Don't Fight Hallucinations, Use Them: Making Use of LVLM Hallucinations to Quantify the Image Realism

Elisei Rykov[1], Kseniia Petrushina[1,2], Kseniia Titova[3], Alexander Panchenko[1,4], Vasily Konovalov[2,4]

[1]Skoltech, [2]MIPT, [3]MTS AI, [4]AIRI

Abstract

Quantifying the realism of images remains a challenging problem in the field of artificial intelligence. For example, an image of Albert Einstein holding a smartphone violates common-sense because modern smartphone were invented after Einstein's death. We introduce a novel method to assess image realism using Large Vision-Language Models (LVLMs) and Transformer-based encoder. Our approach rests on the premise that LVLMs may generate hallucinations when confronted with images defying common sense. By leveraging LVLM to extract atomic facts from these images, we obtain a mix of accurate facts and erroneous hallucinations. We proceed by fine-tuning a compact attention-pooling classifier over encoded atomic facts. This process serves to identify contradictions between genuine facts and the hallucinatory elements, thereby signaling the presence of images that violate common sense. Our approach has achieved a new state-of-the-art performance on the WHOOPS! dataset while leveraging a compact fine-tuning component.[1]

1 Introduction

When presented with an unusual image, human perception swiftly detects discordant elements. For example, am image of Einstein holding a smartphone can appear ordinary in their components yet strikingly abnormal in their arrangement. While humans intuitively spot the non-sense of the image, the cognitive process behind this is intricate. Linking visual elements to common-sense extends beyond simple object recognition (Zellers et al., 2019).

In this work we propose a visual commonsense classifier that utilizes observation that LVLMs may generate hallucinations when confronted with images defying common sense (Liu et al., 2024b). By leveraging LVLMs to extract atomic facts from these images, we obtain a mix of accurate facts and erroneous hallucinations. Then we fine-tune a compact attention-pooling model over encoded atomic facts (Figure 1).

The attention-pooling classifier learns to aggregate atomic facts in a manner that enables it to identify contradictions between genuine facts and hallucinatory elements, thereby signaling the presence of images that violate common sense.

Our findings suggest that rather than relying on intricate models, we can effectively exploit the imperfections of LVLMs – their tendency to generate hallucinations – in conjunction with a classifier over encoded atomic facts to identify images that defy common sense. Remarkably, our approach surpasses all open solutions (Guetta et al., 2023).

Our contributions are as following:

- We confirmed the previous observation that LVLMs may generate hallucinations when confronted with images defying common sense (Liu et al., 2024b). We then leveraged these imperfections to solve the task of detecting unusual images.
- By leveraging this observation we built a simple yet effective attention-pooling classifier over encoded representation of atomic facts.
- The fine-tuned classifier achieved the state-of-the-art performance on the WHOOPS! benchmark.
- In order to validate our results, we synthesized WIERD datasets containing 400 normal/strange images. Using this dataset, we confirmed our findings.

2 Related Work

Recently, commonsense reasoning has attracted substantial interest, spanning across disciplines within Natural Language Processing (NLP) and

[1]The code and the dataset will be available online

Generated atomic facts

*Santa Claus is riding a **reindeer** through a forest.*
*Santa Claus is depicted riding **reindeer** in this image.*
*The **reindeer** are pulling a sleigh with Santa Claus.*

Generated atomic facts

*Santa Claus is riding a **horse**-drawn sleigh.*
*Santa Claus riding a **horse** in a snowy landscape.*
*Santa Claus is riding a sleigh pulled by **reindeer**.*

Figure 1: A pair of images from the WHOOPS! dataset with corresponding atomic facts. The normal image is on the left, and the unusual image is on the right. All the facts associated with the normal image are consistent and accurately describe the image. However, in the case of the weird image, LVLM hallucinates and generates untruthful facts, such as *"Santa Claus is riding a sleigh pulled by reindeer."*

computer vision (CV), with numerous tasks being introduced.

Guetta et al. (2023) introduced the WHOOPS![2] benchmark, comprising images specifically designed to challenge common sense. In their effort to detect unconventional images, they employed BLIP-2 Flan-T5 (Li et al., 2023a) at multiple scales. While the fine-tuned model managed to outperform a random baseline, it still falls significantly short of human performance.

Theis (2024) provided a theoretically basis why quantifying realism is challenging. He suggests to consider not just the visual fidelity of images but also the contextual and commonsense coherence. An image might appear visually convincing but still fail to align with our understanding of how the world works, thereby diminishing its realism.

Liu et al. (2024b) introduced a novel PhD benchmark, which includes a subset of counter-commonsense images. This subset consists of unrealistic images accompanied by questions related to them.[3]

LLMs are capable of producing highly fluent responses to a wide range of user prompts, but they are notorious for hallucinating and making non-factual statements. Manakul et al. (2023b) proposed SelfCheckGPT, a straightforward sampling-based method that enables fact-checking of black-box models with zero resources.

To assess consistency among multiple sampled responses, SelfCheckGPT utilizes several techniques, including BERTScore, an automatic multiple-choice question answering generation (MQAG) framework (Manakul et al., 2023a), and NLI contradiction scores to detect hallucinations in the generated responses. However, the most effective method found was prompting the LLM to verify if the generations are supported by the context or not.

