PatchTrust: Black-Box Hallucination Detection via Patch-Level Retrieval Scoring

Vaibhav Varshney
ServiceNow
vaibhav.varshney@servicenow.com

Manjunatha Naik MC
ServiceNow
manjunathanaik.mc@servicenow.com

Abstract

Vision-language models frequently hallucinate objects and attributes that lack grounding in the source image. Most detection methods require white-box access to model internals or task-specific training data, limiting their use with API-served or rapidly evolving VLMs. We observe that retrieval relevance scores, designed to measure query-document match, transfer directly to measuring claim-image grounding. PatchTrust applies this insight: it decomposes a VLM response into atomic claims and scores each claim against the source image using the late interaction MaxSim mechanism from a frozen ColPali encoder. Each claim token is matched to its best-aligned image patch. Claims that find no strong patch-level support are flagged as hallucinations. PatchTrust requires no training and no access to model internals, yet consistently outperforms single-vector similarity baselines and closes the gap to white-box detectors across five evaluation settings and two VLMs.

CCS Concepts

• **Information systems** → **Retrieval models and ranking**; • **Computing methodologies** → *Natural language processing*; *Computer vision*.

Keywords

vision-language models, hallucination detection, late interaction, multi-vector retrieval, trustworthy AI

ACM Reference Format:
Vaibhav Varshney and Manjunatha Naik MC. 2026. PatchTrust: Black-Box Hallucination Detection via Patch-Level Retrieval Scoring. In *Proceedings of Proceedings of the 16th ACM International Conference on Multimedia Retrieval (ICMR '26)*. ACM, New York, NY, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Ask a vision-language model to describe a street scene and it will confidently mention a stop sign that is not there. Current VLMs such as LLaVA [12], InstructBLIP [2], and Qwen2-VL [19] all exhibit this failure mode, termed object hallucination by Rohrbach et al. [17]. The problem persists across model generations and undermines any downstream task that treats VLM output as factual.

ICMR '26, Amsterdam, The Netherlands

ACM ISBN 978-1-4503-XXXX-X/2026/06
https://doi.org/XXXXXXX.XXXXXXX

Detecting which specific claims in a response lack visual support requires either access to the VLM's internal representations or an external grounding signal. White-box methods probe token logits [21], attention maps [6], or intermediate embeddings [16] and achieve strong detection rates, but they assume access to model weights. That assumption fails for API-served systems. CLIP-based scoring [5] works without model access, but it compresses the entire image and the entire response into one cosine similarity, destroying spatial evidence. Trained classifiers such as ViLU [9] recover some of that evidence at the cost of requiring labeled hallucination examples.

The gap is at the intersection of three requirements: per-claim spatial grounding, no model access, and no training. Multi-vector retrieval fills it. ColPali [3] encodes images as patch-level embeddings and text as token-level embeddings, then scores relevance through MaxSim. For each query token, it finds the maximum cosine similarity to any image patch and sums across tokens. This preserves exactly the spatial granularity that single-vector methods destroy.

We apply this retrieval scoring to hallucination detection. Our approach, *PatchTrust*, decomposes a VLM response into atomic noun-phrase claims, encodes each against the source image with ColPali, and assigns a length-normalized MaxSim grounding score. Tokens describing objects present in the image match strongly to corresponding patches. Tokens describing hallucinated objects find no well-matched patch, pulling the score down.

In this work, we make the following contributions:

- We introduce *PatchTrust*, a training-free, black-box hallucination detector that repurposes late-interaction retrieval scores as per-claim visual grounding signals.
- We show that PatchTrust outperforms single-vector similarity baselines across all evaluation conditions and matches white-box detectors on the hardest splits, a black-box method applicable to every setting without modification.

