Taming Hallucinations: Boosting MLLMs' Video Understanding via Counterfactual Video Generation

Supplementary Material

A. Datset Detail

We categorize video anomalies into three levels: Visual anomalies refer to pixel-wise distortions, including abnormal contrast, saturation, brightness, blurring, and local distortions, etc., which primarily affect visual quality without explicit semantic alteration. Semantic anomalies involve violations of scene semantics, such as object disappearance, unexpected object emergence, and object substitution, which result in temporal inconsistencies. Commonsense anomalies capture more abstract and holistic violations involving spatio-temporal or physical implausibility, such as unnatural deformations, implausible object movements, unreasonable interaction and human motion anomalies, etc.

A.1. DualityForge

Table A.1. Definitions of video anomaly categories.

Category	Definition
Visual	Pixel-wise distortions that primarily affect visual quality without explicit semantic alteration. These include abnormal contrast, saturation, brightness, blurring, and local distortions.
Semantic	Violations of scene semantics, such as object disappearance, unexpected object emergence, and object substitution, resulting in temporal inconsistencies.
Commonsense	Abstract and holistic violations involving spatio-temporal or physical implausibility ($e.g.$, unnatural deformations, implausible object movements, unreasonable interactions, and human motion anomalies).

Video Source. To improve video-editing quality and dataset diversity, we adopt two widely used public datasets Pexels [2] and OpenVid [5] which are commonly employed in video-generation research. From OpenVid, we randomly sample around 3,000 videos from each of the 20 most populated categories, yielding a candidate pool of 61,591 clips. From Pexels, we additionally sample 36,333 clips, for a total of 97,924 videos.

Visual anomalies. We employ OpenCV to synthesize visual anomalies within the video data. We divide visual anomalies into **entire-frame** level, **region** level, and **object** level. To introduce anomalies, we randomly select a temporally consistent segment in which to insert visual perturbations. At the object level, we first extract all noun entities present in the video and randomly select one object. Then we utilize Grounding DINO[4] and SAM[6] to localize the position of the selected object, on which the visual anomaly synthesis operation is performed.

Semantic anomalies. We categorize semantic anomalies to include both the temporal instability of entities ($e.g.$, unexpected appearance, disappearance, or substitution) and appearance-level abnormalities (such as unreadable text or blurred faces). To enable controlled injection of anomalies into the video while keeping the other part unchanged, we utilize the advanced video editing model, VACE[3], to edit the specific area in the video.

Common sense anomalies. We categorize anomalies that contradict common sense into the following types: violations of physical laws, causal inconsistencies, material abnormalities, and abnormal human movements. To introduce the first three types of anomalies into videos, we first employ a Multimodal Large Language Model (MLLM) to analyze the visual elements within an image and generate an editing instruction targeting the anomaly. Next, we use FLUX-Kontext[1] to edit the image according to this instruction. After validating the edited image, we create a video by performing frame interpolation with VACE using the original and edited image pair.

Finally, we collect a total of 135, 168 videos with anomalies, which are subsequently subjected to an additional screening process to ensure quality prior to their use in QA construction. The statistics of video types are shown in Tab. A.2. This takes around 40k GPU hours on NVIDIA H20 GPUs.

Table A.2. Video dataset type statistics

Type	Count
color	27,353
replacement	9,961
appearance	6,092
disappear	5,016
common sense	86,746
All	135,168

A.2. DUALITYVIDQA

Training Data Construction. To enhance VLM counter-commonsense reasoning while preserving general VideoQA performance, we adopt a two-stage training framework: Supervised Fine-Tuning (SFT) and Reinforcement Learning (RL). For each stage, we curate a tailored dataset to support its specific training objective. We conducted **two rounds** of data curation to ensure optimal training quality. In our first round, we constructed initial datasets for both SFT and RL stages. We generated 200k QA pairs from 80k videos. After analyzing the training performance,

we observed that samples with zero reward were predominantly associated with failed video edits where no meaningful visual changes were created. Thus, we use the first stage trained model to filterout around 30% of the samples with zero reward and low-quality video. This insight led us to create a refined dataset through the following process:

