer 2	The camera is divided into two parts: the body and the lens.
Incorrect examples	• The apple was crushed. • The apple was sliced. (when slicing did not occur)
Notes	Do not omit the subject (the object) or the verb.

Table 21. Instruction for annotators to annotate **State** category

Purpose	To identify the specific part of the object that is being affected, considering the object's structure, function, and spatial relation to the hands.
Required answer information	A detailed verbal description of the affected region and its position (including segmentation mask if applicable).
Example question	What part of the hammer is being held?
Correct answer	The bottom of the hammer handle is being held.
Incorrect example	Close to the head of the hammer is being held.
Notes	Do not omit the subject (the part) or the verb (effect).

Table 22. Instruction for annotators to annotate **Parts** category

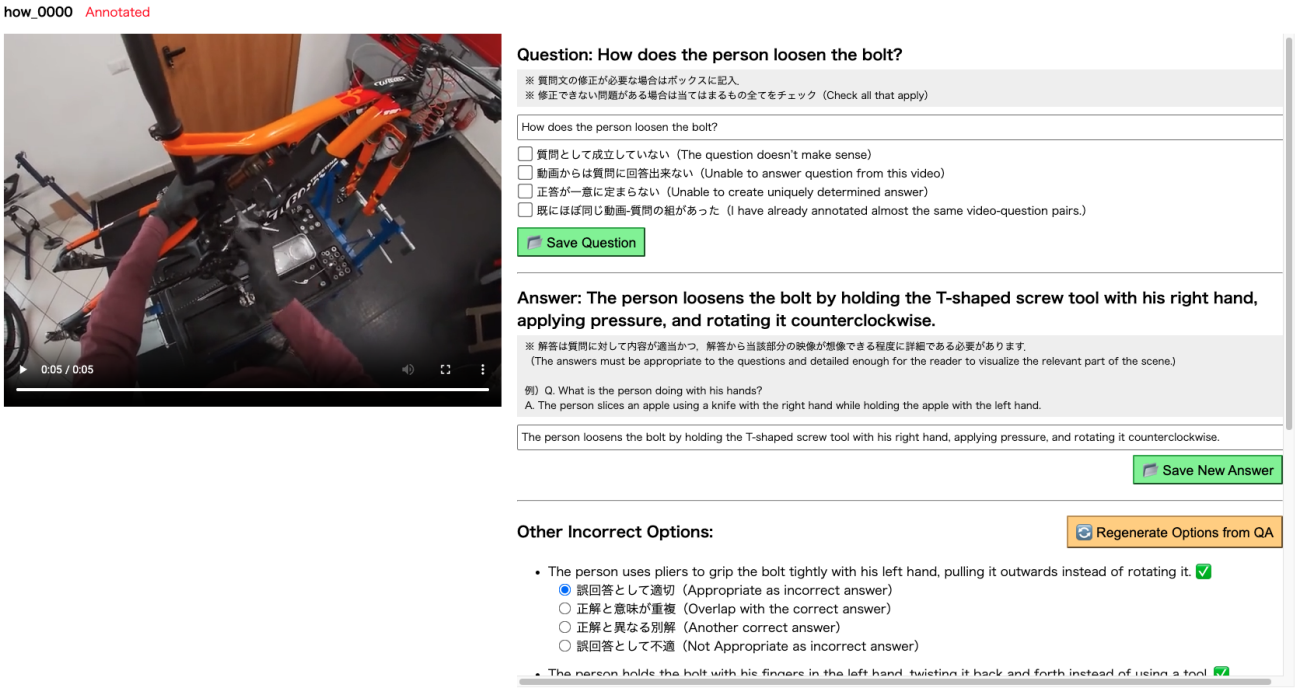

Figure 9. Screenshot of the annotation tool interface. Annotators proceed from top to bottom, sequentially annotating question, correct answer, and distractor options.

Final QA post-processing using LLMs. After human QA annotation, we refined each option, including the correct answer, using LLMs to correct their grammar and ensure a consistent tone across all choices, without changing their meaning. This was especially done for the **Action**, **Process**, and **State** categories, where longer sentences tend to introduce textual biases (e.g., the correct answer being more likely to contain grammatical errors than the distractors). All the prompts used for each question category are shown in Table 23 and Table 24.

