GlitchBench: Can large multimodal models detect video game glitches?

Mohammad Reza Taesiri[1], Tianjun Feng[1], Cor-Paul Bezemer[1], Anh Nguyen[2]
[1]University of Alberta, {mtaesiri, robbie020428, bezemer}@ualberta.ca
[2]Auburn University, anh.ng8@gmail.com

Abstract

Large multimodal models (LMMs) have evolved from large language models (LLMs) to integrate multiple input modalities, such as visual inputs. This integration augments the capacity of LLMs for tasks requiring visual comprehension and reasoning. However, the extent and limitations of their enhanced abilities are not fully understood, especially when it comes to real-world tasks. To address this gap, we introduce ***Glitch****Bench, a novel benchmark derived from video-game quality assurance tasks, to test and evaluate the reasoning capabilities of LMMs. Our benchmark is curated from a variety of unusual and glitched scenarios from video games and aims to challenge both the visual and linguistic reasoning powers of LMMs in detecting and interpreting out-of-the-ordinary events. Our evaluation shows that* ***Glitch****Bench presents a new, interesting challenge to state-of-the-art LMMs. Code and data are available at: https://glitchbench.github.io/*

1. Introduction

The video game industry boasts an estimated annual revenue of USD 217 billion [57] with a total of 3.2 billion gamers worldwide in 2022 [1]. Automatically detecting in-game glitches is, therefore, a highly demanding task, but that remains a long-standing challenge [12, 39, 51, 55, 56, 65, 66, 72, 83]. A *glitch* is an unexpected frame that occurs within a game due to either an unforeseen software bug, player actions, or unanticipated interactions between game elements and does *not* result in a program crash. From a computer vision perspective, glitch detection involves recognizing an extremely wide spectrum of long-tail video frames, from rendering (*e.g.*, 3D objects with missing textures; Fig. 2f), unrealistic physics (*e.g.*, two people sitting in an invisible car; Fig. 2b), to semantic errors (*e.g.*, indoor rain; Fig. 1). Unlike software bugs that can be caught by examining the code alone, glitches are more non-trivial to detect because they are mostly the result of a one-time event that unexpectedly happens in-game.

A holy grail of game quality assurance is to build a general glitch detector that works for any game of any genre and mechanics. We set the first step toward this goal by building **Glitch**Bench, an evaluation benchmark of 593 glitches, leveraging the public's crowd knowledge from the game community's reports on reddit.com/r/GamePhysics. The glitches span across 205 games of various genres. Each glitch has a video clip, a representative frame, a one-line description, and a reference to a corresponding Reddit thread where gamers discussed the error.

GPT-4V fails to focus on the rain inside the room

Question: What is wrong with this image?
Ground truth: It is raining inside the building
GPT-4V: There's nothing inherently "wrong" with this image; however, if you're referring to inconsistencies or peculiarities that might not align with the established Batman canon, the red Batman suit is unusual since Batman is traditionally known in black or very dark colors. The use of red could be an artistic choice for the game, an alternate suit available within the game's universe, or indicative of a different character who takes on the Batman mantle.

Figure 1. The image depicts a screenshot in which it rains inside a room. While the rain should be what is wrong with the image, GPT-4V fails to reason correctly and instead focuses on the color of Batman's costume. Note that the ground truth is never presented as part of the prompt in our study.

Large image-text, multimodal models (LMMs), such as GPT-4V [2], are improving at an unprecedentedly fast pace. They excel in many existing tasks, including object detection [44, 75], multi-step reasoning [4, 5, 10, 35], and detailed image captioning [2, 38, 42, 52, 76]. Testing LMMs on **Glitch**Bench may yield important findings not only to the game industry but also to the Artificial Intelligence (AI) community because glitch detection requires a combination of knowledge and understanding of image aesthetics, computer graphics, physics and commonsense reasoning (skills that are often tested individually in a benchmark [8]).

In this paper, we evaluate how well LMMs perform in detecting glitches from a single frame. Our main findings and contributions include:

1. We introduce **Glitch**Bench, which contains 330 glitch-free and 593 glitch screens taken from 205 games for evaluating LMMs (Sec. 3).
2. We evaluate 11 state-of-the-art LMMs, including GPT-4V [2] and LLaVA [42] on our benchmark and in comparison with the performance on 6 other common benchmarks (Sec. 4).
3. LMMs are better at detecting glitches that violate simple physical laws (*e.g.*, a car flying in the air) than other more subtle glitches (*e.g.*, human limbs in an implausible pose; Fig. 6).
4. The state-of-the-art model on **Glitch**Bench is GPT-4V with 43.4% accuracy. In the extensive captioning setup, we estimated the upper limits of models, and GPT-4V can achieve an accuracy of 64.9%, which is almost twice that of LLaVA, the second-best model (30.5%).
5. In sum, there exists a headroom of 30–35% on **Glitch**Bench for future LMM models to improve, presenting an interesting challenge to the AI community.

2. Related Work

2.1. Multimodal, image-text datasets

Recently, there has been rapid development of large multimodal models that can process multiple modalities, including visual and textual inputs. Existing datasets that come with human-generated image captions, such as COCO Caption [13], Nocaps [3], CapFilt: [36] and Flickr30k [53], can serve as a simple way to evaluate language models. By providing the image, we can ask a model to describe it and then compare the generated caption with the ground truth [42, 43, 77]. Image captioning is a narrow domain and can be extended into visual question answering (VQA) by asking questions related to an image. Datasets like GQA [27], OK-VQA [49], VQAv2 [22], and Vizwiz [23] contain image-question pairs to probe the visual reasoning and understanding of LMMs.