Table A.3. Question type frequency statistics

QA Type	Real Video	Counterfactual Video
Multiple Choice	12,210	10,224
Open-Ended	42,669	39,776
All	54,879	50,000

(1) SFT data construction through two stages: dense captioning and question-answer (QA) generation. During dense captioning, a red box is used to indicate the anomaly region, and video editing metadata is provided to the model to generate detailed, high-coverage captions under controlled conditions. The detailed prompt is **Dense Caption Prompt Template** in Fig. A.1. During QA generation, we followed LLaVA-Video, categorizing questions into 16 types and using GPT-5 and Gemini 2.5 Pro to generate questions and answers based on video content and dense captions. To ensure diversity and stability, we sampled 5,000 examples from LLaVA-Video's 170k dataset as a pool, randomly selecting three same-category examples

Table A.4. 16 Question type frequency statistics with descriptions

QA Type	Real Video	Counterfactual Video	Description
Attribute Change	1,436	8,674	Questions about changes in attributes of objects or characters between scenes or frames.
Binary	2,009	1,009	Involves yes or no questions related to the video content.
Camera Direction	1,601	4,887	Tests understanding of the camera's movement or shooting direction within the video.
Causal	737	216	Focuses on explaining actions/events, determining intentions of actions or causes for events.
Count	363	438	Tests ability to count instances of objects, people, or actions.
Description Human	15,360	4,324	Involves describing actions or attributes of people.
Description Object	8,450	4,404	Assesses ability to describe attributes of objects.
Description Scene	19,067	8,317	Assesses ability to describe the major scene of the video.
Fine-grain Action Understanding	811	1,303	Creates questions challenging comprehension of subtle actions.
Non-Existent Actions with Existent Scene Depictions	29	113	Tests ability to identify actions that did not occur despite related scene elements being present.
Object Direction	420	3,374	Tests understanding of the movement or facing direction of objects within the video.
Plot Understanding	981	151	Challenges ability to interpret the plot in the video.
Spatial	2,074	8,641	Tests ability to perceive spatial relationships between observed instances in a video scene.
Speed	221	998	Involves estimating or comparing the speed of moving objects or actions.
Temporal	768	2,789	Designed to assess reasoning about temporal relationships between actions/events.
Time Order Understanding	552	362	Tests comprehension of the chronological order of events or actions in the video.
All	54,879	50,000	Aggregate counts for all question types.

at each generation step as in-context references to maintain stylistic consistency and content diversity. Finally, we curated 25K real videos and 25K edited videos, generating 100K QA pairs with an 8:2 ratio of open-ended to multiple-choice items using **Real Video QA Generation Prompt Template** in Fig. A.3 and **Counterfactual Video QA Generation Prompt Template** in Fig. A.2 respectively. Then we use GPT-4o to classify each QA into question types based on the LLaVA-Video taxonomy. The qa detail statistics are shown in Tab. A.4 and Tab. A.3. The examples of SFT QA are shown in Fig. A.5.

(2) RL data construction centers on creating *shared-question* counterfactual QA pairs: for each *real* and *edited* video pair, we design the same question and identical answer candidates, but the correct answer differs between the two videos. This forces the VLM to ground reasoning in actual visual content and detect subtle changes, rather than relying on prior plausibility. We construct the RL dataset using Gemini2.5-Pro, which generates counterfactual QA pairs from video captions by identifying visual differences. The prompting strategy follows the **RL Question Generation Prompt** in Fig. A.4. In total, we curate 20K counterfactual QA pairs as the RL training dataset. The examples of RL QA are shown in Fig. A.6.

Table A.5. Counterfactual video category statistics in DualityVidQA-Test

Tag	Count
causal reversal	158
counter physical	221
object/scene deformation	187
attribute change	33
All	599

(3) Test Set. We construct a high-quality test set, DualityVidQA-Test, to evaluate counterfactual understanding. Firstly, we sampled around 2000 pairs from our paired video pool. Then, we employ Gemini 2.5 Pro to generate candidate based on video content and dense captions. The prompt is **RL Question Generation Prompt** in Fig. A.4. Then we employ 3 human annotators and 3 expert reviewers to filter and refine the generated QA pairs, ensuring each question is valid, unambiguous, and answerable based on the video content.