You will be given a triplet consisting of a question, a correct answer, and a set of distractors.
First, revise the answer if it includes grammatical errors or is not in a natural tone without changing its meaning. If the answer is already correct, please keep it as is.
Then, rephrase each distractor to make it more plausible and similar in tone to the correct answer, without significantly changing its original meaning, since the distractors are currently written in a biased way, making them too easy to eliminate.
To do this, you may:
- Use words or phrasing that commonly appear in the correct answer, or avoid words frequently used in distractors.
- Tone down any exaggeration to make the distractors sound more natural and believable.
- Remove or rephrase strong negations (such as “without” or “instead”) if they clearly oppose the correct answer, unless they are essential to the meaning.

The answer (rephrased distractors) format should be json:
{
"answer": "<revised_answer>",
"options":{"1": "<sentence1>", "2": "<sentence2>", "3": "<sentence>", ..., "4": "<sentence4>" }
}

Make sure that you only answer json.
Note that all the sentences should start with in the same way as the original sentences (mostly "The person...").

Question: {question}
Correct Answer: {answer}
Distractors: {options}

Table 23. Prompt to refine options for **Action** and **Process** question. We replace {question}, {answer}, {options} with created question, answer, and options respectively

Choice of LLMs. We used `gpt-4o-mini-2024-07-18` to generate the initial QA pairs. For final refinement, we used `gpt-4o-2024-08-06`. We note that generated questions/options are for reference and all the pairs were thoroughly reviewed and corrected to form the final QA pairs.

You will be given a triplet consisting of a question, a correct answer, and a set of distractors.
First, revise the answer if it includes grammatical errors or is not in a natural tone without changing its meaning.
Note that since the answer is mainly written in the passive voice to focus on the state of the object, please make sure to keep it in passive form. If the answer is already correct, please keep it as is.

Then, rephrase each distractor to make it more plausible and similar in tone to the correct answer, without significantly changing its original meaning, since the distractors are currently written in a biased way, making them too easy to eliminate. Also, it’s better to avoid using adverbs (e.g., "gently") in the distractors, since adverbs typically describe human action rather than the state change of the object.

The answer (rephrased distractors) format should be json:
{
"answer": "<revised_answer>",
"options":{"1": "<sentence1>", "2": "<sentence2>", "3": "<sentence>", ..., "4": "<sentence4>" }
}

Make sure that you only answer json.
Note that all the sentences should start with in the same way as the original sentences (mostly "The person...").

Question: {question}
Correct Answer: {answer}
Distractors: {options}

Table 24. Prompt to refine options for **State** question. We replace {question}, {answer}, {options} with created question, answer, and options respectively

Human mask annotation. Egocentric videos often include severe blurring that harms the visual quality of the video clip. To this end, we opted to annotate three representative frames from each 5-second video clip. Annotators manually selected one frame each from the front, middle, and last thirds of the video clip, where the target regions were clearly visible and sampled them from different parts of the video whenever possible. However, the above condition was relaxed when the frames from some of the intervals

Human	Action (Acc)	Process (Acc)	Objects (AP)	Location (Acc)	State (Acc)	Parts (Acc)	Avg. (Acc)
Rater 1	98.9	97.6	96.6	94.2	99.0	99.1	97.7
Rater 2	99.0	98.1	94.2	97.0	91.3	97.4	96.5
Rater 3	97.9	92.0	97.1	98.7	95.7	94.2	95.7
Avg.	98.6	95.9	96.0	96.6	95.3	96.9	96.6
Agreement	95.8	87.8	76.5	90.1	87.0	90.7	90.3

Table 25. **Human evaluation results.** Each rater independently selected correct answer(s) for each question. **Agreement** indicates proportion of questions for which all three raters selected exactly same set of options.

were unusable.

For **Object Parts** questions, clips often involve object state changes, which change the appearance of the components. In such cases, both the frames before and after the state change were selected.