LVLM as well as LLM are known to hallucinate facts. Jing et al. (2023) proposed a benchmark and FAITHSCORE method that measures faithfulness of the generated free-form answers from LVLMs. The FAITHSCORE first identifies sub-sentences containing descriptive statements that need to be verified, then extracts a comprehensive list of atomic facts from these sub-sentences, and finally conducts consistency verification between fine-grained atomic facts and the input image. The results show that current LVLMs despite doing well on color and counting, still struggle with long answers, relations, and multiple ob- jects.

Our approach is similar to the preceding methods, as we also utilize LVLMs to extract atomic facts from the image. We then fine-tune a supervised model to learn the relationships between the derived facts. If the classifier identifies a high contradiction among the atomic facts, it indicates that

[2]CC-By 4.0

[3]Unfortunately, the dataset could not be located in open-source repositories.

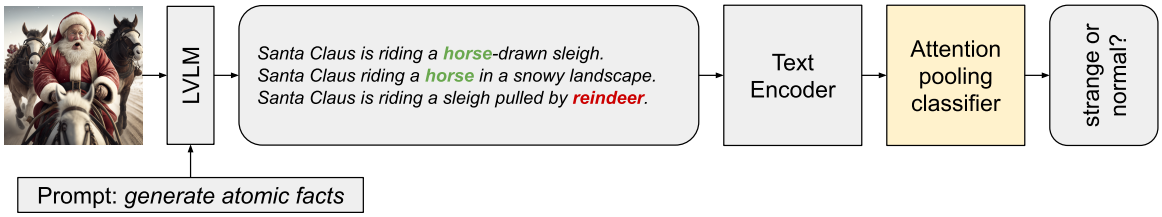

Figure 2: The proposed approach comprises the following pipeline steps: (i) prompt the LVLM to generate multiple atomic facts that describe the image; (ii) encode the generated atomic facts using a frozen (non-learnable) text encoder; (iii) fine-tune an attention-pooling classifier on the encoded atomic facts.

one of the generated atomic facts is likely a hallucination. This often occurs when the LVLM has encountered an unusual image (Liu et al., 2024b), which leads to such inconsistencies in most cases.

3 Dataset

To evaluate our methods, we employ the WHOOPS![4] benchmark, focusing on a subset comprising 100 pairs of weird and normal images. Performance is measured by binary accuracy within this paired dataset, where a random guess would yield 50% accuracy. To assess human performance, three annotators were enlisted to categorize each image as weird or normal, relying on a majority vote for the final determination. Impressively, human agreement reached 92%, indicating that, despite subjectivity, there is a clear consensus on what constitutes weirdness within the specific context of the WHOOPS! benchmark.

Furthermore, to validate our methodology, we systematically generated a larger image realism benchmark – WEIRD[5]. Specifically, we sample data from the WHOOPS! dataset and employ proprietary GPT-4o to create new caption pairs: one for a normal image and another for a weird image. At each generation step, we pass 5 samples with the same common-sense discrepancy category from WHOOPS!, consisting of the normal image caption, strange image caption, and an explanation of the strange situation. The prompt asks to first reason and generate an explanation of the strange situation, and only then generate pair of captions. Subsequently, we utilized DALL·E to generate images for each caption. Approximately 400 pairs were created, but only 200 remained after manual filtering. Additionally, we excluded rare and challenging categories from WHOOPS!, such as those involving celebrities, since DALL·E had difficulty rendering accurate images of celebrities, often distorting the original caption's intent. A human achieves 94.07% accuracy on this dataset.

[4]**W**eird and **H**eterogene**O**us **O**bjects, **P**henomena, and **S**ituations

[5]**W**eird **E**xamples of **I**mages with **R**eal-life **D**iscrepancies

4 Problem Statement

The task is to obtain a quantitative score of the realism of a particular image. Given two sets with usual and weird images that contradict common sense, we need to find a method to separate real images from weird ones. We propose *reality-check* function $\mathrm{f}_{\text{real}}(\mathbf{x})$, which returns image reality score. For real image $\mathbf{x}_\text{r}$ and weird image $\mathbf{x}_\text{w}$ we get

$$\mathrm{f}_{\text{real}}(\mathbf{x}_\text{r}) > \mathrm{f}_{\text{real}}(\mathbf{x}_\text{w}).$$

5 Proposed Method

Our *reality-check* methods is based on three steps: (i) we prompt the LVLM to generate multiple atomic facts describing the image; (ii) we encode atomic facts with frozen (non-learnable) Transformer-based text encoder; (iii) finally, we train classifier based on learned aggregation technique on encoded atomic facts.

5.1 Atomic Fact Generation

The initial step of our approach is to generate a list of straightforward atomic facts about the input image.

$$F_A = LVLM(I, P)$$

Where I – an input image, P – a textual prompt to generate simple atomic facts about the image. We generate n different facts using the Diverse Beam Search (Vijayakumar et al., 2016).

5.2 Text Encoder

Given a set of generated atomic facts, some of which are hallucinations while others are truly ac-

curate, it is necessary to learn the distinction between these relationships, ensuring that the connection among genuine facts stands apart significantly from the connection between hallucinated and genuine facts. This problem could be solved by a Natural Language Inference (NLI) models or Paraphrase Detection models, where the relationship among genuine facts would result in a high paraphrase/entailment score, and the connection between real facts and hallucinations would yield a strong contradiction score[6]. However, instead of relying on predefined tasks, we can train a classifier to discern these relations. Prior to adjusting the classifier, we must encode the generated atomic facts. To demonstrate the impact of fine-tuning tasks on model performance, we employ encoders based on the *DeBERTa-v3-large* (304B) architecture, which have been fine-tuned on various tasks for this purpose.