The final test set consists of 600 real-counterfactual video pairs, each with a shared question and options but different answers. We then cluster the test set into 12 categories, then manually cluster them into 4 major categories: counter physical, object/scene deformation, causal reversal, and attribute change. The statistics of counterfactual video categories are shown in Tab. A.5. The examples of test QA are shown in Fig. A.7.

B. Derivation

Here we show the derivation of

$$S = |G| \sum_{i \in G} \left| \hat{A}_i \right| = 2\sqrt{(1-\overline{R})\overline{R}}. \tag{B.1}$$

We consider the case where the reward values R_i are binary, i.e.,

$$R_i \in \{0, 1\}. \tag{B.2}$$

Dense Caption Prompt Template

You are a professional video understanding and visual anomaly detection expert. Please generate a description for the given video.
Given **[num_frames]** uniformly sampled frames from the video (total duration approximately **[video_duration]** seconds), the time periods are **[time_list]**, please generate a detailed description in chronological order.
I will use a red box to figure out the anomaly region.

- anomaly_type: **[anomaly_type]**
- region_type: **[region_type]**
- region_name: **[region_name]**
- anomaly_start_time(s): **[anomaly_start_time]**
- anomaly_end_time(s): **[anomaly_end_time]**

Please pay special attention to the following points:

1. Describe in detail the important objects, actions, and relationships in each time period of the video, following chronological order, and merge the same content to ensure the dense caption is efficient and clear.
2. Carefully analyze and point out any visual anomalies, such as:
   - Unnatural changes in object appearance (distortion, warping, blurring, etc.)
   - Perspective distortion or geometric deformation in specific regions
   - Discontinuities or unnatural transitions in object edges
   - Abnormal changes in texture or color
   - Unnatural changes in lighting effects
   - Anomalous behavior in specific regions of the video (e.g., lens, objects)
3. For each detected anomaly, please specify in detail:
   - The exact time period when the anomaly appears
   - The specific region or object affected by the anomaly
   - You need to convert the anomaly parameters into natural language descriptions (do not output values like 'saturation factor is xx', and do not output specific region coordinates).
4. If there are other anomalies, such as blurring, missing content, unrecognizable scenes, etc., please clearly point them out in the relevant paragraphs.
   But don't need mention red box in the description.

The output format should be JSON, including the following content:

```
{
    "spatial_location": "[region_name]",
    "merged_timestamps": ["[0.0s - ...]", "[... - ...]"],
    "dense_captions": [
        "[0.0s - ...]: ...",
        "[... - ...]: ...",
        "...",
        "[... - video_end_time]: ..."
    ]
}
```

Figure A.1. Dense Caption Prompt Template

Let $|G|$ be the size of the group, and let

$$\overline{R} \;=\; \frac{1}{|G|}\sum_{i\in G} R_i \tag{B.3}$$

denote the accuracy of the group (i.e., the fraction of correct responses).

Standard Deviation of rewards.

$$\begin{aligned} std(\{R_i\}_{i=1}^{G}) =& \sqrt{\frac{\overline{R}\cdot|G|\cdot(1-\overline{R})^2+(1-\overline{R})\cdot|G|\cdot(0-\overline{R})^2}{|G|}} \\ =& \sqrt{\overline{R}\cdot(1-\overline{R})} \end{aligned} \tag{B.4}$$

The magnitude of the advantage is therefore:

$$|\hat{A}_i| = \begin{cases} \frac{1-\overline{R}}{\sqrt{\overline{R}\cdot(1-\overline{R})}}, & \text{if } r_i = 1, \\ \frac{\overline{R}}{\sqrt{\overline{R}\cdot(1-\overline{R})}}, & \text{if } r_i = 0. \end{cases} \tag{B.5}$$

Counterfactual Video Multiple Choice and Open-Ended Question Generation Prompt

Task Given a detailed description that summarizes the content of the generate-video, generate question-answer pairs to build LLM training data.
Reference Examples: Here is one question dimension and its explanation and example question-answer pairs for reference:
Question Type: [question_type]
Example 1:

```
## caption-1: [Video description]
## question-1: [Question text]
## answer-1: [Answer text]
```

Example 2:

```
## caption-2: [Video description]
## question-2: [Question text]
## answer-2: [Answer text]
```

