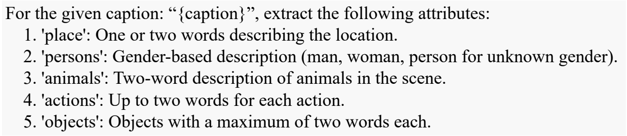
VideoSetDiff: Identifying and Reasoning Similarities and Differences in Similar Videos (Supplementary Material)

I. Additional Dataset Details

Video-set collection from HowTo100M (Figure 2 (a-i) in the main paper). The evaluation split comprises 493 instructional video sets from HowTo100M and 290 video sets from Panda70M. The HowTo100M sets are selected manually. Each set contains four videos depicting the same daily activity (*e.g*. "make shrimp cocktail," "brush a long-haired dog"). To ensure broad coverage, we choose tasks from diverse categories (*e.g*. "Family Life," "Hobbies and Crafts"). Within every category we select the tasks with the largest numbers of videos that are suitable for video-set reasoning. The categories and corresponding tasks are listed in Table 4. For each task we retain one or two video sets, each with four videos showing similar activities.

Video-set collection from Panda70M and HT-Step (Figure 2 (a,b-i)). The training split contains 1000 video sets built from HT-Step and 2000 video sets built from Panda70M; the evaluation split also includes 290 additional Panda70M sets. These sets are created automatically. In both datasets every short clip (around 10 seconds) has a caption. We parse each caption into five attribute fields—*place*, *persons*, *animals*, *actions*, and *objects*—using GPT-4 (prompt described in Figure 8). Each field is vectorised with a bag-of-words model, cosine similarities are computed, and the field-wise similarities are combined with heuristic weights ($3 : 3 : 3 : 2 : 1$). We then greedily group captions into sets of six clips, ensuring that the clips in a set come from different videos and are mutually similar; finally, we keep up to four clips per set for downstream use.

Set diff/sim annotation (Figure 2 (a-ii)). To cover a broad spectrum of video details, we predefined nine top-level categories and 27 sub-categories (Table 3). Annotators were asked to choose ten sub-categories for each video set and to write one sentence that captures both the similarities and differences across the four videos. Typical examples are: "Videos 1 and 2 have a static viewpoint, whereas in Videos 3 and 4 the camera slowly moves toward the person," and "All videos show someone working at a cupboard, but with different groceries, cooking tools, and utensils." Twenty-two annotators participated. For each video set, one annotator produced the initial description, and two other annotators verified it. Only descriptions confirmed by both verifiers were retained for the QA-generation stage of the evaluation set.



For the given caption: "{caption}", extract the following attributes:
1. 'place': One or two words describing the location.
2. 'persons': Gender-based description (man, woman, person for unknown gender).
3. 'animals': Two-word description of animals in the scene.
4. 'actions': Up to two words for each action.
5. 'objects': Objects with a maximum of two words each.

Figure 8. Prompt illustration for caption-to-list generation.

Category	Sub-category
Action	Category, components, attributes, order
Hand	Category, pose
Human	Existence, attributes, pose, facial expression, emotion, relationships
Object	Category, existence, attributes, location, state
Tool	Category, usage, existence, attributes
Place	Category, attributes
Time	Time
Viewpoint	Direction, transition
Motivation	Motivation

Table 3. Categories and their corresponding sub-categories for manual annotations.

QA generation (Figure 2 (a,b-iii)). We employed the GPT-4 model to generate QAs from (1) human-annotated similarity-and-difference descriptions (a-iii, evaluation set) and (2) automatically generated captions and transcripts (b-iii, training set). The corresponding prompts appear in Figure 10 and Figure 11, respectively.

QA validation (Figure 2 (a-iv)). After generating the QAs, the evaluation set undergoes human validation. Each QA pair is answered independently by two annotators. Annotators either supply an answer or mark the question as unanswerable and state the reason. Only items whose generated answer matches the answers from both annotators are kept in the final evaluation set. Inter-annotator agreement is 90.7% for answerability and 86.8% for the answers.