5.3 Classification Model with Attention-Pooling

Our classification model utilizes an attention-pooling mechanism to aggregate the encoded atomic facts and determine whether an image is normal or strange. This approach is based on the learned aggregation technique proposed by Touvron et al. (2021).

The classifier consists of the following components: (i) a learnable query vector q_{cls}; (ii) an attention mechanism; (iii) a final classification layer.

The attention-pooling layer uses a learnable fixed vector q_{cls} instead of the input X to generate a class query Q_{cls}:

$$Q_{cls} = q_{cls} W^Q \tag{1}$$

$$\text{Att}_{cls}(Q_{cls}, K, V) = \text{softmax}\left(\frac{Q_{cls} K^T}{\sqrt{d_k}}\right) V \tag{2}$$

This approach allows the model to learn relationships between the encoded atomic facts, represented as multi-dimensional features. Unlike simple mean/max-pooling, which merely aggregates information, attention-pooling learns to weigh the importance of different features dynamically. The output of this layer is then fed into subsequent linear layer for final classification.

[6]The experiments leveraging NLI for this task can be found in Appendix G

6 Experimental Setup

To run experiments, we strictly follow the experimental setup suggested in WHOOPS! (Guetta et al., 2023). We evaluate the models using a 5-fold cross-validation in a supervised configuration. As baselines, we consider methods proposed in WHOOPS! (Guetta et al., 2023) and other baselines. These involve training a compact classifier over the hidden representation of LVLMs or CLIP-based encoders. The hyper-parameters are mentioned in the Appendix 5.

Baselines. Specifically, for the baselines we utilized three approaches: (1) Instruction-following LVLMs, where we simply prompt the LVLMs to detect whether an image is strange or normal; (2) Single layer classifier over the hidden representations of CLIP-based models; (3) Linear probing over the hidden representations of LVLMs, accompanied by prompts or conducted solely on images.

For the instruction-following baseline we leverage LVLM with the prompt, which was found to be effective in detecting weird images (Liu et al., 2024a): `<image> Is this unusual? Please explain briefly with a short sentence.` For the baselines and for generating atomic facts we leverage the following LVLMs:

- *llava-v1.6-mistral-7b-hf*[7]: a 7B LVLM with based on a Mistral (Jiang et al., 2023);
- *nanoLLaVA-1.5*[8]: a 2B LVLM based on a Qwen1.5-0.5B (Bai et al., 2023);
- *llava-v1.6-vicuna-7b-hf*[9]: a 7B LVLM based on a Vicuna (Chiang et al., 2023);
- *llava-v1.6-vicuna-13b-hf*[10]: a 13B LVLM based on a Vicuna.

Linear probing baseline resemble our approach in way that it requires a small learnable component as our approach. This baseline involves learning a logistic regression classifier on the hidden representation of LLaVAs at each layer. We consider two setups: (a) using the <image> as the sole input, and (b) using <image> the with a prompt `Provide a short, one-sentence descriptive fact about this image.`

Our next group of baselines involves supervised fine-tuning of various CLIP-based image encoders on the image binary classification task. During

[7]hf.co/llava-hf/llava-v1.6-mistral-7b-hf
[8]hf.co/qnguyen3/nanoLLaVA-1.5
[9]hf.co/llava-hf/llava-v1.6-vicuna-7b-hf
[10]hf.co/llava-hf/llava-v1.6-vicuna-13b-hf

Encoder	LLaVA Backbone			
	Mistral-7B	Vicuna-7B	Vicuna-13B	Qwen-0.5B
WEIRD				
deberta-v3-large-tasksource-nli	**87.57**	80.51	81.37	77.11
nli-deberta-v3-large	77.97	74.00	77.11	74.57
deberta-v3-large	59.92	63.86	63.59	63.29
WHOOPS!				
deberta-v3-large-tasksource-nli	**73.54**	69.15	64.72	64.68
nli-deberta-v3-large	64.60	63.61	66.59	65.15
deberta-v3-large	49.49	50.48	47.57	53.93

Table 1: The results of our approach with various LVLMs and text encoders for both benchmarks—WHOOPS! and WEIRD—are presented. Accuracy, averaged over five folds, serves as the performance metric. For both benchmarks, LLaVa 1.6 Mistral-7B paired with *deberta-v3-large-tasksource-nli* demonstrates the best outcome. A clear trend emerges: tasksource DeBERTa outperforms all others, partly due to its superior encoding capabilities. This trend is clearer for the WEIRD dataset due to its larger size.

fine-tuning, the image encoder is completely frozen except for the last classification head. We employ the following three image encoders:

- *clip-vit-base-patch32*[11]: a pre-trained CLIP model published by OpenAI with 0.15B parameters (Radford et al., 2021).
- *siglip-so400m-patch14-384*[12]: a novel image encoder with 0.88B parameters trained by Google. This encoder inherit CLIP architecture, but with a better loss function (Zhai et al., 2023).
- *CLIP-ViT-L-14-laion2B-s32B-b82K*[13]: a pre-trained CLIP encoder with 0.43B parameters, trained on LAION-2B dataset (Ilharco et al., 2021).