Building upon simple VQAs, several benchmarks aim to increase the complexity of tasks over different dimensions. TextVQA [63], OCR-VQA [50] and TextCap[62] propose questions about the text shown in the image. ScienceQA [47] and MathVista [48] focus on scientific topics and charts, while VCR [80] and Sherlock [80] focus on commonsense reasoning. Moreover, AI2D [26] is directed at questions concerning scientific diagrams, and IconQA [46] targets the comprehension of abstract diagrams. Each of these benchmarks is designed to push the boundaries of VQA systems by introducing specialized content that requires advanced reasoning and understanding.

There are also comprehensive evaluation frameworks that assess multimodal language models across a wider spectrum of capabilities. These evaluations extend beyond visual and textual reasoning to encompass a variety of skills such as generation, question answering, adherence to instructions, and the application of commonsense logic. Notable among these are SEED-Bench [33] , MME [19], MM-Bench [45], MM-Vet [79], VisIT-Bench [8], which collectively serve to provide a robust measure of a model's proficiency in handling tasks that integrate multiple modalities.

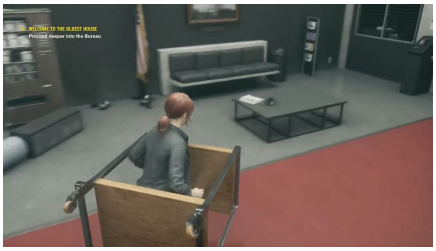
Unlike traditional datasets that contain queries about elements present in the image, our approach is novel in directing models to discern the atypical aspects, *i.e.*, glitches, with no linguistic hints provided. We show an image to the model and ask it to report unusual aspects of it. Such questions require a more integrated approach to visual and linguistic processing within an LMM to formulate a response.

2.2. Vision-language Stress Testing

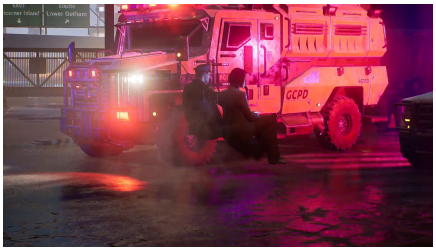
Out-of-distribution (OOD) datasets have become a cornerstone for evaluating the capabilities and progress of machine learning models. In standard image classification, in particular the ImageNet [59] dataset, the introduction of datasets [24, 25, 25, 64] has underscored the importance of robustness and generalization in model evaluation. As we move from simple image classification tasks to more complex multimodal tasks, there is an increasing need for similar OOD datasets that can comprehensively test the generalization abilities of LMMs.

There are several studies that stress test various aspects of vision from different angles, such as compositional and spatial reasoning [20, 28, 29, 67], objects placed out of context and implausible scenes [9, 14, 84], and the exploitation of language and vision priors [18, 40].

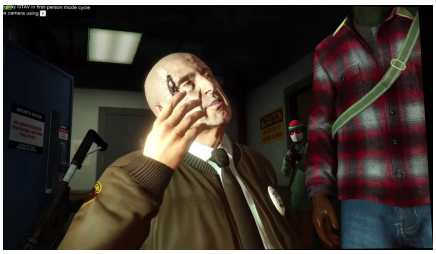
The closest benchmark to ours is Whoops [9], which is designed to challenge commonsense knowledge and reasoning in LMMs. However, our dataset differs in several ways: (1) The tasks in **Glitch**Bench come from real-world tasks, specifically video game quality assurance, and are not artificially created to test models. (2) Whoops requires cultural and background knowledge to answer; for example, *A panda bear is catching salmon fish* is unusual



(a) A person stuck in a piece of furniture



(b) Two people driving an invisible car



(c) A rifle floating in the air

(d) A person is floating in the air

(e) The gun in the hand is missing

(f) The table cover has a placeholder texture

Figure 2. Sample images from the **Glitch**Bench showing glitches in various games with distinct styles. Samples (a)–(e) are captured from online videos, while sample (f) is generated inside the Unity game engine.

since pandas subsist almost entirely on bamboo. In contrast, our dataset contains samples that contradict basic commonsense and the physics of the world. (3) Finally, images in Whoops are synthesized using image-to-text models; they are clear without artifacts, centered in the image, and do not stress the visual side of the image, focusing on the context. In contrast, for **Glitch**Bench, models need to fully scan the image to identify its unusual aspects (Fig. 2), and there are many distracting elements present in the image, challenging them to focus on the correct part of the image.

2.3. Empirical Analysis of Recent LMMs

With the release of recent proprietary LLMs, such as GPT-4V and Bard [21], some studies attempt to evaluate and report the performance of these models on various benchmarks and tasks [16, 54, 73]. The main goal of these studies is to provide a comprehensive evaluation of the models across various well-established tasks and some narrow domains [71, 74]. The main difference between our work and these studies is that we propose a general, stress-testing benchmark to measure the generalization power of various LLMs, both proprietary and open source, on a specific, glitch-detection task in the game industry.

3. GlitchBench

In this section, we describe the creation process of **Glitch**Bench, a benchmark aimed at stress-testing visual perception and commonsense reasoning in LMMs, motivated by real-world game quality assurance tasks.

During development, video games go through many stages of testing to reach certain quality standards before release. However, even after release, they can still exhibit unusual in-game events, or glitches. Glitches, often viewed as annoying bugs, can also possess a humorous and entertaining aspect. Players frequently report glitches across various social media platforms, particularly on Reddit and YouTube. A critical aspect of understanding glitches is the requirement of commonsense knowledge about the basic laws of physics of the game's universe, making them a suitable and practical candidate for testing machine learning models. Fig. 2 shows six samples from **Glitch**Bench.

3.1. Constructing the Dataset

GlitchBench contains two parts: (1) 513 samples shared by players of video games, *i.e.*, frames collected from online sources, and (2) 75 synthetic samples.

Samples shared by players of video games: To construct our dataset, we sampled 1,000 videos from the GamePhysics [66] dataset. This dataset consists of videos from a subreddit with the same name, containing gameplay video clips with unusual events and glitches.