Question type definition. Binary/multiple-choice ques-

Category	Task
Family Life	Make A Baby Sling Without Sewing , Make A Baby Shower Towel Cake , Change A Diaper , Make Baby Wipes , Set Up A Baby Crib , Make Apple Baby Food
Hobbies and Crafts	Perform A Spread With Cards , Join A New Yarn Ball While Knitting , Attach Granny Squares , Half Double Crochet (Hdc) , Perform A Card Trick Using Math , Make Wrist Wraps
Computers and Electronics	Install A Zagg Invisible Shield On An Iphone , Clean A Mechanical Keyboard , Clean A Macbook Or Macbook Pro Computer , Install Cable Television , Clean A Computer Monitor
Personal Care and Style	Remove Sideburns (For Girls) , Trim A Mustache , Shave With An Electric Shaver , Make Stick Deodorant , Do A Spiral Perm , Get Bouncy Beach Curls , Dye Your Beard , Clean Mac Makeup Brushes , Make Half Ponytail Hairstyles , Create A Smoky Eye Effect , Give A Manicure
Holidays and Traditions	Transfer Images To Easter Eggs , Make A Christmas Centrepiece , Make Christmas Crackers , Make An Envelope Advent Calendar , Make A Freddy Krueger Glove
Food and Entertaining	Make Chicken Cacciatore , Cook Cornish Game Hens , Cook Tomahawk Ribeye Steak , Make Shrimp Cocktail , Make A Unicorn Frappuccino , Cook Walleye , Make Beer Can Chicken , Make Tinga De Pollo , Make Celery Juice
Youth	Keep White Adidas Superstar Shoes Clean , Make Your Lips Look Great (For Girls) , Wash Your Face (Teens) , Do Braided Double Buns
Pets and Animals	Build A Snake Cage , Brush A Long Haired Dog , Groom A Cat , Clip Your Horse , Apply A Horse Tail Bandage , Tack Up A Horse , Trim A Dog'S Nails
Cars and Other Vehicles	Change A Sway Bar Link , Replace A Car'S Side View Mirror , Replace Tie Rod Ends , Remove A Door Panel From A Car , Test An Ignition Coil , Replace Shocks , Fix A Leaky Sunroof
Health	Whiten Teeth , Apply A Tourniquet , Use A Defibrillator , Give A Glucagon Shot
Work World	Milk A Goat By Hand , Milk A Cow With A Milking Machine , Harvest Sugarbag Honey From A Native Australian Beehive , Change A Typewriter Ribbon
Home and Garden	Change The Oil In A Lawn Mower , Install A Tankless Hot Water Heater , Install Electric Radiant Heat Mat Under A Tile Floor , Sharpen Lawn Mower Blades , Sharpen A Chainsaw
Sports and Fitness	Reload Ammo , Load A Shotgun , Wrap A Hockey Stick
Philosophy and Religion	Make A Rosary , Draw The Star Of David

Table 4. Tasks used in the video set construction of the VideoSetDiff.

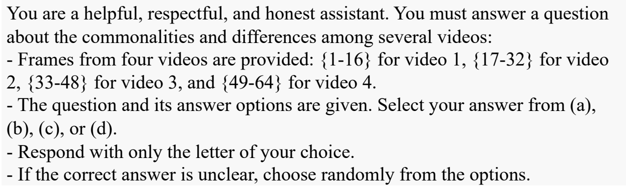

You are a helpful, respectful, and honest assistant. You must answer a question about the commonalities and differences among several videos:
- Frames from four videos are provided: {1-16} for video 1, {17-32} for video 2, {33-48} for video 3, and {49-64} for video 4.
- The question and its answer options are given. Select your answer from (a), (b), (c), or (d).
- Respond with only the letter of your choice.
- If the correct answer is unclear, choose randomly from the options.

Figure 9. Prompt illustration for video-based LLMs.

Encoder	Overall (binary / MC)
TimeSformer (vanilla) [1]	57.1 (67.4 / 46.7)
TimeSformer (w/ KB Transfer) [2]	58.6 (66.2 / 50.8)
TimeSformer (w/ ASR) [3]	60.2 (65.9 / 54.3)
InternVideo2.5 (original)	68.6 (74.2 / 62.7)

Table 5. Results obtained with different video encoders. MC: multiple-choice questions.

tions: choose an answer from Yes/No or four options; recognition/reasoning questions: questions that require direct video content from one or more videos, or more complex comparison or reasoning across multiple videos. We used GPT-4 to determine whether a question is a recognition or a reasoning question.

II. Additional Experiment Details

Evaluation prompts. Previous video-based LLMs typically accept only a single video as input. To evaluate these models on the VideoSetDiff evaluation set, we sample frames from all four videos in each set and concatenate them into one composite clip, arranging the frames from Video 1 to Video 4 in sequence. The prompt used for these video-based LLMs is shown in Figure 9.

