

IrumS at SHROOM-Visions 2026: When One Fix Doesn't Fit All — Category-Aware Hallucination Span Detection

Irum Shehryar

Independent Researcher

irum.shehryar@gmail.com

Abstract

This paper describes our system for the English subset of the SHROOM-Visions 2026 shared task, which requires detecting hallucinated spans in vision-language model outputs across Chinese, English, French, and Italian, and labeling them into five categories. We used Claude Haiku 4.5 as a zero-shot detector, with post-processing relabeling and Sonnet 5-based second-opinion verification for select categories. Our final system achieved 0.3015 (Cor+Lbl), 0.3949 (Cor) and 0.3421 (IoU). Our error analysis found that missed detections, low annotator agreement, and unresolved overlap between the invention and mischaracterization categories account for the majority of remaining errors, with additional score loss when the model finds the right content but its flagged text is a different length than gold's, or is labeled with the wrong category. Systematically testing and comparing individual approaches shows that none of the methods we tried is effective across all categories on its own; category-specific, targeted fixes improved performance more reliably than any single general-purpose mechanism we tested.

1 Introduction

Hallucination refers to content generated by a model that is inconsistent with the given input; while hallucination detection in text-based large language models (LLMs) has been widely studied, hallucination in large vision-language models (LVLMs) remains an ongoing challenge. Verifying factual claims requires comparing them against the image itself, introducing visual-grounding challenges that text-only detection does not face. It is further complicated by the range of distinct ways hallucination can occur (Liu et al., 2024). SHROOM-Visions (Vázquez et al., 2026) approaches this problem through a shared task, requiring teams to detect hallucination spans in

vision-language model outputs and label them into five categories. The task provides image-conditioned text collected from 5 different VLMs across four languages. We focused on the English data set and adopted a zero-shot LLM verifier approach, without fine-tuning. By applying post-processing relabeling and Sonnet 5-based second-opinion verification for select categories, our system¹ ranked 13/28, achieving a modest but real Cor+Lbl improvement (0.3015) over baseline (Table 1). Our analysis found that missed detections were a dominant source of errors, compounded by low annotator agreement (especially for mischaracterization) and overlap between invention and mischaracterization categories that no simple rule could separate. Scores are further reduced in cases where the model correctly locates the hallucinated content but its span differs in width from gold's, or where it identifies the right content but assigns the wrong category. We further found that among the corrections we tested, no single mechanism generalized across all categories, and targeted, category-specific fixes performed more reliably in our deployed pipeline, though one category-agnostic alternative we tested but did not scale up (Appendix B.2) showed a promising early signal. The remainder of the paper covers related work, followed by our system design, results, error analysis, proposed future work and limitations.

2 Related Work

SHROOM-Visions (Vázquez et al., 2026) is the fourth round of SHROOM shared task series, evolving from text-based binary classification (Mickus et al., 2024) to multilingual span-level (Vázquez et al., 2025) and cross-lingual detection (Sinha et al., 2025). Prior text-based systems used zero/few-shot classification (Allen et al., 2024),

¹Code available at <https://github.com/IrumShehryar/shroom-visions-hallucination-detection>

multi-step ensemble voting, where independent models verify candidate spans (Hikal et al., 2025) and retrieval-augmented verification against external context including a minimum-edit-distance variant that compares an original answer to a minimally-corrected version and labels the deleted or substituted words as hallucinations, avoiding free-text span extraction entirely (Huang et al., 2025). For end-to-end VLM hallucination evaluation, Liu et al. (2024) identify two broad categories: LLM-as-judge scoring, applied to multi-modal models directly by Sun et al. (2024), and fine-tuning on labeled hallucination datasets. In contrast, Azachi et al. (2025) detect hallucinations from next-token probabilities alone, avoiding a second model call.

3 System Overview

Our system follows the LLM-as-judge approach mentioned above, where an LLM is prompted with visual information and a candidate response to assess and score potential hallucinations. We used Claude Haiku 4.5 as the base detector to test whether a lower-cost model could serve as an effective zero-shot judge, since LLM-as-judge approaches do not require a judge model as capable as the model being evaluated. This is particularly relevant given the scale of the dataset, at 20,000 samples in four languages. The pipeline was developed iteratively, adding post-processing and second-opinion verification steps in response to category-specific failure patterns observed during development. Figure 1 shows the complete pipeline: detection, followed by post-processing and second-opinion verification, and a final span mapping step that maps the flagged text to character indices.