2 Related Work

2.1 Hallucination Detection in VLMs

Benchmarks like POPE [11], CHAIR [17], and AMBER [18] measure how often VLMs hallucinate, but not which specific claims are wrong. That detection problem has mostly been tackled with white-box methods. The logit lens [6] checks whether generated object tokens have visual support by probing the VLM's unembedding layer. ContextualLens [16] uses middle-layer embeddings instead, catching attribute and relation errors that final-layer logits miss. MTRE [22] aggregates signals from multiple token positions. All three need access to model weights.

On the black-box side, SelfCheckGPT [13] samples multiple responses and flags inconsistent claims. This avoids model internals

but requires repeated VLM queries, which gets expensive with API-served models. *PatchTrust* needs only a single response.

Separately, mitigation methods like visual contrastive decoding [10] and MARINE [20] intervene during generation itself. These are complementary to post-hoc detection.

2.2 Multi-Vector Retrieval and Late Interaction

ColBERT [8] introduced late interaction for text retrieval: each token gets its own embedding, and relevance is scored via MaxSim over all token-to-token similarities. ColPali [3] brought this to vision-language retrieval, encoding document pages as patch-level embeddings and scoring text queries against them. ColQwen [4] swaps the backbone to Qwen2-VL. These models were built for document search, but the same MaxSim that measures query-document relevance also measures claim-image grounding - a connection we exploit.

2.3 External Grounding Signals

CLIPScore [5] computes a single cosine similarity between generated text and the image. F-CLIPScore [15] improves on this with per-object scoring. RCD [1] retrieves visually similar images and uses the confidence gap to suppress hallucinations. ViLU [9] trains a cross-attention classifier over CLIP embeddings, achieving strong results on 16 datasets but requiring labeled data. *PatchTrust* occupies a different spot: patch-level multi-vector scoring, no retrieval of external images, no training.

3 Method

3.1 Problem Setup

Let $\mathcal{M}$ denote a VLM which generates a textual response $y = (y_1, \ldots, y_n)$ given an image I and a prompt p. Some tokens in y describe entities, attributes, or relations that lack grounding in I. The goal is to assign a trust score $s(c, I)$ to each atomic claim c extracted from y, where lower scores indicate likely hallucinations.

3.2 Claim Decomposition

We first decompose the response y into a set of atomic claims $C = \{c_1, c_2, \ldots, c_k\}$. For POPE-style binary responses, each response constitutes a single claim. For free-form captions, we segment y into noun-phrase-anchored claims using spaCy's dependency parse to extract noun chunks. Each claim c_j is a short phrase containing one object or attribute assertion, such as "a red car parked on the left" or "a person holding an umbrella." This decomposition follows the principle of FActScore [14], which breaks model outputs into atomic verifiable facts. Where FActScore uses an LLM to generate propositions, we use rule-based noun-phrase extraction from the dependency tree, keeping the full pipeline training-free and deterministic.

3.3 Late Interaction Grounding Score

The core scoring mechanism repurposes ColPali's late interaction for grounding rather than retrieval.

Given a frozen ColPali model with vision encoder E_v and text encoder E_t, we compute:

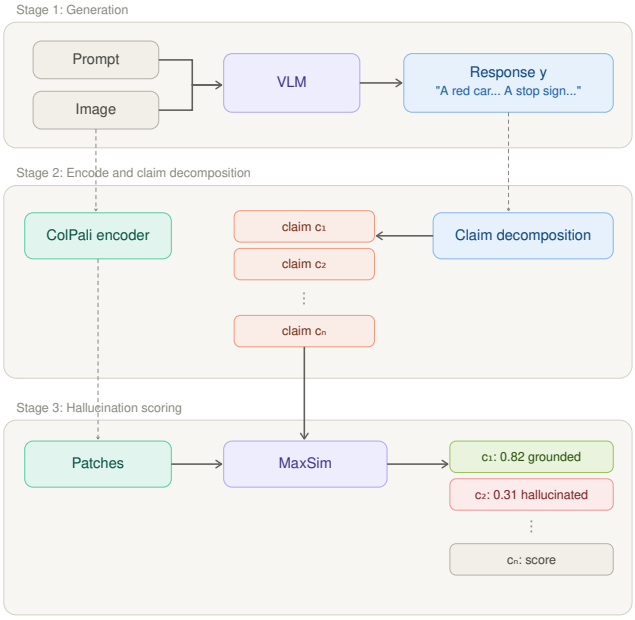

Figure 1: *PatchTrust* pipeline. Stage 1: Prompt and image feed into a VLM, producing response y. Stage 2: A frozen ColPali encoder converts the image into patch embeddings (left), while the response is decomposed into atomic claims $c_1 \ldots c_n$ (right). Stage 3: MaxSim scores each claim against the patches. Low-scoring claims are flagged as hallucinations.

Patch embeddings: $\mathbf{V} = E_v(I)$, where $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m\}$ and each $\mathbf{v}_i \in \mathbb{R}^d$ is an embedding of an image patch.
Claim embeddings: $\mathbf{T}_j = E_t(c_j)$, where $\mathbf{T}_j = \{\mathbf{t}_1^j, \mathbf{t}_2^j, \ldots, \mathbf{t}_l^j\}$ and each $\mathbf{t}_k^j \in \mathbb{R}^d$ is an embedding of a token in claim c_j.

The *PatchTrust* grounding score for claim c_j is:

$$PatchTrust(c_j, I) = \frac{1}{l} \sum_{k=1}^{l} \max_{i=1}^{m} \cos(\mathbf{t}_k^j, \mathbf{v}_i) \tag{1}$$

This is the MaxSim operator from ColBERT [8] normalized by claim length l to make scores comparable across claims of different lengths. For each token in the claim, the operator finds the single most similar image patch. Tokens describing objects present in the image match strongly to the corresponding patches. Tokens describing hallucinated objects find no well-matched patch, yielding uniformly low similarities that pull the aggregate score down.

3.4 Aggregation and Thresholding

For response-level hallucination detection, we aggregate claim scores into a single response score:

$$S(y, I) = \min_{j=1}^{k} \text{PatchTrust}(c_j, I) \tag{2}$$

We use the minimum rather than the mean because a single hallucinated claim should lower the trust of the entire response. This aligns with the observation from Jiang et al. [7] that a single early hallucination tends to corrupt subsequent generation.

A binary hallucination decision is obtained by thresholding $S(y, I)$ at a value τ. For POPE, we select τ on 20% of each split's questions (stratified by image), held out before evaluation; the remaining 80% form the test set. For CHAIR and AMBER, we use the POPE-derived threshold directly to test cross-benchmark transfer. We report threshold-free metrics (AUROC) as the primary measure, in addition to F1 at the optimal threshold.

4 Experimental Setup

4.1 Datasets and VLMs

We evaluate *PatchTrust* on three benchmarks spanning binary probing, free-form captioning, and multi-type discriminative evaluation. POPE [11] tests object hallucination on MSCOCO validation images across Random, Popular, and Adversarial negative sampling splits, with 500 images and 3,000 questions per split. CHAIR [17] evaluates free-form captions on 500 random MSCOCO validation images; we use CHAIR_S (binary: does the caption contain at least one hallucinated object?) to obtain a binary label per caption. AMBER [18] provides 15,281 discriminative yes/no questions covering existence, attribute, and relation hallucinations with deterministic LLM-free evaluation.

We evaluate outputs from LLaVA-1.5-7B [12] and InstructBLIP-7B [2]. These two models span a range of hallucination rates on POPE: LLaVA-1.5 exhibits higher object hallucination than InstructBLIP under adversarial sampling.

4.2 Baselines

We compare *PatchTrust* against four hallucination detection baselines. Token Logit Probability uses the average log-probability of generated tokens as a confidence score and requires white-box access. CLIPScore [5] computes cosine similarity between the full response and the image in CLIP ViT-L/14 space. F-CLIPScore [15] refines this with fine-grained per-object CLIP similarities. ContextualLens [16] uses middle-layer contextual embeddings for detection and requires white-box access.