Additional baseline setups. Fine-grained video recognition is crucial for detecting similarities and differences across multiple videos. Recent work [2, 3] explores step-level video–text alignment. To evaluate this aspect, we replaced VidSetReasoner's encoder with three TimeSformer [1] variants—vanilla, plus the procedure-aligned models from [2] and [3]—and trained them under identical settings (Table 5). Both procedure-aligned encoders outperformed the vanilla TimeSformer, yet they still lagged behind the InternVideo-2.5 encoder. These results suggest that large-scale pre-training remains essential and could be further enhanced by step-level supervision. In Table 2 of the main paper (rows 2–6), the models use either image captions or video captions as input, followed by GPT-4 reasoning. Inspired by [4], we hypothesized that combining video-level captions with image-level descriptions could further enhance performance. We therefore built a new ensemble: InternVideo2.5 generates the video-level caption, InstructBLIP adds four-frame image-level captions for each video, and GPT-4 answers the questions. This ensemble achieves slightly higher performance than "InternVideo 2.5 captions + GPT-4," (row 6 in Table 2) but still lags behind the model that lets GPT-4 answer directly from frames via InternVideo2.5 (row 11 in Table 2), confirming that large-scale pre-training already captures most details.

Subsampling influences. VideoSetDiff includes diverse questions that demand fine-grained video understanding. Heavy subsampling during experimentation can blur subtle actions. Figure 5 of the main paper shows that simply adding more frames seldom improves accuracy. We also tested up to 256 frames and a resolution of 672 px; most models peaked at 32–64 frames and 224–448 px. These results suggest that the bottleneck lies in fine-grained reasoning rather than in missing pixels.

References

[1] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, 2021. 2

[2] Xudong Lin, Fabio Petroni, Gedas Bertasius, Marcus Rohrbach, Shih-Fu Chang, and Lorenzo Torresani. Learning to recognize procedural activities with distant supervision. In *CVPR*, pages 13853–13863, 2022. 2, 3

[3] Yiwu Zhong, Licheng Yu, Yang Bai, Shangwen Li, Xueting Yan, and Yin Li. Learning procedure-aware video representation from instructional videos and their narrations. In *CVPR*, pages 14825–14835, 2023. 2, 3

[4] Zhenhailong Wang, Manling Li, Ruochen Xu, Luowei Zhou, Jie Lei, Xudong Lin, Shuohang Wang, Ziyi Yang, Chenguang Zhu, Derek Hoiem, et al. Language models with image descriptors are strong few-shot video-language learners. *NeurIPS*, 35:8483–8497, 2022. 3

You are an AI assistant that generates an advanced and challenging Yes/No question-answer pair based on the given description from multiple videos.
Instructions:
1. Generate one Yes/No question-answer pair from the description with the correct answer being {yes_no_correct_answer}.
2. Ensure the question requires synthesizing specific visual information from multiple videos, making it more challenging but avoid general or overly broad questions.
- The question should involve visual comparisons, observations, or understanding of processes across multiple videos.
- Avoid questions that require abstract reasoning, inferences about efficiency, safety, or any concepts not directly observable.
- Avoid questions answerable by referencing only a single piece of information.
3. The question should have a corresponding answer list:
- [Yes, No]
4. The correct answer should be included in the answer list.
5. Ensure that the question is not trivial and requires understanding visual interplay.
6. Present the output in the following JSON format without any additional text:
{{question: Your question here,
  answer_list: [Yes, No],
  answer: Correct Answer}}
7. Do not include code fences or markdown formatting in your response. Output only the JSON object.
8. Avoid overly general or vague questions; focus on specific visual details from the description.
9. Avoid non-visual or abstract questions.

Additional Requirements:
- DO NOT invent details beyond what the data supports. If data is not sufficient, state 'No direct information is provided.'

(a)

You are an AI assistant that generates an advanced and challenging Multiple-Choice question-answer pair based on the given description from multiple videos.
Instructions:
1. Generate one Multiple-Choice question-answer pair from the description.
2. Ensure the question requires synthesizing specific visual information from multiple videos, making it more challenging but avoid general or overly broad questions.
- The question should involve visual comparisons, observations, or understanding of processes across multiple videos.
- Avoid questions that require abstract reasoning, inferences about efficiency, safety, or any concepts not directly observable.
- Avoid questions answerable by referencing only a single piece of information.
3. The question should have a corresponding answer list with four options:
- One correct answer and three plausible but incorrect options.
- Do not use We don't know as an option or as the correct answer.
4. The correct answer should be included in the answer list.
5. Ensure that the question is not trivial and requires understanding visual interplay.
6. Present the output in the following JSON format without any additional text:
{{question: Your question here,
   answer_list: [Option1, Option2, Option3, Option4],
   answer: Correct Answer}}
7. Do not include code fences or markdown formatting in your response. Output only the JSON object.
8. Avoid overly general or vague questions; focus on specific visual details from the description.
9. Avoid non-visual or abstract questions.