Moreover, for WHOOPS! dataset we mention two additional fine-tuning baselines based on BLIP2 (Li et al., 2023b): BLIP2 FlanT5-XL and BLIP2 FlanT5-XXL that were reported in Guetta et al. (2023).

Fact generation. To perform diverse atomic fact generation, we employ the aforementioned four LLaVAs.

For fact generation, we set `num_beams` and `num_beam_groups` to 5, and the `diversity_penalty` was set to 1.0. Regarding penalty, we find this value to be optimal for adding diversity and preserving the model's ability to follow instructions.

For LLaVAs, with its various backbone architectures, we utilized the following prompt for fact generation: `Provide a brief, one-sentence descriptive fact about this image.`

Text Encoder. Given the generated atomic facts, we encode them using several *DeBERTa-v3-large*-based text encoders, including:

- *microsoft/deberta-v3-large*: an original DeBERTa without fine-tuning;
- *cross-encoder/nli-deberta-v3-large*[14]: DeBERTa fine-tuned by Sentence Transformer (Reimers and Gurevych, 2019) on NLI datasets. Specifically the model was fine-tuned on the SNLI (Bowman et al., 2015) and MultiNLI (Williams et al., 2018) datasets.
- *sileod/deberta-v3-large-tasksource-nli*[15]: a multi-task text encoder based on DeBERTa-v3-large fine-tuned on 600 `tasksource` tasks, outperforming every publicly available text encoder of comparable size in an external evaluation (Sileo, 2024).

7 Results

The results, averaged over five folds, for the evaluated text encoders paired with various LLaVAs on both benchmarks are presented in Table 1. The highest performance for both benchmarks was attained by generating facts using LLaVA 1.6 Mistral 7B in conjunction with *deberta-v3-large-tasksource-nli* as the text encoder.

A distinct pattern emerges: DeBERTa models fine-tuned on the `tasksource` collection outperform methods relying on alternative text encoders,

[11] hf.co/openai/clip-vit-base-patch32
[12] hf.co/google/siglip-so400m-patch14-384
[13] hf.co/laion/CLIP-ViT-L-14-laion2B-s32B-b82K
[14] hf.co/cross-encoder/nli-deberta-v3-large
[15] hf.co/sileod/deberta-v3-large-tasksource-nli

Model	#	Mode	WHOOPS!	WEIRD
Humans	–	–	92	94.07
BLIP2 FlanT5-XL	3.94B	fine-tuned	60.00	71.47
BLIP2 FlanT5-XXL	12.4B	fine-tuned	73.00	72.31
BLIP2 FlanT5-XXL	12.4B		50.00	63.84
nanoLLaVA Qwen1.5 0.5B	1.05B		66.66	70.90
LLaVA 1.6 Mistral 7B	7.57B	zero-shot	56.86	62.14
LLaVA 1.6 Vicuna 7B	7.06B		65.68	79.66
LLaVA 1.6 Vicuna 13B	13.4B		56.37	63.00
LLaVa lp_{best}	–	fine-tuned	73.48	84.47
$CLIP_{best}$	–	–	60.78	85.89
$Ours_{best}$	–	fine-tuned	**73.54**	**87.57**
GPT-4o	–	zero-shot	79.90	81.64

Table 2: The results of different approaches on both benchmarks: WHOOPS! and WEIRD. Both benchmarks are balanced and accuracy is the evaluation metric. Fine-tuned methods are displayed at the top, while zero-shot methods are presented at the middle. The best linear probing results for all configuration along with our method are displayed at the bottom. The proprietary GPT-4o model has been included as a baseline to illustrate the complexity of the benchmarks for proprietary systems and to demonstrate the gap in performance between human-generated and proprietary systems. It should not be directly compared with the other open-source methods. The linear probing baselines results can be found in Appendix.

largely due to their enhanced encoding capabilities. This superiority can be attributed to extensive fine-tuning on a diverse range of knowledge-intensive tasks sourced from the `tasksource` repository. Using `tasksource` DeBERTa, the best performance was achieved with Mistral-7B backbone, while the poorest performance was observed with the smallest Qwen-0.5B model, and Vicuna fell in the middle.

The outcomes of our baselines, which were conducted using LLaVa and CLIP, are detailed in the Appendices C, B. In the case of the LLaVa models, the Vicuna 13B variant excelled with an accuracy of 73.48% when operating in image-only mode for WHOOPS!. Conversely, on the WEIRD dataset, the Vicuna 7B model performed supremely, achieving an accuracy of 84.47% under conditions where images were complemented by textual prompts.

As for the CLIP baseline, OpenAI/CLIP excelled with an accuracy of 60.78% in zero-shot mode for WHOOPS!. On the other hand, on the WEIRD dataset, Google/SigLIP performed supremely, achieving an accuracy of 85.89% in fine-tuning mode.

The best performing modes for both baselines are presented in the Table 2.

Next, we compare baselines and our top-performing approach with the baselines from Guetta et al. (2023). Not only does our method excel, but the LLaVas linear probing baselines also surpass the original fine-tuned approach (BLIP2-FLAN-T5-XXL) presented in Guetta et al. (2023). This indicates that the task of detecting anomalous images should be tackled by fine-tuning a compact classifier on either textual representations or images, rather than adjusting an entire LVLM model for this purpose.