3.1 Prompt Design

The prompt receives an image, the question prompt, the candidate response, and a file name hint (background context only; the model is instructed to rely on visual evidence, not the filename), and returns a JSON array of objects, one per flagged span, each containing the hallucinated span, a category label, a hallucination probability, and a short reasoning. It gives short examples for each of the four categories (mischaracterization, miscounting, invention, and OCR), including a boundary clarification distinguishing invention (content absent from or unverifiable in the im-

age) from mischaracterization (content present but described incorrectly); these illustrate definitions rather than full worked examples, keeping the approach zero-shot. Hallucinations fitting none of the four, including responses that echo the question back, are labeled “other.” The prompt gives examples of correct and incorrect span extraction, such as extracting “four” rather than “four legs”, so spans stay minimal. The confidence scores follow a three-tier guide, with an instruction to withhold a flag rather than guess. For OCR, the model compares image text against the response word by word, abstaining when the text is too blurry to read confidently. The prompt was gradually refined over three iterations; all versions are provided in Appendix A.

3.2 Post-Processing Layer

Two post-processing steps were implemented to target systematic labeling errors observed during development. For the miscounting category, a post-processing step narrows the marked span to only the hallucinated count word. For mischaracterization labels, a second step checks whether the span actually represents mischaracterization rather than a numeral. For spans of three tokens or fewer, if a number is present (excluding percentages, clock-time expressions, and “one” when it ends a span or precedes a capitalized word), the label is replaced by miscounting which consequently, narrows the span to the number word as well; changing both category and boundaries together. This relabeling rule exists because the detector frequently confuses the two categories when a number is involved. Across the full labeled train set (3,799 rows), roughly 100 spans show this same ambiguity between annotators; of these, 40 have a clear preference towards one label, and miscounting is preferred in 33 of those 40 cases (82.5%). Unlike the three filters in Appendix B.1 that reversed on held-out data, this rule was validated on the 286-sample dev set and a separate 226-sample sanity check on an earlier prompt version, then confirmed as a real gain on the official leaderboard (Table 1).

3.3 Second-Opinion Verification

A second verifier, Claude Sonnet 5, was used to reverify hallucinated spans flagged by Haiku across two categories (miscounting and mischaracterization), since Haiku consistently overestimates its own confidence, consistent with Huang

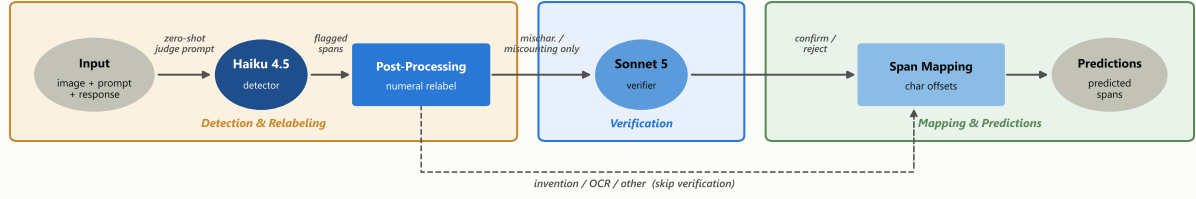


Figure 1: System pipeline for staged hallucination detection and verification.

et al. (2025)’s finding that self-reported LLM confidence scores tend to be uniformly high regardless of correctness. Across a 551-sample pool, false positives average 0.861 confidence (90.6% at 0.8+), against 0.890 (95.1% at 0.8+) for true positives. Sonnet is given the flagged span and Haiku’s stated reason, and directly confirms or rejects it; the flag is dropped if Sonnet disagrees, measurably improving scores in both categories. The same approach was also tested on invention flags but rejected as Sonnet’s catch rate on known false positives was only 20% (2/10), largely rubber-stamping Haiku’s original flags rather than independently re-checking them, and the real scorer confirmed a net-negative effect (Cor -0.0143 , Cor+Lbl -0.0230). Sonnet-based verification was not tested on OCR or “other,” since both had the lowest false-positive volumes in the 551-sample pool and were among the more reliably correct categories, leaving little apparent room for improvement.

3.4 Span Mapping

The final step maps each flagged span to the character-level start and end indices in the response. Before mapping, the system strips incidental boundary punctuation, the detector sometimes adds around a flagged span. It also tracks already-claimed character ranges, so repeated occurrences of the same span text, for example “four” appearing multiple times in one response, map to distinct positions instead of collapsing onto the first match. Beyond the deployed pipeline, several filters, a recall-focused verification approach, and EasyOCR corroboration were tried but not adopted (Appendix B.1, B.2, B.3).