4.3 Implementation Details

PatchTrust uses the frozen ColPali-v1.2 model based on PaliGemma-3B with a 128-dimensional projection layer. Images are processed at 448×448 resolution, yielding 1,024 patch embeddings. Claim decomposition uses spaCy for noun-phrase extraction. All experiments run on a single NVIDIA A100 40GB GPU. Scoring 500 MSCOCO images with their VLM-generated captions takes approximately 12 minutes.

4.4 Metrics

All three benchmarks provide binary labels: POPE and AMBER through yes/no question format, CHAIR through the CHAIR_S indicator. We report AUROC as a single unified metric across all evaluation settings, treating the *PatchTrust* grounding score as a binary classifier of response correctness. This enables direct comparison across datasets without metric-dependent confounds.

Table 1: Hallucination detection AUROC across datasets and VLMs. Best black-box result per row in bold. C.Lens = ContextualLens (white-box). "-" = method not applicable to setting.

Dataset	CLIP	F-CLIP	C.Lens	PatchTrust
VLM: LLaVA-1.5-7B				
POPE-Random	69.2	72.6	77.5	**77.1**
POPE-Popular	67.3	71.0	75.9	**75.4**
POPE-Adversarial	64.9	68.3	73.6	**74.8**
CHAIR_S	65.4	69.7	-	**73.2**
AMBER	67.5	70.9	75.1	**75.6**
VLM: InstructBLIP-7B				
POPE-Random	66.8	70.1	74.9	**74.3**
POPE-Popular	65.1	68.5	73.2	**72.8**
POPE-Adversarial	62.7	66.0	71.0	**71.5**
CHAIR_S	63.1	67.2	-	**70.8**
AMBER	65.2	68.6	72.7	**73.4**

5 Results and Analysis

5.1 Main Results

Table 1 presents hallucination detection performance across five evaluation settings and two VLMs. All values are AUROC, treating *PatchTrust*'s grounding score as a binary classifier of response correctness.

PatchTrust outperforms both black-box baselines (CLIPScore, F-CLIPScore) across all ten evaluation conditions, with gains of 7.5-9.9 AUROC points over CLIPScore and 3.5-6.5 points over F-CLIPScore. Against ContextualLens, a white-box method requiring access to model internals, *PatchTrust* wins on the hardest settings: POPE-Adversarial and AMBER for both VLMs. On the easier POPE-Random and POPE-Popular splits, ContextualLens leads by at most 0.6 points. The gain over F-CLIPScore is largest where fine-grained patch-level evidence matters most: POPE-Adversarial (+6.5 on LLaVA-1.5) and CHAIR_S (+3.5).

ContextualLens is not applicable to CHAIR_S because it requires probing VLM internals during discriminative question answering, which free-form captioning does not provide. *PatchTrust* is the only black-box method that covers all five evaluation settings without modification.

Token logit probability (white-box) scores 60.5-64.1 AUROC on POPE, below all external methods, and is omitted from the table.

All methods perform 2-4 AUROC points lower on InstructBLIP than on LLaVA-1.5. InstructBLIP hallucinates less frequently (~88.7% accuracy on POPE-Random vs. ~86% for LLaVA-1.5), reducing the hallucination signal available for detection. The relative ordering of methods is preserved across both VLMs.

5.2 Ablation Study

Table 2 isolates the contribution of each *PatchTrust* component.

Replacing MaxSim with AvgSim (averaging all token-patch similarities instead of taking per-token maxima) drops AUROC by 4.6 points, confirming that the token-level maximum operation is the

Table 2: Component ablation (AUROC on POPE-Adversarial, LLaVA-1.5). Component ranking is consistent across all five evaluation settings.

Variant	AUROC	Δ
Full PatchTrust	74.8	-
– MaxSim → AvgSim	70.2	−4.6
– ColPali → SigLIP (single-vec)	66.1	−8.7
– min agg → mean agg	73.1	−1.7
– claim decomposition	71.9	−2.9

Table 3: Claim-level hallucination detection against GPT-4o judge labels on 200 CHAIR images (LLaVA-1.5).