Furthermore, in the Appendix G, we delve deeper and propose an unsupervised approach to detect unusual images by leveraging Natural Language Inference (NLI) to identify contradictions between trustworthy atomic facts and hallucinations. This method demonstrates its effectiveness by achieving an accuracy of 72.5% on the WHOOPS dataset, all without the need for labeled data.

Moreover, we prompted GPT-4o to illustrate the complexity of the benchmarks for proprietary systems and to demonstrate the gap in performance between human-generated content and proprietary systems (it should not be directly compared with other open-source methods). The results are rather surprising; GPT-4o outperforms all the methods mentioned here on the WHOOPS! dataset. However, it lags significantly behind all considered baselines and our method on newly generated WEIRD dataset.

8 Analysis of the Generated Facts

In order to provide insights into the model's performance, we manually observed the generated atomic

facts. We analyze atomic facts generated by all four selected LLaVAs.

For each LVLM we look at the average length of a generated fact, the presence of so-called marker words, as well as the semantic correlation between facts related to the same image. By marker words, we mean a set of specific words that can indicate certain attributes of the image. There are four sets of word markers related to the antonymous characteristics: the pair `common-hallucination` and the pair `real-digital`. Constructed sets are presented in Appendix D.

Measure	Value
$\mathbb{P}$(weird | hallucination)	0.65
$\mathbb{P}$(weird | digital)	0.65
$\mathbb{P}$(weird | hallucination & digital)	**0.81**
$\mathbb{P}$(digital | hallucination)	0.52
corr(weird, hallucination)	0.18
corr(weird, digital)	**0.30**

Table 3: The conditional probability of class 'weird' given the occurrence of the marker from the corresponding set of marker words are displayed at the top, while the correlation between class 'weird' and the set of marker words is presented at the bottom. $\mathbb{P}$(digital | hallucination)=0.52 indicates that sets `digital` and `hallucination` provide highly independent information.

We measure lexical/semantic similarity of the generations by using *e5-mistral*[16] (Wang et al., 2024) embeddings and computing cosine similarity; also we calculate ROUGE (Lin, 2004) metric. We calculate the metric values pairwise for each pair of facts and then averaging the results. There is no significant difference in lexical/semantic similarity (as measured by ROUGE and *e5-mistral*) between strange and normal images within the same LLaVa. However, a significant difference can be observed when comparing similarity between different LLaVAs. In Table 4 we provide metrics on generated atomic facts.

nanoLLaVA 1.5B generates significantly different facts from all others LLaVA models in terms of used vocabulary. By analyzing occurring marker words, it becomes evident that nanoLLaVA-1.5 more frequently employs words from the `common` and `hallucination` sets, indicating a greater tendency to comment on the plausibility of images and use evaluative terms. Conversely, it uses words from the `real` and `digital` sets less often. The facts of nanoLLaVA-1.5 are significantly longer than others.

LLaVA 1.6 Mistral 7B vs LLaVA 1.6 Vicuna 7B. The difference between facts generated by these two is quite noticeable. The Mistral based LLaVA generates the shorter responses, and judging by the ROUGE metric, these responses are less similar to each other. In terms of the atomicity of the generated facts, the facts produced by Mistral can be considered more qualitative. However, the presence of digital markers can be misleading for the model.

LLaVA 1.6 Vicuna 7B vs 13B. The metrics of both Vicuna-based models are largely identical; however, the generations from 13B are shorter on average. We also notice that the facts generated for strange images are generally longer than those for truthful ones.

Using the previously introduced sets of marker words indicating the artificial nature and weirdness of an image, we calculated the dependencies of these two characteristics. Empirical probabilities of the image being weird depending on the digital phenomena and Spearman correlation between them are presented in the Table 3.

Thus, the presence of `hallucination` and `digital` markers have a positive effect on determining image weirdness. Moreover, presence of both phenomena boosts the probability even further.

9 Conclusion

In this work, we propose a straightforward yet effective approach to visual common sense recognition. Our method exploits an imperfection in LVLMs, causing them to generate hallucinations when presented with unrealistic or strange images.

Our method entails transitioning to a text modality and addressing the problem from this perspective. Remarkably, despite the shift in modality, our approach surpasses previously established baselines as well as other supervised methods that have been applied within the image modality, including CLIP-based image encoders and the linear probing of LVLMs.

Our three-step process involves generating atomic facts, encoding atomic facts with Transformer-based text encoder, and training classifier based on attention-pooling to detect strange images.

[16] hf.com/intfloat/e5-mistral-7b-instruct

Model	Type	Length	ROUGE	e5-mistral	Marker words			
					common	weird	real	digital
llava-v1.6-mistral-7b	normal	61.80	0.4546	0.7965	9	1	33	37
	strange	64.34	0.4628	0.7957	5	12	19	68
nanoLLaVA-1.5	normal	140.15	0.4502	0.8319	55	4	20	8
	strange	144.01	0.4507	0.8336	46	26	17	17
llava-v1.6-vicuna-7b	normal	99.57	0.6471	0.8827	8	0	54	42
	strange	103.63	0.6375	0.8788	5	4	25	66
lava-v1.6-vicuna-13b	normal	86.69	0.6424	0.8824	8	0	21	37
	strange	92.88	0.6464	0.8813	4	8	15	58

Table 4: Metrics for generated atomic facts on the WHOOPS! dataset are computed separately for each of the four models, assessing them on both normal and strange images. ROUGE and *e5-mistral* metrics evaluate the similarity of facts derived from a single image, while marker words denote the presence of at least one characteristic marker term in the group of facts. From these results, we can conclude that the facts generated by *llava-v1.6-mistral-7b* are of the finest quality in atomicity — they are the briefest and exhibit the greatest semantic independence.