Next, we conducted a manual review process to filter videos based on two criteria: (1) the presence of a glitch in the video, and (2) the potential for humans to detect the glitch from a single frame. The second criterion is key because certain glitches, such as those involving rapid shaking or changes in size over time, cannot be detected from a still image alone.

After applying these filters, we extracted one frame from

each remaining video, resulting in a collection of 650 samples. Our final round of manual reviews revealed two potential issues: (1) some glitches are not detectable from the extracted image and require more context to understand, and (2) some images contain the faces of gamers who streamed the content on an online platform (which could cause the LMM to identify these faces as what is wrong with the images). After removing videos that contain one of these issues, our final glitch set contains 513 images.

Generating synthetic samples with Unity: To enhance our dataset, we supplemented samples from the GamePhysics dataset with 75 synthetic examples created inside the Unity game engine. These samples were specifically designed to mimic a subset of common development-stage bugs [39, 55, 65] that are not readily available in online social media platforms and, hence, to diminish the survivor bias effect. These flaws are often fixed before the public release of a game through the quality assurance process of a game development company and are therefore not often posted on social media.

Our synthetic sample generation process involves the injection of three categories of glitches into each scene: (1) placeholder textures, (2) object mesh distortions, and (3) low-resolution textures.

Glitch-free images: Our focus is on glitch frames, as they are more challenging to capture and collect. However, to establish a baseline for comparison, we also included a set of glitch-free images. To accomplish this, we randomly selected gameplay walkthroughs from various games on YouTube. From these walkthroughs, we extracted a random subset of frames, resulting in the compilation of a dataset consisting of 330 frames sourced from a diverse array of games. The groundtruth captions for these glitch-free images is *"There is nothing wrong with this image"*.

3.2. Labeling the Dataset

For all images, we provide a short description of the glitch present in the image. Our goal is to label the images briefly, highlighting only the unusual elements in simple language. For instance, if an image depicts a character with a contorted physique, the label would simply state, *"character has an unnatural body position"*.

It is important to highlight that some images can be described in many different ways. Diverse phrases such as *"falling from the sky"*, *"suspended in mid-air"*, or *"jumping in the air"* might all refer to a single event. Instead of handling such cases in the labeling process, in the evaluation process, we incorporate a language model to diminish the effect of this (see Sec. 4.1).

3.3. Categorizing the Glitch Types in the Images

In this section, we provide a high-level categorization of glitches in our dataset. While there have been some attempts to provide a taxonomy of video game bugs [32, 69], these taxonomies do not provide descriptions that are adequate to automate bug categorization.

We propose a novel human-AI team-based method to build a categorization based on the descriptions of the images. This process is a collaborative effort between GPT-4 and humans, where GPT-4 suggests initial categories, and then humans refine these suggestions by providing feedback or asking the model to re-evaluate its output, harnessing the reflective ability of GPT-4 [61]. Finally, we manually bridge the resulting categories to those proposed by Lewis et al. [32] based on the semantics and instances of the glitches in our dataset.

Process: We prompt GPT-4 with all the glitch descriptions in our dataset and ask it to generate a categorization based on the descriptions and semantics of the glitches. In each subsequent iteration, we provide feedback in one of two ways: (1) we ask GPT-4 to review its previous answer through reflection, or (2) we explicitly instruct the model to merge two categories that are semantically similar. We stop when the model no longer changes its answer through reflection or when we can no longer merge categories.

In the last step, to assign each image to a category, we prompt GPT-4 with the description of the glitch and the final categories and ask it to assign each image to one of them. The final categories, the number of instances, examples for each category, and the parent category proposed by Lewis et al. [32] are outlined in Table 1.

4. Experiments

4.1. Experimental Setup

Formulating Questions: We designed `Glitch`Bench as a free-text response benchmark, in contrast with traditional LMM benchmarks that utilize Yes/No or multiple-choice formats [19, 33]. We ask models to describe the unusual aspects of an image by answering three questions:

(Q1) *What is unusual about this image?*
(Q2) *What is wrong with this image?*
(Q3) *Describe the image in detail*

Note that we do not explicitly use the word *glitch* in the question, and we use simple language similar to what a layperson would use. During the inference, we allow models to come up with their own reasoning, and after the model generates the full response, we record it for further evaluation and comparison with the ground truth.

The rationale for free-text answers is that including an 'unusual' event description among choices hints to the LMM, letting it answer while disregarding visual aspects.

Table 1. Categorization of video game glitches in **GlitchBench**. Numbers highlighted in ▫ show the number of images in each category. Categories highlighted in ▪ show the corresponding categories proposed by Lewis et al. [32].

Physics, Collision, and Spawn (Non-Temporal → Invalid position)	Images: 422
1. Objects and characters floating or stuck in the air (Fig. 2d). 2. Characters or objects clipping through solid objects like walls, floors, or ground. 3. Vehicles or characters falling under the game map.	
Animation and Pose (Non-Temporal → Invalid graphical representation)	Images: 75
1. Unusual or impossible body poses and positions (Fig. 6). 2. Characters in a T-pose or with distorted body parts. 3. Incorrect animations for certain actions.	
Rendering and Texture (Non-Temporal → Invalid graphical representation)	Images: 67
1. Mesh stretches or objects with distorted shapes. 2. Missing textures or objects displaying a "default" placeholder texture (Fig. 2f). 3. Objects with low-resolution.	
Camera, User Interface, and Lighting (Non-Temporal → Invalid value change)	Images: 26
1. Camera issues such as clipping inside objects or improper character views. 2. In-game menus displaying incorrect elements. 3. Shadows or lighting effects that do not match the environment.	