Method	Precision	Recall	F1
CLIPScore	0.54	0.61	0.57
F-CLIPScore	0.59	0.64	0.61
PatchTrust (ours)	**0.67**	**0.72**	**0.69**

key differentiator. Replacing ColPali with SigLIP single-vector scoring produces the largest degradation at 8.7 points, isolating the value of multi-vector patch-level representations. Claim decomposition contributes 2.9 points: scoring the full response as a single unit dilutes the signal from individual hallucinated objects.

5.3 Validation with LLM-as-a-Judge

The benchmark evaluations above rely on annotated ground-truth labels. A user deploying *PatchTrust* on a new VLM or domain has no such labels. We test whether *PatchTrust* captures hallucinations in this label-free setting using GPT-4o as an independent multimodal judge on the CHAIR evaluation set (LLaVA-1.5 captions, 200 images).

We decompose each caption into atomic claims (698 claims total) and present GPT-4o with each image-claim pair, prompting: "Does the image contain evidence supporting the following claim? Answer Yes or No." The LLM judge labels each claim as grounded or hallucinated. We then evaluate how well each detection method's scores separate the two classes, using the LLM labels as pseudo-ground-truth. Table 3 reports the results.

PatchTrust outperforms F-CLIPScore by 8 F1 points against the LLM judge (Table 3). The gap is wider than on $CHAIR_S$ (Table 1), where ground truth is based on object-list matching. GPT-4o catches attribute mismatches ("a wooden bench" when the bench is metal) and relation errors ("next to the fountain" when the fountain is absent) that CHAIR's keyword matching misses. *PatchTrust*'s patch-level scoring captures more of these fine-grained failures than single-vector baselines. GPT-4o is used here for validation only and is not part of the *PatchTrust* pipeline.

5.4 Error Analysis

PatchTrust's advantage over F-CLIPScore follows a gradient across hallucination types, visible in AMBER's per-type annotations. On existence questions, *PatchTrust* leads by 4.7 AUROC points (76.5 vs. 71.8, LLaVA-1.5). On attribute questions (state, number, action), the gap narrows to 3.2 points (67.8 vs. 64.6). On relation questions, *PatchTrust* leads by only 1.9 points (62.1 vs. 60.2).

The existence-to-relation gradient follows from MaxSim's design. Each claim token independently matches to its best-aligned image patch. Existence detection benefits directly: the token "dog" either has a high-similarity patch or it does not. Attribute detection is harder because the attribute token ("red") and the object token ("car") match to the same patch, leaving the color assertion unverified against a specific patch region. Relation detection fails most because "to the left of" requires reasoning about relative positions across patches, which per-token MaxSim does not capture.

ColPali's domain mismatch contributes to the weaker attribute and relation performance. ColPali was trained for document retrieval, and its patch embeddings are optimized for textual and tabular content rather than natural scene understanding. A backbone fine-tuned on natural image-caption pairs would likely yield stronger grounding for these claim types. The rule-based claim decomposition also introduces noise: it occasionally merges distinct claims or splits a single assertion into fragments, degrading per-claim scoring. *PatchTrust*'s PaliGemma backbone (3B parameters) adds overhead at inference (~45ms per image, ~8ms per claim on an A100), comparable to CLIPScore and lower than trained detectors.

6 Conclusion

MaxSim scores from a retrieval encoder, applied per-claim rather than per-document, produce grounding signals that separate hallucinated objects from real ones. *PatchTrust* exploits this without touching the VLM or training a classifier. The gains over single-vector baselines are largest on the hardest evaluation splits, where per-token patch matching recovers spatial evidence that global pooling destroys. On easier splits the advantage narrows, and on relation-type hallucinations it nearly vanishes. Per-token MaxSim checks whether individual tokens match individual patches. It does not reason about spatial relationships between patches, which is why "to the left of" defeats it while "a red car" does not.