In addition, our methods outperformed proprietary GPT-4o on our newly generated WEIRD benchmark.

10 Limitations

We acknowledge that we did not consider all possible open LVLMs that became available recently. In addition, among the proprietary systems we only evaluated GPT-4o. Although we tested several prompts for zero-shot baselines and selected the best one, more profound research can be done in this regard.

11 Ethics Statement

We have carefully curated the generated WEIRD dataset, and we have not encountered any inappropriate or offensive content within it.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *Preprint*, arXiv:2309.16609.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 632–642. The Association for Computational Linguistics.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Nitzan Bitton Guetta, Yonatan Bitton, Jack Hessel, Ludwig Schmidt, Yuval Elovici, Gabriel Stanovsky, and Roy Schwartz. 2023. Breaking common sense: Whoops! A vision-and-language benchmark of synthetic and compositional images. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 2616–2627. IEEE.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. 2021. Openclip. If you use this software, please cite it as below.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Liqiang Jing, Ruosen Li, Yunmo Chen, Mengzhao Jia, and Xinya Du. 2023. FAITHSCORE: evaluating hallucinations in large vision-language models. *CoRR*, abs/2311.01477.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023a. Blip-2: Bootstrapping language-image pre-

training with frozen image encoders and large language models. *Preprint*, arXiv:2301.12597.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023b. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *Preprint*, arXiv:2301.12597.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. *Preprint*, arXiv:2310.03744.

Jiazhen Liu, Yuhan Fu, Ruobing Xie, Runquan Xie, Xingwu Sun, Fengzong Lian, Zhanhui Kang, and Xirong Li. 2024b. Phd: A prompted visual hallucination evaluation dataset. *CoRR*, abs/2403.11116.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023a. Mqag: Multiple-choice question answering and generation for assessing information consistency in summarization. *Preprint*, arXiv:2301.12307.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023b. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *Preprint*, arXiv:2303.08896.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *Preprint*, arXiv:2103.00020.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Damien Sileo. 2024. tasksource: A large collection of NLP tasks with a structured dataset preprocessing framework. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15655–15684, Torino, Italia. ELRA and ICCL.

Lucas Theis. 2024. What makes an image realistic? *CoRR*, abs/2403.04493.

Hugo Touvron, Matthieu Cord, Alaaeldin El-Nouby, Piotr Bojanowski, Armand Joulin, Gabriel Synnaeve, and Hervé Jégou. 2021. Augmenting convolutional networks with attention-based aggregation. *Preprint*, arXiv:2112.13692.

Ashwin K. Vijayakumar, Michael Cogswell, Ramprasaath R. Selvaraju, Qing Sun, Stefan Lee, David J. Crandall, and Dhruv Batra. 2016. Diverse beam search: Decoding diverse solutions from neural sequence models. *CoRR*, abs/1610.02424.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. *Preprint*, arXiv:2401.00368.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. From recognition to cognition: Visual commonsense reasoning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 6720–6731. Computer Vision Foundation / IEEE.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 11941–11952. IEEE.

A Attention-pooling Classifier Architecture Details and Training Hyperparameters

Hyperparameter	Value
Optimizer	AdamW
Batch size	256
Learning rate (LR)	1e-4
Epochs	55

Table 5: Attention-pooling classifier hyperparameters.

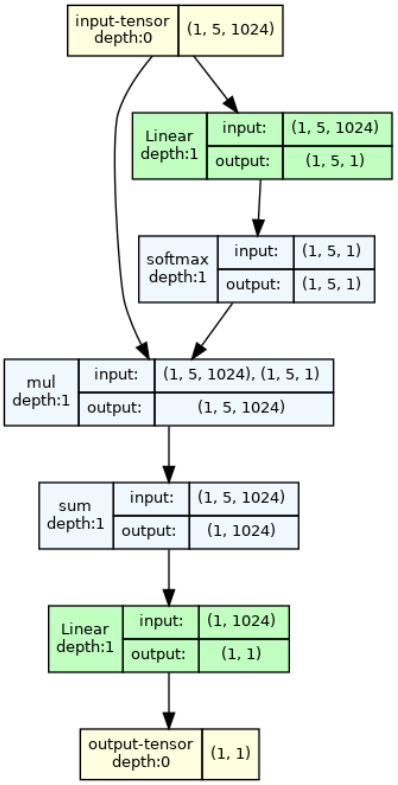

Figure 3: Architecture of the attention-pooling based classifier. In each of the five input atomic facts, we first extract the last hidden states for each token, then average the token hidden states according to the attention masks, and then pass five averaged hidden states to the classifier. Within the classifier, we aggregate five hidden states into one according to the attention scores. The final part of the classifier maps the aggregated hidden state into a binary class.

B LLaVA Baseline Results

We collect hidden states by passing only the `<image>` token (**Image only**) or `<image> Provide a short, one-sentence descriptive fact about this image.` prompt (**with Prompt**), which was used to generate atomic facts. The results are presented in the Table 6.