4 Experimental Setup

Dataset: SHEEP (Mickus et al., 2026), the dataset underlying the SHROOM-Visions shared task, consists of a total of 20,000 samples across train and test splits in four languages (Chinese, English, French, Italian), combining model-

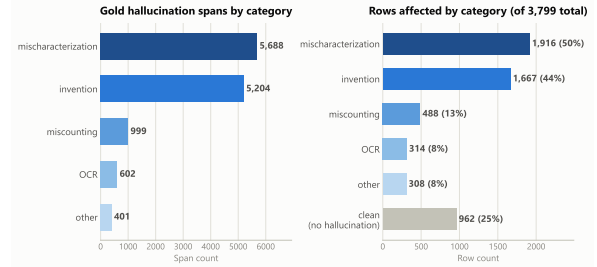


Figure 2: Gold Hallucination span counts and affected row counts by category, English train split (3,799 rows).

generated responses from 5 VLMs and human-written hallucination samples. The English subset that we used contains 3799 train rows each containing an image, a prompt, and a candidate response to be audited. Train rows additionally include gold labels each specifying a character-level span: **start/end** offsets, a **category**: one of invention, mischaracterization, miscounting, OCR, or other, and **prob**: the agreement level among three annotators.

Figure 2 shows the category distribution for the English train split (3,799 rows, 1,607 unique images). The train split exhibits substantial row repetition, with the majority of rows sharing an image with at least one other row, reflecting multiple candidate responses evaluated per visual question. The test split, used for official evaluation, contains 1,201 rows across 374 unique images with similar repetition structure, and without gold category labels.

Evaluation Metrics: Predictions are evaluated using three metrics, and ranked by Cor+Lbl.

Label-aware Correlation (Cor+Lbl) Spearman correlation between predicted and gold hallucination probabilities, requiring the predicted category to also match.

Correlation (Cor) The same measure, without the category requirement.

Stage	Cor+Lbl	Cor	IoU
Baseline	0.2868	0.3816	0.3287
+ Numeral relabeling fix	0.2897	0.3824	0.3284
+ Mischaracterization fix	0.3008	0.3934	0.3384
+ Miscounting fix	0.3015	0.3949	0.3421

Table 1: Official leaderboard progression across three confirmed fixes. Ranked 13th of 28 (Cor+Lbl) / 15th of 28 (Cor) / 15th of 28 (IoU).

Intersection-over-Union (IoU) Character-wise overlap between predicted and gold spans.

Sample Selection: We targeted 300 samples, stratified by primary gold label with a fixed random seed: 50 per category, including clean rows. Fourteen samples failed processing due to oversized images, yielding a final n of 286, used for error analysis and held constant across iterations. The category distribution of these 14 excluded samples was not tracked. Outcomes were scored using the official SHROOM-Visions scorer; both models ran at temperature 1.0 (the API default). In total, approximately 600 unique train-split samples were exercised across development, this fixed 286-sample set plus additional held-out batches at later validation stages, which surfaced overfitting in several candidate post-processing filters before deployment (Appendix B.1).

5 Results and Error Analysis

Our system achieved 0.3015 (Cor+Lbl), 0.3949 (Cor) and 0.3421 (IoU), across three official submission stages (Table 1). Each stage reuses the same cached Haiku detection output from the baseline run rather than re-querying the detector; the numeral-relabeling stage adds no model call at all, while the mischaracterization- and miscounting-fix stages each add one live Sonnet call per flagged span, cached once computed. Sonnet verification checks every flagged span in its target category; rows with no flag in that category are left unchanged. Our error analysis revealed several patterns and underlying causes of remaining mistakes which are discussed below.

Detection Failure Dominates Mislabeling: Comparing predictions against gold labels on the 286-sample dev set, missed detections substantially outnumbered wrong-label predictions in every category, prior to second-opinion verification. Mischaracterization and invention were missed most often: 189 of 245 gold mischaracterization

spans (77%) and 105 of 154 gold invention spans (68%) received no flag at all, while OCR was the most reliably caught category, with the highest annotator agreement (47.2% low-consensus, vs. 78.0% for mischaracterization).