PatchTrust's black-box design makes it directly applicable to API-served VLMs such as GPT-4o, Claude, and Gemini, where no internal state is available. We have not yet tested on these systems. Doing so would reveal whether the score distributions remain separable when the underlying VLM is much larger and hallucinates differently than LLaVA-1.5 or InstructBLIP. ColPali's document-retrieval training distribution is a second open question. Swapping the backbone to ColQwen or a scene-trained encoder would isolate how much of *PatchTrust*'s attribute and relation weakness comes from the late-interaction mechanism itself versus the training data mismatch.

References

[1] Guang Chen, Wenhan Zhang, Dayan Wu, et al. 2025. Retrieve-then-Compare Mitigates Visual Hallucination in Multi-modal Large Language Models. *Intelligent Robotics* 5, 2 (2025), 248–275.

[2] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[3] Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. 2025. ColPali: Efficient Document Retrieval with Vision Language Models. In *International Conference on Learning Representations (ICLR)*.

[4] Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. 2025. ColPali: Efficient Document Retrieval with Vision Language Models. In *International Conference on Learning Representations (ICLR)*. ColQwen2 variant replaces PaliGemma with Qwen2-VL.

[5] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: A Reference-free Evaluation Metric for Image Captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 7514–7528.

[6] Chaoya Jiang et al. 2024. Interpreting and Mitigating Hallucination in VLMs through the Logit Lens. *arXiv preprint arXiv:2402.11124* (2024).

[7] Zihao Jiang, Fangzhi Xu, Luyu Gao, et al. 2024. Investigating and Mitigating the Multimodal Hallucination Snowballing in Large Vision-Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

[8] Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 39–48.

[9] Marc Lafon et al. 2025. ViLU: Learning Vision-Language Uncertainties for Failure Prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[10] Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. 2024. Mitigating Object Hallucinations in Large Vision-Language Models through Visual Contrastive Decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[11] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Evaluating Object Hallucination in Large Vision-Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 292–305.

[12] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual Instruction Tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[13] Potsawee Manakul, Adian Liusie, and Mark J.F. Gales. 2023. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 9004–9017.

[14] Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 11532–11552.

[15] Hongseok Oh and Wonseok Hwang. 2025. Do Vision Encoders Truly Explain Object Hallucination? Mitigating Object Hallucination via Simple Fine-Grained CLIPScore. *arXiv preprint arXiv:2502.20034* (2025).

[16] Anirudh Phukan, Divyansh, Harshit Kumar Morj, Vaishnavi, Apoorv Saxena, and Koustava Goswami. 2025. Beyond Logit Lens: Contextual Embeddings for Robust Hallucination Detection and Grounding in VLMs. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics (NAACL)*. 9661–9675.

[17] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object Hallucination in Image Captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 4035–4045.

[18] Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, and Jitao Sang. 2023. AMBER: An LLM-free Multi-dimensional Benchmark for MLLMs Hallucination Evaluation. *arXiv preprint arXiv:2311.07397* (2023).

[19] An Yang et al. 2024. Qwen2-VL: Enhancing Vision-Language Model's Perception of the World at Any Resolution. *arXiv preprint arXiv:2409.12191* (2024).

[20] Linxi Zhao, Yihe Deng, Weitong Zhang, and Quanquan Gu. 2025. Mitigating Object Hallucination in Large Vision-Language Models via Image-Grounded Guidance. arXiv:2402.08680 [cs.LG] https://arxiv.org/abs/2402.08680

[21] Yue Zhao et al. 2025. First Token Probability as a Reliability Indicator for Vision-Language Models. *arXiv preprint arXiv:2501.09775* (2025).

[22] Geigh Zollicoffer, Minh Vu, and Manish Bhattarai. 2025. MTRE: Multi-Token Reliability Estimation for Hallucination Detection in VLMs. arXiv:2505.11741 [cs.AI] https://arxiv.org/abs/2505.11741