Human Evaluation. We recruited three human raters who each passed the initial screening, achieving over 70% accuracy on a set of 30 questions sampled from the validation split of our dataset. Table 25 reports the performance of the raters and their agreement ratio. Agreement denotes the proportion of questions for which all three raters selected exactly the same set of options. The **Objects** category shows relatively lower agreement because multiple answers may be correct, reducing the likelihood that all raters choose the same set of options. Overall, humans performed over 90% accuracy/AP across all the categories, indicating that our benchmark is solvable by humans and that the current models still have a significant gap between human performance.

Detailed dataset statistics. Figure 10 shows the detailed distribution of the scenarios and the primary verbs included in HanDyVQA. While the scenarios reflect the distribution of the original Ego4D video clips, the verbs are more uniformly selected to ensure diverse HOIs are covered.

Figure 11 shows the spatial and temporal distribution of segmentation mask annotations for **Objects** and **Object Parts** questions. While selected frames are biased towards the beginning and the end of a video, the remaining annotations are evenly distributed throughout the video. Regarding the spatial direction, the segmentation showed a tendency to concentrate in the center of a frame, but also appeared to spread around the center.

Details on compensation. We outsourced the annotation of MCQs and segmentation masks to an agency at a total cost of 3,180,000 JPY (approximately 22,000 USD). For the human evaluation, we commissioned a different company and hired separate personnel as raters, incurring an additional cost of 4,510,585 JPY (approximately 28,600 USD). The company is responsible for managing payments to annotators, ensuring compliance with the minimum wage regulations in the annotator's country.

C. Details on experimental settings

C.1. Multi-Choice Questions

Frame sampling strategy. We sample n frames uniformly from a video of length L by dividing it into n equal segments and selecting one frame from the center of each segment. Specifically, the sampling index i_k for the k-th frame ($k = 0, 1, \ldots, n-1$) is computed as:

$$i_k = \left\lfloor k \cdot \text{gap} + \frac{\text{gap}}{2} \right\rfloor, \quad \text{where gap} = \frac{L}{n}$$

This approach ensures that the sampled frames are evenly distributed over the entire video, while avoiding bias toward the start or end. By choosing the center of each interval, we obtain a more representative snapshot of the temporal progression.

Prompts for zero-shot evaluation. The textual prompts fed to video-language models with integrated LLMs to solve MCQs are shown in Table 28 and Table 29.

Computational cost. The 7B-size video-language models fit on a single NVIDIA H200 (141GB) GPU and completed inference for each question category in under an hour. The 72B-size models were able to run on a single node with eight H200 GPUs, requiring approximately 1.7 hours per category.

C.2. Referring Video Object Segmentation

Frame sampling strategy. Given a set of annotated frame indices $\mathcal{A} \subseteq [0, n]$ and a target number of samples l, we construct a set $\mathcal{S}$ of l indices that are both representative and temporally balanced.

- We initialize the set $\mathcal{S}$ as the sorted, unique subset of annotated indices $\mathcal{A}$ within their valid range:

$$\mathcal{S} \leftarrow \text{sorted}\left(\{x \in \mathcal{A} \mid 0 \le x \le n\}\right)$$

- While $|\mathcal{S}| < l$, we iteratively identify the largest temporal gap between consecutive elements in $\mathcal{S}$, including gaps at the start ($[0, \mathcal{S}_1]$) and end ($[\mathcal{S}_{|\mathcal{S}|}, n]$), and insert the midpoint of the largest such gap:

$$\text{midpoint} = \left\lfloor \frac{i+j}{2} \right\rfloor \quad \text{for each gap } [i, j]$$

- This process continues until $|\mathcal{S}| = l$, or no more meaningful midpoints can be added.
- If the final size $|\mathcal{S}| > l$ (e.g., due to duplicate insertions), we resample l indices from $\mathcal{S}$ to evenly cover $[0, n]$. Specifically, we define l ideal positions:

$$t_i = \text{round}\left(\frac{i \cdot n}{l-1}\right), \quad i = 0, 1, \ldots, l-1$$