Spatial Confusion: Haiku often treats close spatial phrasing, such as “on the shelf” versus “on top of the shelf,” as materially different claims that annotators never disputed. This also caused 59% of mischaracterization predictions to be pure false positives, versus 52% for invention, computed after the numeral relabeling fix but before Sonnet verification (Figure 3).

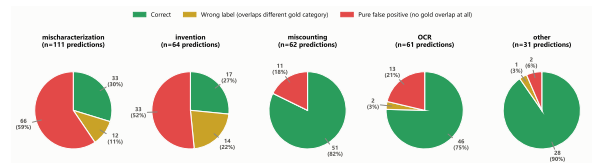


Figure 3: Category-wise breakdown of prediction outcomes on the 286-sample dev set (missed gold spans not shown). Mischaracterization and invention predictions are majority false positive; miscounting has no wrong-label predictions, only correct or pure-false-positive outcomes.

Low Human Agreement: Mischaracterization is the hardest of the four core hallucination categories to detect: 78.0% of its spans have the lowest possible annotator agreement (1 of 3), in the 286-sample set. Even human annotators struggled to agree on whether a hallucination was present, matching Allen et al. (2024)’s finding that models agree with human labels more often when the humans themselves agree.

Mischaracterization / Invention Overlap: Haiku also frequently confuses invention and mischaracterization. Pooling three sample subsets evaluated at different points in development (286 dev + 226 old-prompt + 39 held-out, 551 samples total), 81 spans were confused between the two categories (51 invention gold spans predicted as mischaracterization; 30 mischaracterization gold spans predicted as invention); the largest single confusion found in the pipeline. Unlike the miscounting/mischaracterization confusion (§3.2), no simple rule could reliably separate the two here. Both categories often involve naming or identifying something in the image, so the line between “made up entirely” and “identified incorrectly” is genuinely hard to draw.

Boundary and Label Mismatches: Cor compares predicted and gold hallucination probabilities character by character across the response, an exact span match is not required for credit, but boundary mismatches reduce the score proportionally. Cor+Lbl uses the same character-by-character comparison but further penalizes a correctly-located span assigned the wrong category, scoring it as a complete miss on both categories. In a controlled test on 15 known-missed spans, the model flagged content in the correct sentence 100% of the time but matched gold’s exact boundaries only 13.3% of the time (Appendix B.5).

6 Conclusion and Future Work

We present a system for detecting hallucinated spans in vision-language model outputs, using a zero-shot LLM verifier, ranking 13th by Cor+Lbl (0.3015), through two independently validated interventions (numeral relabeling; Sonnet verification). An inexpensive, zero-shot VLM judge is a reasonable starting point, but the underperforming baseline shows it is not enough alone; targeted, category-specific fixes produced only a modest improvement, limited by detection failure, the invention/mischaracterization boundary overlap, and boundary and label mismatches on otherwise correctly-located spans. Among the methods we tested, no single mechanism, prompt calibration, span-extraction heuristics, or second-opinion verification, works equally well across all error types; a robust system requires routing each error type to its own specialized fix rather than one general-purpose approach.

Future work could pursue category-specific improvements: retrieval-augmented verification (Huang et al., 2025) for invention’s factual or named claims (Appendix B.4), a higher-end commercial OCR engine for corroboration, and next-token-probability signals (Azachi et al., 2025) to flag low-confidence detections for fine-grained categories without a full second-model call. It would also be worth testing whether the category-agnostic recall-focused approach in Appendix B.2, which showed a promising gain on a small sample, matches or exceeds our category-specific pipeline at full scale. Restricting evaluation to high-agreement gold spans (annotator agreement $\geq 2/3$) would also help separate genuine category-specific difficulty from a label-noise ceiling.

7 Limitations

This section discusses known constraints and open risks in our system and evaluation.

Sample-Scoped Analysis. Our error analysis is based on a 286-sample subset of the train set, not the full 3,799 rows, due to the cost of manually auditing predictions against gold labels at scale. While this sample was stratified to ensure balanced category coverage, findings from it may not perfectly generalize to the full dataset. This gap may be larger than it looks: the train split we used for all of our error analysis and validation is entirely LVLM-generated, while about a third of the test split is human-written hallucination samples instead. We never checked our fixes against human-written data directly. Several supplementary checks used small samples ($n=10$ in §3.3, $n=25$ in Appendix B.2, and $n=15$ in Appendix B.5); these were used to decide whether to pursue or discard a candidate fix, not as statistically conclusive findings on their own.