We included question **Q3** to assess whether the models can accurately report any glitches or unusual elements within the image in extensive captioning. Essentially, this question serves as a visual perception test, evaluating whether the models can identify and describe unusual aspects of the image in a more relaxed condition. For example, in the sample shown in Fig. 1, we test the model to see if it can identify the presence of rain in the room. In this case, it indicates that it is raining outside.

Evaluation: Following recent successes [8, 41, 78, 82] we employ a language model as a judge to evaluate the model's responses. We use Llama-2-70B-Chat [68] to compare the model-generated text with the ground truth and determine whether the text conveys the same meaning or mentions the event highlighted by the ground truth (see Fig. 3).

We report the accuracy of each model on each tested question and present the average performance for **Q1** and **Q2** as the final benchmark result. **Q3** serves as the visual perception test, and we report the performance of the models on it separately.

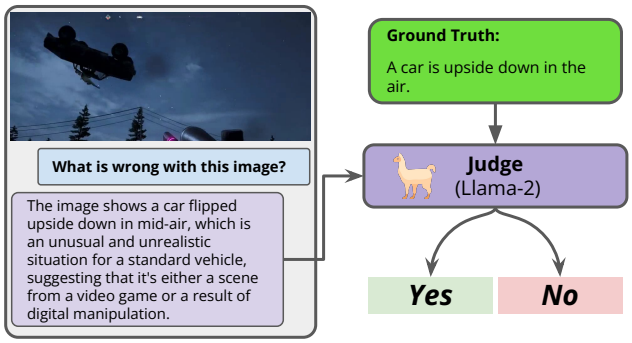


Figure 3. To evaluate a model's response, we ask a judge (the Llama-2-70b-Chat model) to compare it semantically with the ground truth.

To assess Llama-2's judgment and determine if it can effectively serve as an evaluator, we manually reviewed a subset of responses for each model. For each model, we manually labeled 20 samples, with a total of 220 samples.

Models: In total, we evaluated 11 LMMs, including GPT-4V [2], and 10 open source models: LLaVA-1.5 (7B and 13B) [42], SPHINX (7B and 13B) [38], InstructBLIP (7B and 13B) [17], Qwen-VL-Chat (10B) [6], MiniGPT-v2 (7B) [11], OtterHD [34], and Fuyo (8B) [7]. We used the default temperature and top-p configurations provided with the model and API. We increased `max_token` to get full responses from models. (See Sec. A1 for details).

4.2. Quantitative Results

Table 2 shows the performance of all the tested models for the three questions. The Average performance on **Q1** and **Q2** is the main result of our benchmark. GPT-4V is the best-performing model, achieving 57.2% (**Q1**) and 29.5% (**Q2**) and an average of 43.4%. Next, LLaVA-1.5-13B achieves an average of 35.5% and is the best performing open-source model. These findings show **GlitchBench** is challenging for even state-of-the-art commercial & open-source models.

The performance of GPT-4V on glitch-free images is much higher than on glitch images, with an average accuracy of 91.6%, which suggests that glitch-free images are much easier to handle.

Models exhibit different performance depending on the questions being asked, but all except for the SPHINX family show better performance when prompted with **Q1**. Nevertheless, the gap in performance varies, with GPT-4V showing the largest gap of 27.7pp (57.2% vs. 29.5%). These results highlight that different prompts steer the behavior

Table 2. Accuracy of various LMMs on **Glitch**Bench. Numbers highlighted in represent the average results of Q1 and Q2, which are the main results of the benchmark. Numbers related to Q3 serve as a visual perception test to measure the ability of models to report glitches in a relaxed manner. Numbers highlighted in show the maximum agreement achievable with ground truth as perceived by Llama-2's judgment (%). Numbers highlighted in represent the results obtained from GPT-4V on glitch-free images.

Question	GPT-4V [2]		LLaVA-1.5 [42]		SPHINX [38]		InstructBLIP [17]		OtterHD [34]	Qwen -VL [6]	MiniGPT -v2 [11]	Fuyu [7]
	n/a	n/a	7B	13B	7B	13B	7B	13B	8B	10B	7B	8B
Q1. What is unusual about this image?	88.2	57.2	35.2	36.3	19.2	25.3	25.3	21.9	24.8	21.2	19.1	8.6
Q2. What is wrong with this image?	95.5	29.5	23.9	34.7	30.9	30.5	13.8	8.9	23.3	9.3	17.9	8.4
Average	91.6	43.4	29.6	35.5	25.0	27.9	19.6	15.4	24.0	15.2	18.5	8.5
Q3. Describe the image in detail.	-	64.9	28.0	30.5	17.5	21.9	16.0	11.8	21.6	14.0	16.0	7.6
Maximum Agreement	95.5	64.9	35.2	36.3	30.9	30.5	25.3	21.9	24.8	21.2	19.1	8.6

of LMMs differently and suggest that multi-step reasoning [31, 70] could also help LMMs.

Our results also highlight that higher resolutions improve the performance. In particular, SPHINX-13B, which operates at a higher resolution than SPHINX-7B (448×448 vs. 224×224), on average performs +2.9 pp (27.9% vs. 25.0%) better than the base model. Similarly, OtterHD, which employs Fuyu as the base model with enhanced flexibility and support for higher image resolutions, outperforms Fuyu on average by +15.5 (24.0% vs. 8.5%).

Asking LMMs to extensively caption the image using **Q3** only triggers GPT-4V to produce a very verbose response. In many cases, GPT-4V describes many details in the image and can touch upon the unusual aspects of the image. In this setup, GPT-4V can achieve 64.9%, which is an increase of +7.7 over **Q1** and +21.5 pp better than the benchmark results. This gap suggests that GPT-4V can *see* many details in the image, but it cannot easily focus on the unusual aspects in the frame, indicating a gap in its reasoning capabilities across different modalities and prompts.