Additional Requirements:
- DO NOT invent details beyond what the data supports. If data is not sufficient, state 'No direct information is provided.'

(b)

Figure 10. Prompt illustration of binary- and multiple-choice QA generation for the evaluation set.

You have the above textual information describing four videos. Your task is to produce a single json array. Each element in this array corresponds to exactly one of the templates (1 to 12) listed below. For each template, produce exactly 2 examples, so the final json array has 24 items total (2 for each template x 12 templates). When creating examples for templates 4..12 (the question-answer templates): About half should be yes/no questions, and about half should be multiple-choice (4 options, exactly one correct). For multiple-choice questions, produce varied answer lists, sometimes short references (e.g. ['video 1','video 2','video 3','video 4']), other times descriptive sentences.
Here are the 12 templates you must fill:

1) Template 1 (Similarities/differences among all 4 videos):
 - Instruct: e.g. ‘Describe the similarities and differences of the four videos in action in detail.’
 - Attribute: One of [action | tool | hand | object | human | time | place | motivation | viewpoint].
 - Correct answer: A factual statement describing the similarities/differences.
2) Template 2 (How a single video differs from the remaining videos):
 - Instruct: e.g. ‘Describe how video 2 differs from the remaining videos in detail.’
 - Attribute: One of [action | tool | hand | object | human | time | place | motivation | viewpoint]
 - Correct answer: Focus on how one video is distinct.
3) Template 3 (Find the most distinct video among the 4):
 - Instruct: e.g. ‘Find the video that differs the most from the remaining videos in action.’
 - Attribute: One of [action | tool | hand | object | human | time | place | motivation | viewpoint]
 - Correct answer: Identify which video is distinct, describe how.
4) Template 4 (Question-Answer about action):
 - Question: A yes/no or multiple-choice (4 options) question.
 - Answer list: ['yes','no'] or 4 distinct choices.
 - Correct answer: Must match exactly one entry from Answer list.
5) Template 5 (Question-Answer about hand):
 - Question: A yes/no or multiple-choice.
 - Answer list: ['yes','no'] or 4 options.
 - Correct answer: Must match an entry in Answer list.
6) Template 6 (Question-Answer about object):
 - Question: A yes/no or multiple-choice.
 - Answer list: ['yes','no'] or 4 options.
 - Correct answer: Must match an entry in Answer list.
7) Template 7 (Question-Answer about tool):
 - Question: A yes/no or multiple-choice.
 - Answer list: ['yes','no'] or 4 options.
 - Correct answer: Must match an entry in Answer list.
8) Template 8 (Question-Answer about human):
 - Question: A yes/no or multiple-choice regarding humans or their attributes in the videos.
 - Answer list: ['yes','no'] or 4 options.
 - Correct answer: Must match an entry in Answer list.
9) Template 9 (Question-Answer about time):
 - Question: A yes/no or multiple-choice regarding time or season.
 - Answer list: ['yes','no'] or 4 options.
 - Correct answer: Must match an entry in Answer list.
10) Template 10 (Question-Answer about place):
 - Question: A yes/no or multiple-choice regarding places.
 - Answer list: ['yes','no'] or 4 options.
 - Correct answer: Must match an entry in Answer list.
11) Template 11 (Question-Answer about motivation):
 - Question: A yes/no or multiple-choice regarding motivations.
 - Answer list: ['yes','no'] or 4 options.
 - Correct answer: Must match an entry in Answer list.
12) Template 12 (Question-Answer about viewpoint):
 - Question: A yes/no or multiple-choice regarding camera viewpoint.
 - Answer list: ['yes','no'] or 4 options.
 - Correct answer: Must match an entry in Answer list.

Additional Requirements:

- DO NOT invent details beyond what the data supports. If data is not sufficient, state 'No direct information is provided.’

Figure 11. Prompt illustration of binary- and multiple-choice QA generation for the training set.