Example 3:

```
## caption-3: [Video description]
## question-3: [Question text]
## answer-3: [Answer text]
```

You need to generate similar question-answer pairs like the examples.
Guidelines For Question Generation:

- Please formulate questions using only objectively observable information, without presupposing or emphasizing any abnormal, strange, or logically impossible phenomena in the questions themselves.
- The questions should be neutral and natural, while the answers may accurately describe the observed phenomena.
- Each multiple-choice question should have 4 options (A, B, C, D), with only one correct answer.
- The answer must be correct with respect to the video visual content.
- For abnormal object/event questions, include an option stating "The video is normal" as a distractor.
- Generate 1-4 question-answer pairs.
- Do not mention people's reactions to abnormal phenomena.
- For open-ended questions, provide detailed descriptions including speed and direction of actions/camera movements.

Input: Dense Caption: [dense_caption]
Output Format: Your output should be formatted as a JSON file:
For Multiple Choice Questions:

```
[{
    "Question": "<question-1>",
    "Options": ["<option-0>", "<option-1>",
                "<option-2>", "<option-3>"],
    "Answer": "index of correct option"
}]
```

For Open-Ended Questions:

```
[{
    "Question": "<question-1>",
    "Answer": "<a detailed answer-1>"
}]
```

Figure A.2. Counterfactual Video QA Generation Prompt Template

Sum of ℓ_1 norm. The sum of ℓ_1 norm of $\hat{A}_i$ over the group is:

$$\begin{aligned} S =& \frac{1}{|G|}\sum_{i\in G}|\hat{A}_i| \\ =& \frac{1}{|G|}\left[|G|\cdot\overline{R}\cdot\frac{1-\overline{R}}{\sqrt{\overline{R}\cdot(1-\overline{R})}} \right. \\ & \left. +|G|\cdot(1-\overline{R})\cdot\frac{\overline{R}}{\sqrt{\overline{R}\cdot(1-\overline{R})}}\right] \\ =& 2\sqrt{\overline{R}\cdot(1-\overline{R})} \end{aligned} \tag{B.6}$$

Real Video Multiple Choice and Open-Ended Question Generation Prompt

Task: Given a detailed description that summarizes the content of video, generate question-answer pairs to build LLM training data.
Reference Examples:
Question Type: [question_type]
For Multiple Choice:

```
## caption-1: [Video description]
## question-1: [Question text]
## options-1: [A. Option1, B. Option2, C. Option3, D. Option4]
## answer-1: [Correct answer]
```

For Open-Ended:

```
## caption-1: [Video description]
## question-1: [Question text]
## answer-1: [Detailed answer]
```

You need to generate similar question-answer pairs like the examples.
Guidelines For Question Generation:
For Multiple Choice Questions:
- Generate appropriate multiple-choice question-answer pairs based on the description
- Each question should have 4 options (A, B, C, D)
- Only one option should be correct
- Other options should be plausible distractors
- Distractor options must be reasonable, relevant to the question, and not obviously wrong

For Open-Ended Questions:
- Generate appropriate question-answer pairs based on the description
- Answers should be detailed and comprehensive

General Guidelines:
- Generate 1-4 question-answer pairs
- Questions should focus on observable content in the video
- Maintain natural and objective question formulation

Output Format:
For Multiple Choice Questions:

```
[{
    "Question": "<question-1>",
    "Options": ["<option-0>", "<option-1>",
               "<option-2>", "<option-3>"],
    "Answer": "index of correct option"
}]
```

For Open-Ended Questions:

```
[{
    "Question": "<question-1>",
    "Answer": "<a detailed answer-1>"
}]
```