Human evaluation: Table 3 shows the results of comparing between Llama-2 judgments and human evaluations, with the level of agreement for each model measured by Cohen's Kappa [15]. Cohen's Kappa demonstrates varying levels of concordance for each model. GPT-4V (0.80), InstructBLIP-7B (0.83), and Qwen-VL (1.00) exhibit substantial to perfect agreement. In contrast, OtterHD (0.50) had fair agreement, and Fuyu (-0.09) shows less than chance agreement, suggesting significant discrepancies. Overall, on all models except for Fuyu, we found above moderate agreement between Llama-2 and human judgment, while on six models, this agreement is substantial.

Accuracy breakdown by category of glitches: Fig. 4 shows the breakdown of the performance of all tested models across the four studied glitch categories. GPT-4V is the best-performing model across all categories, with the exception of the *Rendering and Texture* category, where LLaVA-1.5-13B slightly outperforms it by +2.3 (41.0% vs. 43.3%). Overall, the *Animation and Pose* category consistently proves to be the most challenging. This category contains images of characters in unusual poses, distorted body joints, or twisted bodies (see an example in Fig. 6).

Table 3. Evaluating a subset of responses for comparing Llama-2 with human judgments: Llama-2 and humans exhibit moderate to substantial agreement on all models except for Fuyu.

Model	Llama-2	Human	κ
GPT-4V	60.0	50.0	0.80
LLaVA-1.5-13B	25.0	20.0	0.57
LLaVA-1.5-7B	35.0	15.0	0.49
Long-SPHINX	25.0	35.0	0.53
SPHINX	30.0	25.0	0.63
InstructBLIP-13B	20.0	10.0	0.62
InstructBLIP-7B	20.0	15.0	0.83
MiniGPT-v2	10.0	5.0	0.64
Qwen-VL	20.0	20.0	1.00
OtterHD	25.0	10.0	0.50
Fuyu	20.0	5.0	-0.09
$\mu \pm \sigma$	26.4 ± 12.8,	19.1 ± 13.5	0.64

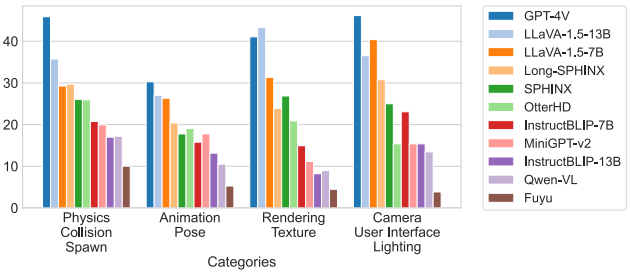


Figure 4. The performance of all tested models on different categories of images in **Glitch**Bench.

4.3. Qualitative Observations and Analysis

Failing to reason about unusual aspects of the image: We observed that in several cases, particularly in open-source models, the model reports phrases such as "*the problem with this image is that it is computer-generated*" or "*this is not an actual scene but a scene from a video game*", along with similar phrases conveying the same meaning. These phrases suggest that, despite the model's ability to *see* the content of the image, the language component of the model completely fails to reason about the content of the image.

Another observation is that InstructBLIP-13B often responds with "*nothing*" or similar phrases and completely fails to reason about the image. This is the reason why the smaller InstructBLIP-7B can achieve higher accuracy on **Glitch**Bench. (See Sec. A3.1 for samples.)

GPT-4V struggles with faces: GPT-4V is the best-performing model, yet it struggles with characters' faces, as shown in Fig. 5. We found several issues when processing glitches related to faces, and in the majority of cases, GPT-4V fails to detect the glitch and sometimes hallucinates about characters wearing costumes (Fig. A2), where there are basically no discernible facial features. On the other hand, smaller open-source models can sometimes detect glitches where GPT-4V fails, but they cannot describe the glitch clearly. We hypothesize that this might be due to the privacy features of GPT-4V, preventing it from seeing the face clearly (see Sec. A3.2 for more samples).

Poor performance in the *Animation and Pose* category: Fig. 4 shows that *Animation and Pose* is the hardest category of glitches. During manual analysis, we found that LMMs struggle to detect unnatural body and limb configurations and incorrect animations being displayed. For instance, Fig. 6 shows an example of an unnatural arm position for a basketball player that GPT-4V cannot detect. This category can be further divided into three subcategories: (1) Heavily distorted body, *e.g.*, when a character's body is overstretched or expanded; (2) Nuances in body configuration, such as contorted or twisted limbs and hands (Fig. 6); and (3) Characters playing animations with invisible props (*e.g.*, a missing gun in the hand, as shown in Fig. 2e).

Prevalent hallucination in open-source models: *Hallucination* typically refers to situations in which the model's generated text contains information not present in the image [16, 58, 81, 85]. We noticed that open-source models often hallucinate extra objects or content in the image, e.g., we found that Fuyu's responses almost always contain hallucinations (see Sec. A3.4). The hallucination can contain additional characters or entirely new objects. These extra elements sometimes mislead Llama-2 into accepting an incorrect response as correct (see Sec. A3.5).

GPT-4V fails to recognize the absence of facial skin

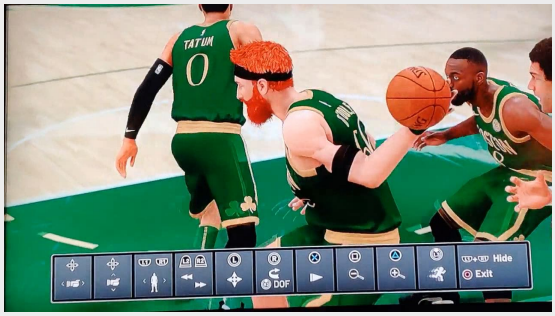
Question: What is wrong with this image?
Ground truth: The character's face mesh is not rendered correctly.
GPT-4V: The image appears to be from a video game or some form of CGI (computer-generated imagery) with very high-quality rendering. The characters depicted are very lifelike, suggesting a significant level of detail and investment in graphics. However, there's nothing inherently "wrong" with the image in terms of content; it seems to portray a close-up moment between two characters.
Llama-2 (Judge): No, the generated description does not accurately convey the key information and themes of the ground truth. While it acknowledges the high quality of the graphics and the lifelike nature of the characters, it does not address the issue of the character's face mesh not being rendered correctly, which is the main focus of the ground truth.