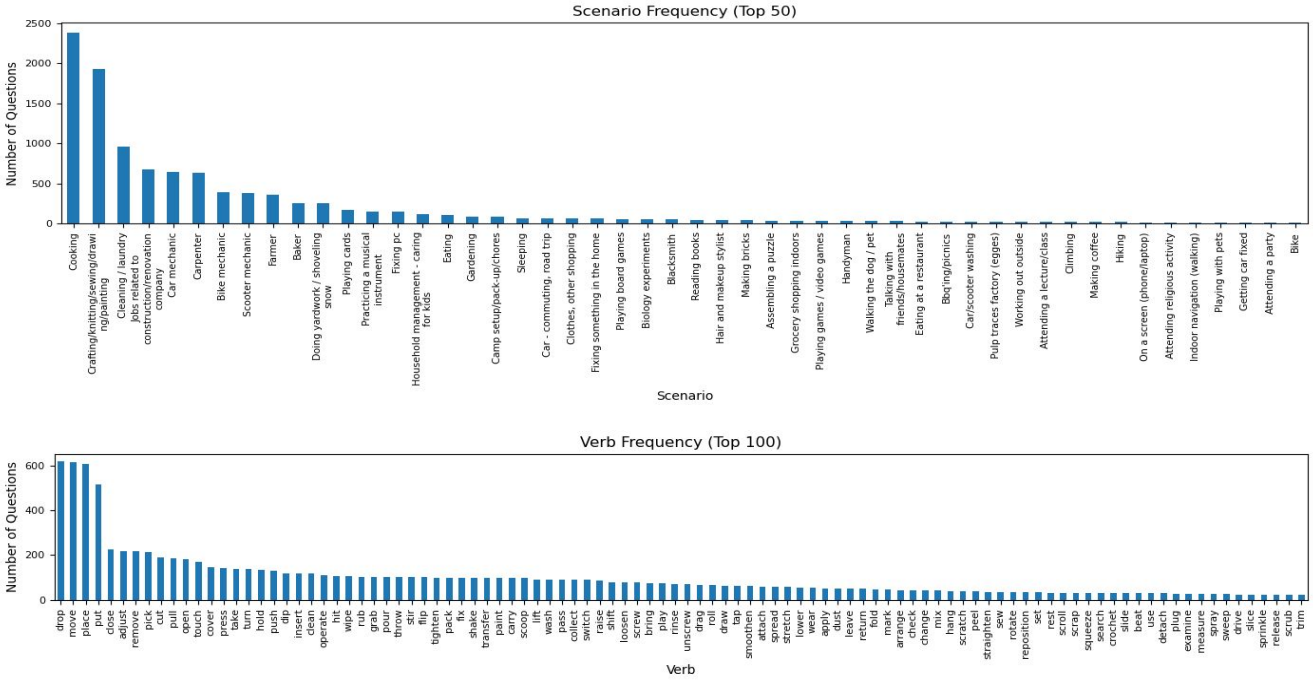

Figure 10. Top-50 scenario distribution (top) and top-100 verb frequency distribution (bottom) of HanDyVQA.