Language-Scoped Findings. All error analysis and design decisions in this paper are based on the English subset only. Whether the same patterns hold, mischaracterization being hardest to detect, OCR having the highest annotator agreement, the specific category confusions observed, in French, Chinese, or Italian is untested, and error patterns tied to language-specific text or naming conventions may not generalize.

Zero Predictions. Out of 1,201 test rows, 24 (2.0%) received an empty default prediction and scored as a full miss. Among these 24 rows, 23 were affected by four images exceeding the API’s 10MB size limit, and one additional row (test-en-20348) had a missing image file, with the same effect.

Non-Deterministic Outputs. Both models were run at temperature 1.0 (the API default); Sonnet 5 does not support a non-default temperature parameter. In small-scale rerun tests on identical inputs, Haiku produced a substantively different flag set in 2 of 5 calls and Sonnet in 2 of 4 calls, a relabeled or added/dropped flag, not just reworded reasoning. Re-running the full pipeline from scratch would likely yield similar, but not identical, results to those submitted. Because of this, the two Sonnet-verification stages in Table 1 (mischaracterization fix: +0.0111 Cor+Lbl; miscounting fix: +0.0007

Cor+Lbl) are each based on asking Sonnet only once per flagged span. We don't know how much these numbers would change if we asked Sonnet again on the same spans, and we did not test this directly, so the smaller miscounting gain in particular should be read with some caution.

Structural Ceiling on Verification. Our verifier is itself a VLM, so it may share the same visual limitations as the VLMs that generated the hallucinations it is checking. As an informal check, we separately asked four VLMs (Gemini, ChatGPT (GPT-5.5 Instant), Claude Haiku 4.5, Claude Sonnet 5) to count the flags in one dataset image, outside our evaluation pipeline; all four miscounted, despite the original dataset response and gold label correctly identifying the count. This illustrates that visual counting is unreliable across models generally, not just the model that produced a given response. Non-VLM tools, such as a dedicated OCR engine (e.g., EasyOCR), are not subject to this shared ceiling, since they are built for one specific task rather than general vision-language understanding.

Acknowledgments

All analyses, findings, and system design decisions in this paper are the author's own. Claude Code and Gemini were used to assist with writing and debugging the system's implementation code. Claude code was also used to run and verify the numerical claims reported in this paper against the underlying code and data, and to check drafted text for accuracy against those verified numbers, fix grammar and wording, and help with L^AT_EX formatting.

References

- Bradley Allen, Fina Polat, and Paul Groth. 2024. [SHROOM-INDElab at SemEval-2024 task 6: Zero- and few-shot LLM-based classification for hallucination detection](#). In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 839–844, Mexico City, Mexico. Association for Computational Linguistics.
- Ofir Azachi, Kfir Eliyahu, Eyal El Ani, Rom Himelstein, Roi Reichart, Yuval Pinter, and Nitay Calderon. 2025. [Leveraging NTPs for efficient hallucination detection in VLMs](#). In *Proceedings of the 1st Workshop on Confabulation, Hallucinations and Overgeneration in Multilingual and Practical Settings (CHOMPS 2025)*, pages 35–48, Mumbai, India. Association for Computational Linguistics.
- Baraa Hikal, Ahmed Nasreldin, and Ali Hamdi. 2025. [MSA at SemEval-2025 task 3: High quality weak labeling and LLM ensemble verification for multilingual hallucination detection](#). In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 989–995, Vienna, Austria. Association for Computational Linguistics.
- Sicong Huang, Jincheng He, Shiyuan Huang, Karthik Raja Anandan, Arkajyoti Chakraborty, and Ian Lane. 2025. [UCSC at SemEval-2025 task 3: Context, models and prompt optimization for automated hallucination detection in LLM output](#). In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 1981–1992, Vienna, Austria. Association for Computational Linguistics.
- Hanchao Liu, Wenyuan Xue, Yifei Chen, Dapeng Chen, Xiutian Zhao, Ke Wang, Liping Hou, Rongjun Li, and Wei Peng. 2024. A survey on hallucination in large vision-language models. *arXiv preprint arXiv:2402.00253*.
- Timothee Mickus, Claudio Savelli, Eduardo Calò, Emilio Raimond, Stella Frank, Hengyu Luo, Flavio Giobergia, Vincent Segonne, Chuyuan Li, Aman Sinha, Lorenzo Vaiani, Jörg Tiedemann, and Raúl Vázquez. 2026. [Can humans dream of electric sheep? human-written samples for fine-grained vision-and-language hallucination benchmarking](#). *Preprint*, arXiv:2608.01021.
- Timothee Mickus, Elaine Zosa, Raúl Vázquez, Teemu Vahtola, Jörg Tiedemann, Vincent Segonne, Alessandro Raganato, and Marianna Apidianaki. 2024. SemEval-2024 task 6: SHROOM, a shared-task on hallucinations and related observable over-generation mistakes. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, Mexico City, Mexico. Association for Computational Linguistics.
- Aman Sinha, Federica Gamba, Raúl Vázquez, Timothee Mickus, Ahana Chattopadhyay, Laura Zanella, Binesh Arakkal Remesh, Yash Kankanampati, Aryan Chandramania, and Rohit Agarwal. 2025. [SHROOM-CAP: Shared task on hallucinations and related observable overgeneration mistakes in crosslingual analyses of publications](#). In *Proceedings of the 1st Workshop on Confabulation, Hallucinations and Overgeneration in Multilingual and Practical Settings (CHOMPS 2025)*, Mumbai, India. Association for Computational Linguistics.
- Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. 2024. Aligning large multimodal models with factually augmented RLHF. In *Findings of the Association for Computational Linguistics: ACL 2024*, Bangkok, Thailand. Association for Computational Linguistics.
- Raul Vazquez, Timothee Mickus, Elaine Zosa, Teemu Vahtola, Jörg Tiedemann, Aman Sinha, Vincent