Figure 5. One of the several cases in which GPT-4V fails to detect a problem with facial features.

Detecting some glitches requires paying attention to small details: Different glitches in our dataset require varying levels of visual attention. For example, when a car is flying in the air (Fig. A11), it usually occupies a large portion of the pixels on the screen, and models can easily pick up on such issues. This type of glitch is easier to catch, and GPT-4V, in particular, is very accurate at detecting it.

Some types of glitches require close attention to detail, such as clipping issues with clothing, where parts of the clothes intersect with the face or body of a character. While GPT-4V is generally the best model for detecting clipping, it is still not perfect. In some cases, GPT-4V misses the clipping, and in other cases, it hallucinates about clipping (see Sec. A3.6 for samples).

GPT-4V fails to detect unusual body configuration



Question: What is unusual about this image?
Ground truth: The left hand of the basketball player is in an unusual position and is stretched backward.
GPT-4V: The image appears to be a screenshot from a basketball video game, possibly from the NBA 2K series. The unusual aspects of this image include: One of the players has a bright orange beard that stands out from the rest. Another player seems to be kissing or blowing at the basketball, which is not a typical basketball move or gesture. The characters and graphics look computer-generated, which indicates it's from a video game rather than a real-life game. The interface at the bottom of the image shows various controls and buttons, which further suggests that it's from a video game.
Llama-2 (Judge): No, the generated description does not accurately convey the key information and themes of the ground truth. The ground truth focuses on the unusual position of the basketball player's left hand, while the generated description highlights three unusual aspects of the image, none of which relate to the player's hand position.

Figure 6. The image shows a basketball player with an unnatural, impossible elbow pose. GPT-4V fails to focus on small details such as body configuration and is unable to report this issue.

5. Discussion and Limitation

Comparing GlitchBench with other benchmarks: Tab. 4 shows the performance of various models across different benchmarks, which shows that GPT-4V has different performance against open-source models compared to GlitchBench. E.g., LLaVA-1.5 and QWEN-VL score +5.8 (80.0% vs 74.2%) and +5.3 pp (79.5% vs 74.2%) higher than GPT-4V on VQAv2. On GlitchBench they lag behind by -9.9 (33.4% vs. 43.5%) and -28 pp (15.4% vs. 43.4%). The largest gap is seen in Fuyu's performance against GPT-4V: while Fuyu exceeds on both OKVQA and AI2D, it significantly lags behind on GlitchBench with only 8.5% compared to GPT-4V's 43.4%.

Table 4. Comparing GlitchBench with other visual benchmarks — the bold numbers show the best model per benchmark (%)

Model/Task	Glitch (Ours)	VQAv2 [22]	OKVQA [60]	AI2D [30]	SEED [33]	POPE [37]	MMB [45]
GPT-4V	**43.4**	74.2	60.6	64.5	-	-	-
LLaVA	33.5	**80.0**	-	-	70.7	-	**67.7**
SPHINX	27.9	-	-	-	71.6	**90.8**	67.1
InstructBLIP	19.6	62.1	-	-	-	78.9	36.0
MiniGPT	18.5	-	57.0	-	-	-	-
QWEN-VL	15.4	79.5	58.6	62.3	58.2	-	60.6
OtterHD	15.2	-	-	-	-	86.1	58.5
Fuyu	8.5	77.4	**63.1**	**73.7**	-	-	-

In sum, across multiple existing LMM benchmarks, open-source models can perform on par with or even surpass GPT-4V. However, their performance on GlitchBench, which is derived from a real-world task in game quality assurance, falls significantly short of GPT-4V. In other words, the performance of models in real-world settings does not correlate well with existing benchmarks. This discrepancy partly comes from the design choices typical of LMM benchmarks, as they often opt for Yes/No or multiple-choice formats [19, 33, 45]. These formats allow models to find shortcuts for scoring high without necessarily generalizing well to other tasks.

Limitation: We constructed our dataset by randomly sampling videos and observed a prevalence of video games with an open-world genre on the Reddit website. Consequently, during our sampling process, video games from this genre, characterized by their distinct mechanics, were more frequently represented compared to other types.

6. Conclusion

We introduce GlitchBench, a new challenging benchmark for evaluating multimodal models on the video game glitch detection task. Detecting glitches requires various levels of reasoning skills, such as an understanding of the laws of physics and commonsense, making it well-suited for testing the generalization capabilities of large multimodal models. Comparing models' performance on various multimodal benchmarks and GlitchBench reveals a disparity: High performance on prior benchmarks does not guarantee high performance on real-world tasks that demand extensive reasoning abilities. We show that GlitchBench, derived from real-world video game quality assurance, presents a new challenge for the AI community and is a valuable addition to existing multimodal benchmarks.

Acknowledgement: AN is supported by the NaphCare Foundation, Adobe Research gifts, and NSF grant no. 2145767.