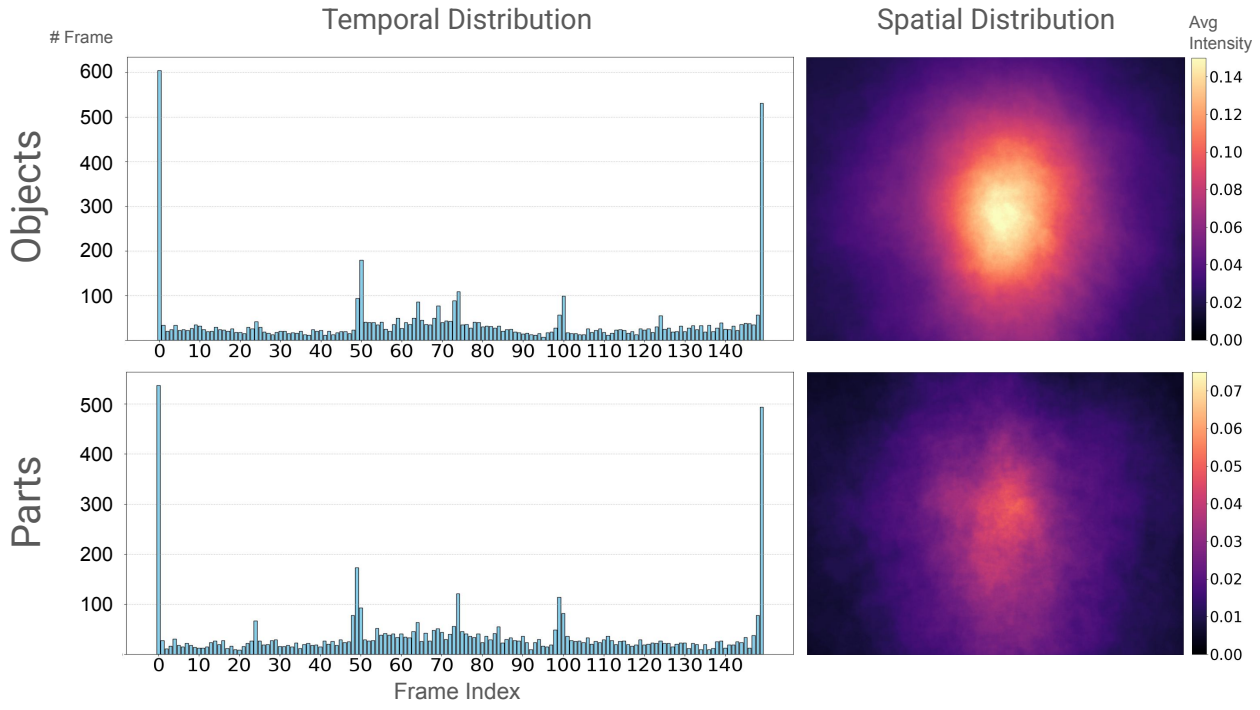

Figure 11. Temporal distribution (right) and spatial distribution (left) of mask annotations per category.

and for each t_i, we select the closest available index in $\mathcal{S}$ without duplication.

This strategy ensures that manually annotated indices are selected while interpolating additional indices to maximize temporal coverage.

Prompts for zero-shot evaluation. The prompt used for the baseline that takes questions as input is provided in Table 30. The prompt that uses the ground-truth option is provided in Table 31.

Grouping for evaluation. We group videos into three size bins—small (S), medium (M), and large (L)—based on the average area of their ground-truth masks, so we can examine performance as a function of target size. The S/M/L thresholds differ between the **Area** and **Object** settings:

- **Area**: S→M at 372, M→L at 2,127 (pixels)
- **Object**: S→M at 3,581, M→L at 13,063 (pixels)

Computational cost. Inference with the VideoLISA-3.8B model was performed on a single NVIDIA H200 GPU and took roughly 1.5 hours per category. Inference with the Sa2VA-8B model on a single NVIDIA H200 GPU required about 2 hours per category for the video baseline and around 1.5 hours for the frame-wise baseline.

C.3. Integration of HOI cues

Implementation details. Figure 12 illustrates the architecture of our model, which integrates multiple modalities: frame-level RGB visual features, hand poses, bounding boxes of manipulated objects, and visual features of the manipulated objects.

Hand poses are extracted using an off-the-shelf 3D hand pose detector [31], resulting in a tensor of shape $[B, T, 21, 3]$ for each hand, where B is the batch size and T is the number of frames in the video. The bounding boxes of manipulated objects for each hand are obtained using AMEGO [14] and represented as $[B, T, N, 4]$, where N is the maximum number of objects per hand. We set $N = 8$ in our experiments. For each detected bounding box, we crop the corresponding image region and extract CLIP features [32].

For each modality, we use separate processing modules for the left and right sides, resulting in seven feature vectors in total. All modality-specific features, except for the global RGB visual feature, have a shape of $[B, 128]$. Before concatenation, we apply a modality-dropout layer in which entire modalities (i.e., both left- and right-hand streams of a modality) may be jointly dropped. Up to two such modalities are randomly dropped with a probability of $p = 0.2$, encouraging the model to rely on the remaining modalities. Finally, all features are concatenated and passed through a multi-layer perceptron (MLP) to produce the final fused representation of shape $[B, 512]$.