Segonne, Fernando Sanchez Vega, Alessandro Raganato, Jindřich Libovický, Jussi Karlgren, Shaoxiong Ji, Jindřich Helcl, Liane Guillou, Ona De Gibert, Jaione Bengoetxea, Joseph Attieh, and Marianna Apidianaki. 2025. [SemEval-2025 task 3: MU-SHROOM, the multilingual shared-task on hallucinations and related observable overgeneration mistakes](#). In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 2472–2497, Vienna, Austria. Association for Computational Linguistics.

Raúl Vázquez, Aman Sinha, Chuyuan Li, Artem Shelmanov, Artem Vazhentsev, Claudio Savelli, Eduardo Calò, Emilio Raimond, Stella Frank, Hengyu Luo, Flavio Giobergia, Vincent Segonne, Lorenzo Viani, Jörg Tiedemann, and Timothee Mickus. 2026. [Overview of shroom-vision 2026: A shared task on hallucination detection in large vision-language models](#). *Preprint*, arXiv:2608.25662.

A Prompt Versions

The prompt was developed in three iterations. The final version is reproduced in full below; earlier versions differed only as noted in §A.2.

A.1 Final Prompt

Role.

You are a precise hallucination span detector analyzing outputs from vision-language models.

You will be given: - An image
- A prompt that was asked about the image - A response that a vision-language model generated

Your job is to find specific words or phrases in the response that do not precisely match what is visible in the image.

IMPORTANT INSTRUCTIONS: - Even if the response is broadly correct, individual words or phrases may be inaccurate. Find those specific words. - Do not evaluate the response as a whole. - Evaluate each factual claim individually against the image. - A response can be mostly right but contain specific wrong details --- your job is to find those details. - Pay attention to: specific names, varieties, brands, colors, quantities, spatial relationships, and visible text. - If the response restates the same incorrect claim in different words across multiple sentences, flag EACH restatement as its own separate entry -- do not stop after the first one.

Categories.

There are five hallucination categories:

INVENTION: The response mentions something that does not exist in the image at all, OR states a fact, explanation, or piece of terminology that cannot be verified from the image and is not something a careful observer could determine just by looking. This includes invented locations, objects, invented technical/scientific explanations, and invented names for things.

Example (spatial): Claiming the image shows Auckland New Zealand when no location markers are visible.

Example (fabricated explanation): Claiming a bird's unusual white coloring is "due to a genetic condition called leucism" when nothing about genetics can be determined from a photo --- this is a confidently-stated but unverifiable claim, which counts as invention even though it doesn't contradict anything visible.

MISCHARACTERIZATION: Something real in the image is described incorrectly. The object exists but a property is wrong --- color, material, direction, name.

Example: Saying cables are attached to the ceiling when they are attached to the wall.

MISCOUNTING: The response states a wrong quantity of something visible in the image.

Example: Saying "four legs" when the animal has three.

OCR: The response misreads text that is visibly written in the image.

Example: Reading a slide as "sign on" when it says "log in".