Figure A.3. Real Video QA Generation Prompt Template

RL Question Generation Prompt

Task: Given two captions — **TRUE CAPTION** (original video description) and **MOCK CAPTION** (edited video description after applying an **edit instruction**) — design a question that can be answered differently for the TRUE and MOCK videos. The goal is to produce high-quality, dimension-specific question-answer pairs for training multimodal models.
Reference Example:
TRUE CAPTION: The man places a cake on the table and lights the candles. MOCK CAPTION: The man places a cake on the table without lighting any candles. Edit Instruction: Remove the candle lighting action.
Question: What does the man do with the cake after placing it on the table? Answer for TRUE: He lights the candles on the cake. Answer for MOCK: He leaves the cake as it is without lighting candles. Wrong Answers: [“He cuts the cake into slices”, “He puts the cake back into the oven”]
Guidelines for Question Generation:
Core Requirements:
- Base questions strictly on differences between the TRUE and MOCK videos.
- Do not refer to or mention captions directly in the question.
- No timestamps or meta-information in the question.
- Use the provided **edit instruction** as a design hint.
- Questions must belong to one of the predefined **task dimensions**.
- If no suitable question for the chosen dimension, output an empty question string.
- Wrong answers must be incorrect for both videos, but still plausible.
- Generate answers for each video independently without inferring from the other.

Available Dimensions: Refer to the predefined TASK_EXAMPLES set for dimensions and descriptions.
Output Format: The result must be **valid JSON** with the following structure:

```
{
    "dimension": "<task dimension>",
    "question": "<generated question>",
    "answers_for_true_caption": ["<answer based on TRUE CAPTION>"],
    "answers_for_mock_caption": ["<answer based on MOCK CAPTION>"],
    "wrong_answers": ["<wrong answer 1>", "<wrong answer 2>", ...]
}
```

Figure A.4. RL Question Generation Prompt

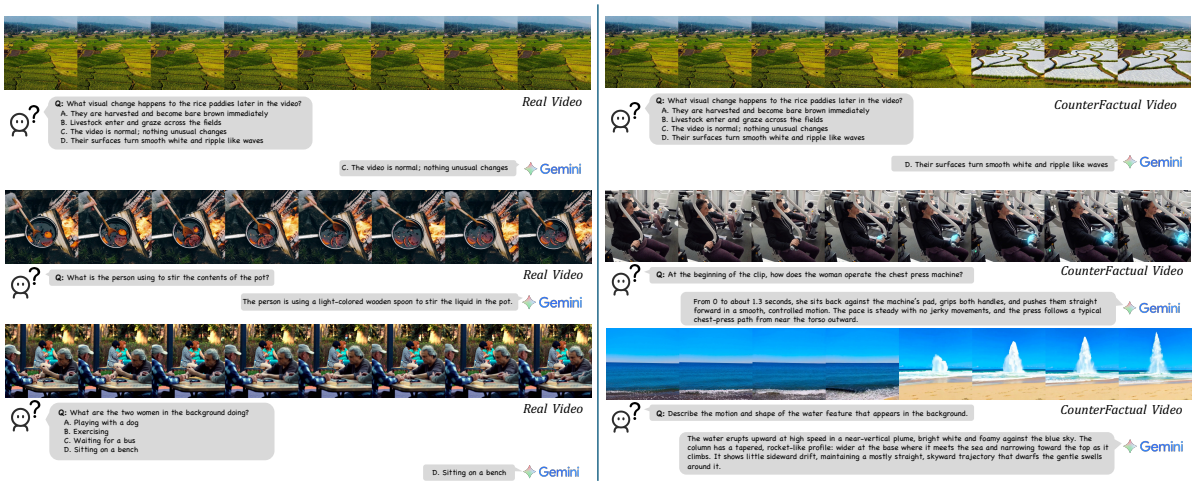

Figure A.5. Examples of DualityVidQA-SFT. We show the real video and counterfactual video pair and the question and answer pair generated based on the counterfactual video.