Training. For each integrated feature $\mathbf{v}_i$, we associate one positive text feature and B_N negative text features sampled from (i) distractors within the same video and (ii) answer options of other videos in the batch. The contrastive training is summarized in Algorithm 1.

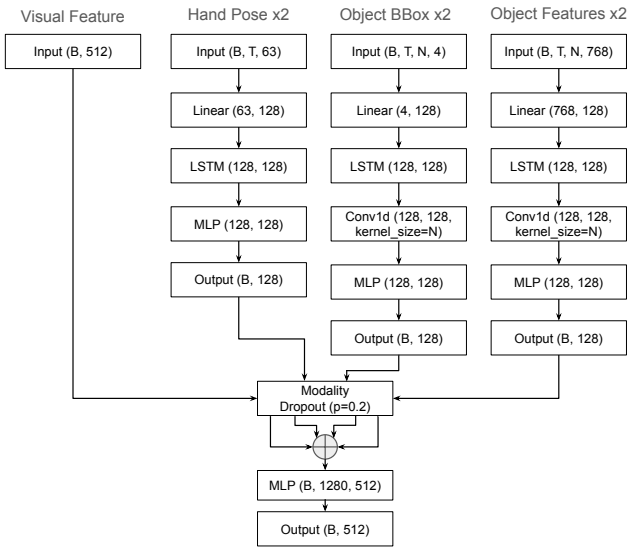

Figure 12. Architecture of additional head used for fine-tuning experiment. Branches are omitted for unused modalities.

Algorithm 1 Contrastive Training for Integrated Features

Require: Integrated feature $\mathbf{v}_i$, positive text feature $\mathbf{p}_i$, negative features $\{\mathbf{n}_{i,j}\}_{j=1}^{B_N}$, temperature τ

$\hat{\mathbf{v}}_i \leftarrow \text{normalize}(\mathbf{v}_i)$

$\hat{\mathbf{p}}_i \leftarrow \text{normalize}(\mathbf{p}_i)$

for $j = 1$ to B_N **do**

  $\hat{\mathbf{n}}_{i,j} \leftarrow \text{normalize}(\mathbf{n}_{i,j})$

end for

Compute logits:

$$\mathbf{s}_i = \left[\frac{\hat{\mathbf{v}}_i^\top \hat{\mathbf{p}}_i}{\tau}, \frac{\hat{\mathbf{v}}_i^\top \hat{\mathbf{n}}_{i,1}}{\tau}, \ldots, \frac{\hat{\mathbf{v}}_i^\top \hat{\mathbf{n}}_{i,B_N}}{\tau} \right]$$

$\mathbf{y}_i \leftarrow [1, 0, \ldots, 0]$

$\mathcal{L}_i \leftarrow \text{BCEWithLogits}(\mathbf{s}_i, \mathbf{y}_i)$

The total loss is computed by averaging across all samples in the batch:

$$\mathcal{L} = \frac{1}{B} \sum_{i=1}^{B} \mathcal{L}_i.$$

We trained separate models for each category. The batch size was set to 16, the learning rate to 1×10^{-5}, the weight decay to 1×10^{-4}, the number of negative samples B_N to 16, and the temperature τ to 0.07. Each model was trained for 500 epochs, and the best-performing checkpoint was selected based on validation performance.

Computational cost. As shown in Table 26, the additional branches for HOI cues introduce only minor computational overhead. Table 27 further reports the cost of extracting each HOI cue. For training, all the model variants fit on a single

	Extra Params	Extra GFLOPs / video
Hand Pose Branch	0.80M	0.0052
Object BBox Branch	0.92M	0.0351
Object Feat Branch	1.02M	0.0602

Table 26. Computational cost of additional layers for HOI cues.

	Params	GFLOPs / frame
Hand Pose (YOLOv8 det. + WiLoR pose)	26.63M + 693.03M	41.56 + 140.09
Object BBox (Faster R-CNN, ResNet-101)	47.36M	219.57
Object Feats (CLIP)	202.05M	51.90

Table 27. Computational cost of feature extraction for HOI cues.