OTHERS: Any other hallucination that does not fit the above categories. This includes cases where the response fails to actually answer the prompt --- for example, if the response simply repeats or echoes the question back (e.g., "Human: what is X?") instead of providing a real answer. In such cases, flag the entire non-answering response as "other" with high confidence.

Span extraction rules.

RULE 1 - MISCOUNTING: You MUST extract ONLY the exact number word or numeral. Do NOT include the noun being counted. * CORRECT: "four" * INCORRECT: "four legs" * CORRECT: "2" * INCORRECT: "2 front legs"

RULE 2 - MISCHARACTERIZATION: Extract ONLY the specific word or short property phrase that is false. Do NOT return full clauses or explanations. * CORRECT: "ceiling" or "the ceiling" (if it's attached to a wall) * INCORRECT: "cables attached to the ceiling"

RULE 3 - OCR: Extract ONLY the misread text exactly as it appears in the RESPONSE (the WRONG reading) -- not the correct text from the image, and not surrounding correct words. * Example: if the response says "sign on" but the image actually reads "log in", the span.text is "sign on".

CRITICAL MANDATE: The "span.text" must be the shortest possible verbatim substring from the RESPONSE. If you include extra correct words surrounding the error, the evaluation system will penalize it with a 0 score.

OCR guidance.

If the prompt asks about text visible in the image: - Read the text in the image very carefully before evaluating the response - Compare word by word against what the response claims - Flag any word that differs from what the image shows - If the image text is too blurry to read confidently, do not flag OCR errors --- assign low confidence only

Confidence guidance.

For each flagged span assign a confidence score between 0 and 1:

0.8 - 1.0: You are certain this is wrong. The image clearly contradicts this claim. Any reasonable person looking at the image would agree.

0.5 - 0.7: You are fairly confident this is wrong but there is some ambiguity. The image suggests it is wrong but interpretation is needed.

0.2 - 0.4: You think this might be wrong but you are not sure. The claim is questionable but

the image does not clearly contradict it.

If you are genuinely unsure whether something is wrong, assign low confidence rather than not flagging it. If nothing is clearly wrong, return an empty list. Be conservative --- do not flag things unless you have clear visual evidence they are wrong. If you are not completely certain something is wrong based on clear visual evidence, do NOT flag it. Prefer returning empty [] over uncertain flags.

Output format.

Return your answer as a JSON array only. No other markdown formatting, no conversational text.

Before writing each "span.text", verify: Is this the absolute minimum string required to isolate the error? For numbers, is it just the digit/number word?

CRITICAL: Your entire response must be ONLY the JSON array -- nothing before it, nothing after it. Do not think out loud, do not write "wait, let me reconsider" or any re-examination in your output. If you are uncertain whether something should be flagged, resolve that uncertainty silently before writing the JSON --- only include a flag in the JSON if you are confident about it at the moment you write it. Never include a flag you go on to contradict or retract afterward.

Each entry must have: - "span.text": the shortest exact substring from the response - "label": one of invention, mischaracterization, miscounting, OCR, other - "prob": confidence score between 0 and 1 - "reason": one sentence explanation

Input. The final prompt is completed with the filename hint, question prompt, and candidate response appended verbatim.

A.2 Changes Across Iterations

- **Version 1 (initial):** Used a narrower INVENTION definition (facts absent from the image only, with no clause covering unverifiable-but-confidently-stated claims, and no worked

example for that case). The OTHERS category had no non-answer rule. Span extraction rules covered only miscounting and mischaracterization (no Rule 3 for OCR). The output format had no instruction against reasoning aloud in the response.

- **Version 2:** Expanded the INVENTION definition to include unverifiable but confidently-stated claims, with a worked example (a fabricated genetic explanation for a bird’s coloring). Added the non-answer rule under OTHERS, instructing the model to label non-answering responses (e.g., echoing the question back) as “other” with high confidence. Added an instruction to the output format discouraging the model from reasoning aloud or retracting flags mid-output.
- **Version 3 / final:** Added Rule 3 to the span extraction rules, specifying how to isolate misread OCR text from the response.

B Additional Experimental Detail

B.1 Rejected Post-Processing Filters

Filter	Dev	Held-out
Confidence threshold (mischar $\geq$ 0.92)	+11.0%	−13.7%
Hedge-language suppression	+6.5%	−4.6%
Negation-sentence suppression	+1.8%	−2.5%

Table 2: Cor+Lbl change on development vs. held-out data for three rejected heuristic filters. All three showed the same reversal pattern.