Model	Image only	with Prompt
WHOOPS!		
LLaVA 1.6 Mistral 7B	68.59	66.13
LLaVA 1.6 Vicuna 7B	66.61	71.55
LLaVA 1.6 Vicuna 13B	**73.48**	73.04
WEIRD		
LLaVA 1.6 Mistral 7B	79.08	81.07
LLaVA 1.6 Vicuna 7B	82.77	**84.47**
LLaVA 1.6 Vicuna 13B	81.62	84.19

Table 6: LLaVa baselines results on WHOOPS! and WEIRD.

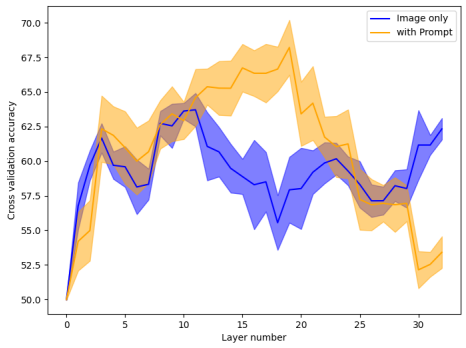

Figure 4: The dependence of cross-validation accuracy on the LLaVA 1.6 Mistral 7B layer index for linear probing on the WHOOPS! dataset. Our findings indicate that the layers containing the most pertinent information reside roughly in the middle of the decoder, as accuracy improves with increasing layer numbers when the model is prompted.

C CLIP Baseline Results and Hyperparamters

Model	#	zero-shot	fine-tuned
WHOOPS!			
OpenAI/CLIP	0.15B	**60.78**	56.86
Google/SigLIP	0.88B	50.49	57.40
LAION/CLIP	0.43B	53.92	54.39
WEIRD			
OpenAI/CLIP	0.15B	56.15	65.65
Google/SigLIP	0.88B	48.87	**85.89**
LAION/CLIP	0.43B	57.34	74.86

Table 7: CLIP baselines results on WHOOPS! and WEIRD.

Hyperparameter	Value
Optimizer	AdamW
Batch size	2
Learning rate (LR)	1e-3
Epochs	5

Table 8: CLIP baselines hyperparameters.

D Analysis on generated facts

The sets of marker words and the corresponding characteristics that we discussed in Section 8 are presented below. We acknowledge that words are used in context and their meaning can be highly dependent on it; therefore, the presence of these words in a sentence does not necessarily indicate the specified attribute of the fact.

Category	Keywords
common	*common* *usual* *normal* *natural* *real*
hallucination	*unusual* *strange* *playful* *creative* *unreal* *weird*
real (as not generated)	*real* *realistic* *photo*
digital	*digital* *generated* *3D* *fantastic* *rendering* *artistic*

Table 9: List of keywords with corresponding categories to analyze generated atomic facts.

E Examples of strange images from WEIRD

F Prompt for WEIRD samples generation using GPT-4o

```
Your task is to create new NORMAL_CAPTION and STRANGE_CAPTION using an EXAMPLES user will send you based on an EXPLANATION.
EXPLANATION is a description of an inconsistent situation. You should create EXPLANATION first and then NORMAL_CAPTION and STRANGE_CAPTION based on EXAMPLES.
NORMAL_CAPTION describes an image that is suitable for common sense, it does not contradict facts about the world, etc.
On the other hand, STRANGE_CAPTION contradicts common sense. Also, captions can represent past time, so a caption about something that happened a long time ago is not strange.
Also, EXAMPLES has a COMMONSENSE_CATEGORY. This is the category of common sense disturbance, so follow this information when creating your own captions, as they must disturb common sense in the same category.
These captions will be used to generate images with DALL-E, so please keep them in mind and make them clear and understandable.
You should generate the EXPLANATION why the STRANGE_CAPTION is strange and defies common sense, follow the EXAMPLES.
Do not generate something that is too hard to understand or imagine.
Make the captions as specific and descriptive as possible. Describe all the details.
Generate only ONE pair.

COMMONSENSE_CATEGORY: Incorrect usage

EXPLANATION: A traffic light with three lights contains three meaningful colors, red to stop, yellow to prepare, and green to go, so three green lights make it useless to route traffic.
NORMAL_CAPTION: a traffic light on a pole
only green lightsA street light with

EXPLANATION: Basketball is made of rubber and designed to be both bouncy and rough to be easily grasped, so playing basketball using a soccer ball designed for other purposes is pointless.
NORMAL_CAPTION: a basketball is going through the hoop
STRANGE_CAPTION: A soccer ball is being shot into a basketball hoop

EXPLANATION: Soccer is played with a lightweight and flexible ball, not with a bowling ball which is heavy and can cause injuries.
NORMAL_CAPTION: a person kicking a soccer ball on a field
STRANGE_CAPTION: A soccer player is about to kick a bowling ball

EXPLANATION: Desktop computers are designed to be used on a flat surface like a desk, so it is impractical to use a desktop computer while sitting on a bicycle.
NORMAL_CAPTION: A person typing on a desktop computer at a desk
STRANGE_CAPTION: A person using a desktop computer while riding a bicycle

EXPLANATION: People usually brush their hair with a comb, and a bald person has no hair to brush.
NORMAL_CAPTION: a man combing his hair
STRANGE_CAPTION: A bald man is holding a hair comb.
```

Figure 5: Example of prompt used for synthetic samples generation for WEIRD benchmark. In total, 5 random examples from WHOOPS! for some random common sense discrepancy category were taken on each step of generation. The model is expected to generate a new explanation and a pair of caption. Furthermore, captions are used for image generation.