NVIDIA H200 GPU and finish in roughly 0.05–2.5 hours per category, depending on the number of additional cues.

Carefully watch the first-person view video and pay attention to the cause and sequence of events, the details and movements of objects, and the actions and poses of persons.
Question: {question}
Choose **only one** option from the following list.
Options:
(A) {option1}
(B) {option2}
(C) {option3}
(D) {option4}
(E) {option5}

Answer format:
(A) <Description of Option A>

Table 28. Prompt for zero-shot evaluation of video-language models integrated with LLMs when there is only one correct answer. {question} is replaced with question and {option$n$} is replaced with n-th option.

Carefully watch the first-person view video and pay attention to the cause and sequence of events, the detail and movement of objects, and the action and pose of persons.
Question: {question}
Choose **all** options that apply from the following list.
Options:
(A) {option1}
(B) {option2}
(C) {option3}
(D) {option4}
(E) {option5}
...

Answer format:
(A) <Description of Option A>

Table 29. Prompt for zero-shot evaluation of video-language models integrated with LLMs when there are multiple correct answers. {question} is replaced with question and {option$n$} is replaced with n-th option.

Segment the area that corresponds to the answer to the question.
Question: {question}

Table 30. Prompt for Sa2VA (frame-wise/video) baseline in referring video object segmentation. {question} is replaced with question.

Segment all the mentioned area:
{GT}

Table 31. Prompt for GT + Sa2VA baseline in referring video object segmentation. {GT} is replaced with ground truth.

Models	Action (Acc)	Process (Acc)	Objects (AP)	Location (Acc)	State (Acc)	Parts (Acc)	Avg. (Acc)
Qwen2.5-VL-7B zero-shot	**58.6**	53.4	53.5	45.5	57.1	47.8	52.5
Qwen2.5-VL-7B fine-tuned	58.2	**64.9**	**57.5**	**64.8**	**63.1**	**50.2**	**60.2**

Table 32. Results of fine-tuning Qwen2.5-VL-7B model on HanDyVQA.

D. Fine-tuning Qwen2.5-VL-7B

We conducted instruction tuning on the training split of HanDyVQA to explore the effectiveness of fine-tuning. Specifically, we used Qwen2.5-VL-7B as the base model and trained LoRA adapters using QLoRA [9].

Implementation details. We trained separate models for each category. We only trained the LoRA parameters injected into the query and value projection layers (q_proj and v_proj) of the attention modules, while keeping all other model weights frozen. The number of input video frames is 16, and the resolution is 224×398. We set the batch size to 4, learning rate to 1×10^{-4}. Each model was trained for 150 epochs, and the best-performing checkpoint was selected based on the validation performance.

Computational cost. Training was performed on a single NVIDIA H200 GPU and took roughly two hours per category.

Results. Table 32 shows the results of fine-tuning the model. The **Process** and **Location** categories show improvements of roughly 10 points, followed by a 6-point gain in the **State** category, while gains in **Objects** and **Parts** remain below 5 points. A slight performance drop is observed for the **Action** category. These results indicate that small-scale instruction tuning offers limited benefits for relatively easier conventional tasks such as **Action** and **Objects**, but yields larger gains for categories that involve longer textual descriptions, likely because LLMs are better at leveraging textual biases.

E. Broader Impacts

The proposed HanDyVQA dataset provides a detailed evaluation of fine-grained hand-object interactions. As such, it serves as a valuable benchmark for systems designed to assist human workers using visual information captured by wearable cameras in diverse real-world scenarios [29]. This enables the development of systems that can better understand subtle interactions and deliver more accurate and context-aware feedback to the users.

Such recognition capabilities are also essential for applications in Augmented Reality (AR) and Virtual Reality (VR), where systems must respond to users actions and changes in the environment in real time. Unlike previous datasets that focus primarily on action recognition or object detection, HanDyVQA offers a unique benchmark that evaluates a model's ability to comprehend nuanced hand-object interactions and underlying dynamics, pushing the boundaries beyond conventional video recognition tasks.