Before the numeral and Sonnet-based fixes, we tried three simpler filters aimed at the same noisy mischaracterization flags: a confidence threshold that drops flags below 0.92, a rule that drops flags whose stated reason uses hedging words like “appears to” or “might be,” and a rule that drops flags in a sentence containing a negation word. Each filter was first tested on the development set, then checked again on a separate held-out set it had never seen. All three looked like real improvements on the development set, and all three got worse on the held-out set (Table 2). This showed that Haiku’s confidence scores and wording choices don’t reliably separate correct flags

from wrong ones across different batches of data, so none of the three were used in the final system.

We also tried a narrow negation-parsing rule (a bare noun right after “not”/“no” treated as an unfair claim) and a directional rule for relabeling OCR/invention confusions, based on an apparent 18-vs-6 imbalance. Neither made it to a scoring run: a search of 325 samples found zero cases matching the negation pattern (the two examples that motivated it turned out to be the same image sampled twice), and a closer look at the OCR/invention imbalance found 11 of the 18 cases came from one duplicated row, leaving nothing real to fix.

As an alternative to dropping Sonnet-rejected spans outright, we tried scaling a rejected span’s probability down instead of removing it. This did worse than the deployed hard-removal approach on both metrics (−0.0059 Cor, −0.0029 Cor+Lbl): Spearman correlation rewards long runs of correctly-matched zeros, and most rejections are correct, so the occasional wrong rejection isn’t worth the tradeoff.

B.2 Recall-Targeted Verification

Since missed detections were the biggest source of error across categories, we tried a second, independent Sonnet pass: asking Sonnet to look for spans Haiku missed, without showing it Haiku’s own flags, then combining both sets of results.

Sonnet checked each image and response fresh, on its own, looking for anything Haiku hadn’t already flagged. We added its findings to Haiku’s original flags rather than using them to filter or replace anything, since the goal here was catching misses, not fixing wrong labels.

On a natural 25-sample test, this gained +0.0215 Cor+Lbl (+0.0646 Cor) over Haiku alone, catching 53.7% of the spans Haiku had missed (54 spans in total). It came at a real but manageable cost: 28.3% of the newly-added spans turned out to be false positives. A realistic estimate for the full deployment is likely lower than this small-sample result, probably somewhere around the confirmed mischaracterization result (+0.0111 Cor+Lbl) or a bit higher, since small-sample estimates in this project consistently overstated the real gains once deployed. Running this across all 1,201 test rows would have cost about \$16.70 (or \$8.35 with the Anthropic Batch API), which was more than we had left in the budget, so we didn’t do it.

B.3 OCR Corroboration (EasyOCR)

We also tried EasyOCR, a standalone OCR tool, as a way to double-check OCR flags, on a 105-sample manual breakdown. It only confirmed 22.4% of true positives, and couldn't read any text at all in 17.1% of images. On the other hand, when it found nothing, that lined up with a real false positive 60.0% of the time, and it would have caught 38.5% of the OCR errors Haiku missed entirely. So it was good at spotting false positives, bad at confirming real ones, and prone to failing outright, which made it too unreliable to use as a general filter. We didn't add it to the pipeline.

B.4 RAG Feasibility Estimate

We manually sorted 154 gold invention spans into five types: 46.1% were purely visual claims, 23.4% were non-coherent annotation fragments, 14.9% were text-verifiable facts, 11.7% were named-entity identifications, and 3.9% were speculative. The text-verifiable and named-entity spans together (26.6%) are the kind of claim that could plausibly be checked against an outside source, without needing the image at all. The rest are either purely visual judgments or annotation artifacts that no amount of retrieval could fix.

B.5 Recall-Gap Mechanism Test

To test whether missed detections reflect the model losing track of content in long responses, we re-audited 15 known-missed gold spans by feeding the model only the half of the response containing the target, as if that were the whole response. The model matched gold's exact span boundaries in only 2 of 15 cases (13.3%), but flagged something in the same sentence as the target in 15 of 15 cases (100%), both in this isolated-half rerun and in cached full-length runs of a subset of the same cases. For example, where gold marked only "long," the model flagged the wider "long barrel." This indicates the model was not truly missing this content; it consistently located the right area, but disagreed with gold on how wide the span should be. This sample is small ($n=15$) and cannot distinguish cases where the model's own span was too wide from cases where gold's span was unusually narrow; the finding should be read as suggestive rather than conclusive.