References

[1] Report: Epic games business breakdown & founding story. https://research.contrary.com/reports/epic-games. (Accessed on 11/15/2023). 1

[2] OpenAI's GPT-4V(ision), 2023. 2, 5, 6

[3] Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. Nocaps: Novel object captioning at scale. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8948–8957, 2019. 2

[4] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022. 2

[5] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023. 2

[6] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 5, 6

[7] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. Introducing our multimodal models, 2023. 5, 6

[8] Yonatan Bitton, Hritik Bansal, Jack Hessel, Rulin Shao, Wanrong Zhu, Anas Awadalla, Josh Gardner, Rohan Taori, and Ludwig Schimdt. Visit-bench: A benchmark for vision-language instruction following inspired by real-world use. *arXiv preprint arXiv:2308.06595*, 2023. 2, 5

[9] Nitzan Bitton-Guetta, Yonatan Bitton, Jack Hessel, Ludwig Schmidt, Yuval Elovici, Gabriel Stanovsky, and Roy Schwartz. Breaking common sense: Whoops! a vision-and-language benchmark of synthetic and compositional images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2616–2627, 2023. 2

[10] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023. 2

[11] Jun Chen, Deyao Zhu1 Xiaoqian Shen1 Xiang Li, Zechun Liu2 Pengchuan Zhang, Raghuraman Krishnamoorthi2 Vikas Chandra2 Yunyang Xiong, and Mohamed Elhoseiny. Minigpt-v2: Large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*, 2023. 5, 6

[12] Ke Chen, Yufei Li, Yingfeng Chen, Changjie Fan, Zhipeng Hu, and Wei Yang. Glib: towards automated test oracle for graphically-rich applications. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 1093–1104, 2021. 1

[13] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 2

[14] Myung Jin Choi, Antonio Torralba, and Alan S. Willsky. Context models and out-of-context objects. *Pattern Recognition Letters*, 33(7):853–862, 2012. Special Issue on Awards from ICPR 2010. 2

[15] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1): 37, 1960. 6

[16] Chenhang Cui, Yiyang Zhou, Xinyu Yang, Shirley Wu, Linjun Zhang, James Zou, and Huaxiu Yao. Holistic analysis of hallucination in gpt-4v (ision): Bias and interference challenges. *arXiv preprint arXiv:2311.03287*, 2023. 3, 7

[17] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 5, 6

[18] Stella Frank, Emanuele Bugliarello, and Desmond Elliott. Vision-and-language or vision-for-language? on cross-modal influence in multimodal transformers. *arXiv preprint arXiv:2109.04448*, 2021. 2

[19] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 2, 4, 8

[20] Tejas Gokhale, Hamid Palangi, Besmira Nushi, Vibhav Vineet, Eric Horvitz, Ece Kamar, Chitta Baral, and Yezhou Yang. Benchmarking spatial relationships in text-to-image generation. *arXiv preprint arXiv:2212.10015*, 2022. 2

[21] Google. Bard, 2023. 3

[22] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017. 2, 8

[23] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617, 2018. 2

[24] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 2

[25] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021. 2

[26] Tuomo Hiippala, Malihe Alikhani, Jonas Haverinen, Timo Kalliokoski, Evanfiya Logacheva, Serafina Orekhova, Aino

Tuomainen, Matthew Stone, and John A Bateman. Ai2d-rst: A multimodal corpus of 1000 primary school science diagrams. *Language Resources and Evaluation*, 55:661–688, 2021. 2

[27] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019. 2

[28] Huaizu Jiang, Xiaojian Ma, Weili Nie, Zhiding Yu, Yuke Zhu, and Anima Anandkumar. Bongard-hoi: Benchmarking few-shot visual reasoning for human-object interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19056–19065, 2022. 2

[29] Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's" up" with vision-language models? investigating their struggle with spatial reasoning. *arXiv preprint arXiv:2310.19785*, 2023. 2

[30] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 235–251. Springer, 2016. 8

[31] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022. 6

[32] Chris Lewis, Jim Whitehead, and Noah Wardrip-Fruin. What went wrong: a taxonomy of video game bugs. In *Proceedings of the fifth international conference on the foundations of digital games*, pages 108–115, 2010. 4, 5

[33] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023. 2, 4, 8

[34] Bo Li, Peiyuan Zhang, Jingkang Yang, Yuanhan Zhang, Fanyi Pu, and Ziwei Liu. Otterhd: A high-resolution multi-modality model. 2023. 5, 6

[35] Chunyuan Li. Large multimodal models: Notes on cvpr 2023 tutorial. *arXiv preprint arXiv:2306.14895*, 2023. 2

[36] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022. 2

[37] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023. 8

[38] Ziyi Lin, Chris Liu, Renrui Zhang, Peng Gao, Longtian Qiu, Han Xiao, Han Qiu, Chen Lin, Wenqi Shao, Keqin Chen, Jiaming Han, Siyuan Huang, Yichi Zhang, Xuming He, Hongsheng Li, and Yu Qiao. Sphinx: The joint mixing of weights, tasks, and visual embeddings for multi-modal large language models, 2023. 2, 5, 6

[39] Carlos Ling, Konrad Tollmar, and Linus Gisslén. Using deep convolutional neural networks to detect rendered glitches in video games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, pages 66–73, 2020. 1, 4

[40] Fuxiao Liu, Tianrui Guan, Zongxia Li, Lichang Chen, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusionbench: You see what you think? or you think what you see? an image-context reasoning benchmark challenging for gpt-4v (ision), llava-1.5, and other multi-modality models. *arXiv preprint arXiv:2310.14566*, 2023. 2

[41] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Aligning large multi-modal model with robust instruction tuning. *arXiv preprint arXiv:2306.14565*, 2023. 5

[42] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 2, 5, 6

[43] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 2

[44] Shilong Liu, Hao Cheng, Haotian Liu, Hao Zhang, Feng Li, Tianhe Ren, Xueyan Zou, Jianwei Yang, Hang Su, Jun Zhu, et al. Llava-plus: Learning to use tools for creating multimodal agents. *arXiv preprint arXiv:2311.05437*, 2023. 2

[45] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023. 2, 8

[46] Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. *arXiv preprint arXiv:2110.13214*, 2021. 2

[47] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022. 2

[48] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023. 2

[49] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 3195–3204, 2019. 2

[50] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *2019 international conference on document analysis and recognition (ICDAR)*, pages 947–952. IEEE, 2019. 2