Algorithm 1 Minimum Aggregation

Require: nli_scores
1: **return** min(nli_scores)

Algorithm 2 Absolute Maximum Aggregation

Require: nli_scores
1: abs_max_index ← argmax(|nli_scores|)
2: **return** nli_scores[abs_max_index]

Algorithm 3 Clustering Aggregation

Require: nli_scores
1: kmeans ← KMeans(n_clusters=2)
2: kmeans.fit(nli_scores)
3: centroids ← kmeans.cluster_centers_
4: **return** min(centroids)

Minimum (`min`): for a given list of scores, we simply select the lowest value as the metric. We assume that the lowest value could represent the contradictory of the whole set of facts.
Absolute maximum (`absmax`): we transform all values from the list of scores to their absolute values, then select the index of the largest absolute value and return the value from the original list to preserve the sign of the original value. So, if some set of facts has a relatively strong contradiction, we choose it as a weird image and vice versa.
Clustering (`clust`): we run the K-means clustering algorithm on the set of individual scores to split them into 2 clusters and then select the centroid with the lowest value as the metric. The idea is similar to the `min` method, but instead of the lowest value over all, we select an average of the values from the lowest cluster. We expect that contradictory facts from the weird images will have lower cluster centers than a related one.

Figure 6: Aggregation strategies for NLI scores to detect strange image.

G Detecting weird images using Natural Language Inference

In this paper, we presented a supervised-based approach to detect non-realistic images. Our approach leverages the fact that LVLMs tend to hallucinate when they encounter strange images (Liu et al., 2024b). These hallucinations manifest in various generations, expressing themselves as confusions in facts about the image (Figure 1).

We employed a supervised approach and fine-tuned an attention-pooling classifier that learns relationships between the generated atomic facts.

However, for detecting hallucinated facts, we could have used already pre-trained models. In this appendix, we describe our additional experiments with pre-trained Natural Language Inference (NLI) models to detect contradictions between hallucinations and valid atomic facts in the WHOOPS! dataset. In most cases, these contradictions indicate that the image is strange.

G.1 Proposed method

Given five generated atomic facts for each image, we applied a pre-trained NLI model in a pairwise manner. As a result, we received 25 triplets of $(s_{ent}, s_{con}, s_{neu})$ indicating entailment, contradiction, and neutrality.

Then, for each ordered pair of facts $(f_i, f_j) \in \mathrm{F_A} \times \mathrm{F_A}$, we compute the entailment score over $\mathrm{f_{nli}}$. We aggregate them using a custom weights (that were chosen on the separate validation set: $w_{con} = -2.0$, $w_{ent} = 1.75$, $w_{neu} = 0.0$):

$$\mathrm{f_{nli}}(f_i, f_j) = w_{con} \cdot s_{con} + w_{neu} \cdot s_{neu} + w_{ent} \cdot s_{ent}$$

After weighting we employed three aggregation strategies to compute a single anomaly score for each image. These aggregation strategies are illustrated in Figure 6.

G.2 Experimental setup

For fact generation, we followed the same setup we used previously. As NLI encoder, we adopt three DeBERTa-v3 based cross-encoders of different sizes: small[17], base[18], and large[19]. All of these models have been tuned in a similar setup on the SNLI (Bowman et al., 2015) and MultiNLI (Williams et al., 2018) datasets.

[17] hf.co/cross-encoder/nli-deberta-v3-small
[18] hf.co/cross-encoder/nli-deberta-v3-base
[19] hf.co/cross-encoder/nli-deberta-v3-large

G.3 Results

The comparison of the described aggregation approaches and NLI models is shown in the Table 10. As a result, the `clustering` method stands out as one of the best performing. This implies that the aggregation of all contradiction scores is crucial, rather than focusing only on extreme values. Abruptly, the `min` approach is the best for the `base` and `small` models. In addition, the largest model (`nli-deberta-v3-large`) outperforms all others for `clust` and `absmax` methods, suggesting that it captures the essence of the problem more effectively.

Model	#	Method		
		`min`	`absmax`	`clust`
nli-deberta-v3-large	304M	66.67	68.14	**72.55**
nli-deberta-v3-base	86M	**65.69**	66.18	**65.69**
nli-deberta-v3-small	47M	**66.67**	62.75	66.18

Table 10: A comparison of various NLI models and distinct aggregation techniques for subset with 5 facts is provided, with accuracy as the evaluation metric.

Figure 7 displays a heatmap depicting the aggregated NLI scores utilizing the most effective method. Noticeably, the generated atomic fact `"The rock appears to be the man's head"` contradicts to all other facts.

Figure 7: A strange image from the WHOOPS! datasets with a corresponding heatmap of aggregated NLI scores.

The comparison of the described aggregation techniques and NLI models is shown in the Table 10.

As a result, the `clustering` method stands out as one of the best performing. This implies that the aggregation of all contradiction scores is crucial, rather than focusing only on extreme values. Abruptly, the `min` approach is the best for the `base` and `small` models. In addition, the largest model (`nli-deberta-v3-large`) outperforms all others for `clust` and `absmax` methods, suggesting that it captures the essence of the problem more effectively.