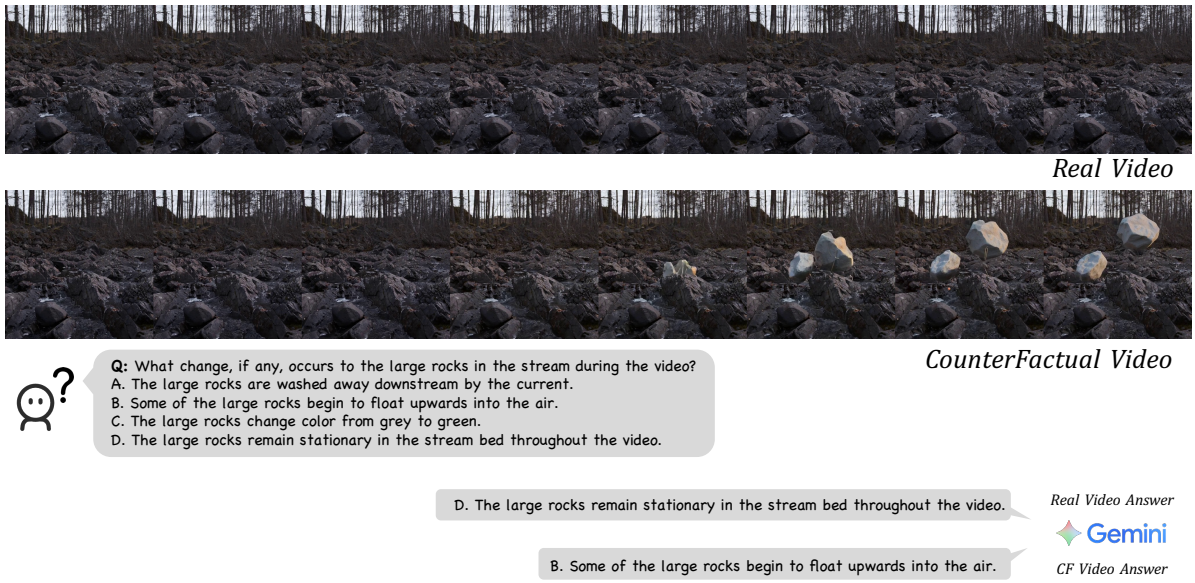

Figure A.6. Examples of DualityVidQA-RL. We show the real video and counterfactual video pair and the generated question and answer.

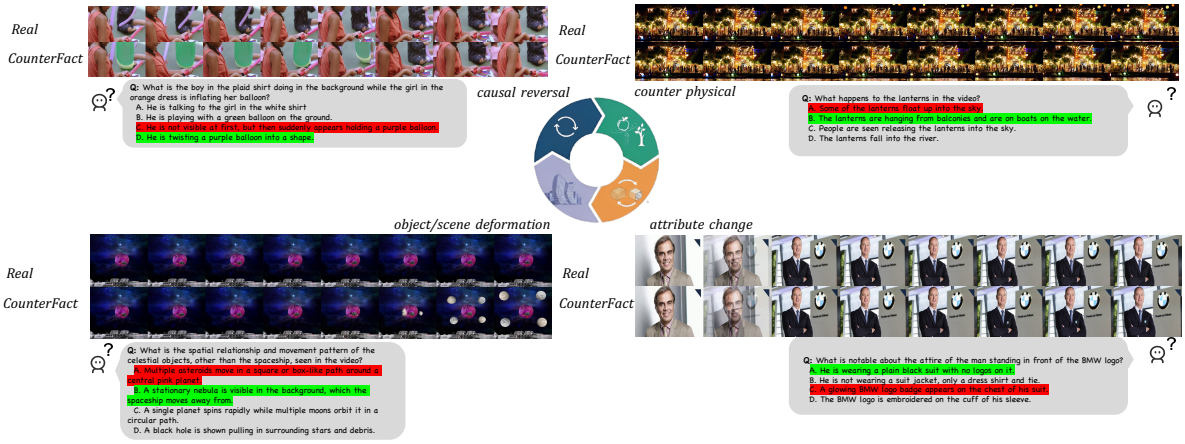

Figure A.7. Examples of DualityVidQA-Test. We show the real video and counterfactual video pair and the generated question. Answers for the counterfactual video are shown in red, and answers for the real video are shown in green.

References

[1] Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv e-prints*, pages arXiv–2506, 2025. 1

[2] Corran. Pexel Videos, 2022. Accessed: 2025-09-19. 1

[3] Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. *arXiv preprint arXiv:2503.07598*, 2025. 1

[4] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024. 1

[5] Kepan Nan, Rui Xie, Penghao Zhou, Tiehan Fan, Zhenheng Yang, Zhijie Chen, Xiang Li, Jian Yang, and Ying Tai. Openvid-1m: A large-scale high-quality dataset for text-to-video generation. *arXiv preprint arXiv:2407.02371*, 2024. 1

[6] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 1