[51] Alfredo Nantes, Ross Brown, and Frederic Maire. A framework for the semi-automatic testing of video games. In *Pro-*

ceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, pages 197–202, 2008. 1

[52] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 2

[53] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015. 2

[54] Haotong Qin, Ge-Peng Ji, Salman Khan, Deng-Ping Fan, Fahad Shahbaz Khan, and Luc Van Gool. How good is google bard's visual understanding? an empirical study on open challenges. *Machine Intelligence Research*, 20(5):605–613, 2023. 3

[55] Farrukh Rahman. Weak supervision for label efficient visual bug detection. *arXiv preprint arXiv:2309.11077*, 2023. 1, 4

[56] Geeta Rani, Upasana Pandey, Aniket Anil Wagde, and Vijaypal Singh Dhaka. A deep reinforcement learning technique for bug detection in video games. *International Journal of Information Technology*, 15(1):355–367, 2023. 1

[57] Grand View Research. Video game market size, share and growth report, 2030, 2023. 1

[58] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. *arXiv preprint arXiv:1809.02156*, 2018. 7

[59] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 2

[60] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European Conference on Computer Vision*, pages 146–162. Springer, 2022. 8

[61] Noah Shinn, Beck Labash, and Ashwin Gopinath. Reflexion: an autonomous agent with dynamic memory and self-reflection. *arXiv preprint arXiv:2303.11366*, 2023. 4

[62] Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. Textcaps: a dataset for image captioning with reading comprehension. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 742–758. Springer, 2020. 2

[63] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019. 2

[64] Mohammad Reza Taesiri, Giang Nguyen, Sarra Habchi, Cor-Paul Bezemer, and Anh Nguyen. Imagenet-hard: The hardest images remaining from a study of the power of zoom and spatial biases in image classification. 2

[65] Mohammad Reza Taesiri, Moslem Habibi, and Mohammad Amin Fazli. A video game testing method utilizing deep learning. *Iran Journal of Computer Science*, 17(2), 2020. 1, 4

[66] Mohammad Reza Taesiri, Finlay Macklon, and Cor-Paul Bezemer. Clip meets gamephysics: Towards bug identification in gameplay videos using zero-shot transfer learning. In *Proceedings of the 19th International Conference on Mining Software Repositories*, pages 270–281, 2022. 1, 3

[67] Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5238–5248, 2022. 2

[68] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 5

[69] Andrew Truelove, Eduardo Santana de Almeida, and Iftekhar Ahmed. We'll fix it in post: what do bug fixes in video game update notes tell us? In *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pages 736–747. IEEE, 2021. 4

[70] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022. 6

[71] Licheng Wen, Xuemeng Yang, Daocheng Fu, Xiaofeng Wang, Pinlong Cai, Xin Li, Tao Ma, Yingxuan Li, Linran Xu, Dengke Shang, Zheng Zhu, Shaoyan Sun, Yeqi Bai, Xinyu Cai, Min Dou, Shuanglu Hu, and Botian Shi. On the road with gpt-4v(ision): Early explorations of visual-language model on autonomous driving, 2023. 3

[72] Benedict Wilkins and Kostas Stathis. Learning to identify perceptual bugs in 3d video games. *arXiv preprint arXiv:2202.12884*, 2022. 1

[73] Yang Wu, Shilong Wang, Hao Yang, Tian Zheng, Hongbo Zhang, Yanyan Zhao, and Bing Qin. An early evaluation of gpt-4v (ision). *arXiv preprint arXiv:2310.16534*, 2023. 3

[74] Zhiling Yan, Kai Zhang, Rong Zhou, Lifang He, Xiang Li, and Lichao Sun. Multimodal chatgpt for medical applications: an experimental study of gpt-4v, 2023. 3

[75] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023. 2

[76] Zhuolin Yang, Wei Ping, Zihan Liu, Vijay Korthikanti, Weili Nie, De-An Huang, Linxi Fan, Zhiding Yu, Shiyi Lan, Bo Li, et al. Re-vilm: Retrieval-augmented visual language model for zero and few-shot image captioning. *arXiv preprint arXiv:2302.04858*, 2023. 2

[77] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers

large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023. 2

[78] Zhenfei Yin, Jiong Wang, Jianjian Cao, Zhelun Shi, Dingning Liu, Mukai Li, Lu Sheng, Lei Bai, Xiaoshui Huang, Zhiyong Wang, et al. Lamm: Language-assisted multimodal instruction-tuning dataset, framework, and benchmark. *arXiv preprint arXiv:2306.06687*, 2023. 5

[79] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023. 2

[80] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6720–6731, 2019. 2

[81] Ming Zhao, Peter Anderson, Vihan Jain, Su Wang, Alexander Ku, Jason Baldridge, and Eugene Ie. On the evaluation of vision-and-language navigation instructions. *arXiv preprint arXiv:2101.10504*, 2021. 7

[82] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. 5

[83] Yan Zheng, Xiaofei Xie, Ting Su, Lei Ma, Jianye Hao, Zhaopeng Meng, Yang Liu, Ruimin Shen, Yingfeng Chen, and Changjie Fan. Wuji: Automatic online combat game testing using evolutionary deep reinforcement learning. In *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 772–784. IEEE, 2019. 1

[84] Kankan Zhou, Eason Lai, Wei Bin Au Yeong, Kyriakos Mouratidis, and Jing Jiang. Rome: Evaluating pre-trained vision-language models on reasoning beyond visual common sense. *arXiv preprint arXiv:2310.19301*, 2023. 2

[85] Yiyang Zhou, Chenhang Cui, Jaehong Yoon, Linjun Zhang, Zhun Deng, Chelsea Finn, Mohit Bansal, and Huaxiu Yao. Analyzing and mitigating object hallucination in large vision-language models. *arXiv preprint arXiv:2310.00754*, 2023. 7