Perceptual Hallucination in Vision–Language Models: Definition, Analysis and Verification

Taewook Hwang[1]* **Inbum Heo**[1]* **Sung Jun Lee**[1] **Sangkeun Jung**[1,2]†
[1]Department of Computer Science & Engineering, Chungnam National University
[2]EurekaAI
✉: {taewook5295, inbum10222, lsungj0920, hugmanskj}@gmail.com

Abstract

Vision-Language Models (VLMs) have demonstrated remarkable performance in document understanding tasks; however, VLMs also suffer from hallucinations inherited from LLMs. While prior work has focused on reasoning-stage hallucinations, the role of visual perception remains underexplored. In this work, we define **perceptual hallucination** as the phenomenon where VLMs generate information as if perceived, despite absent or damaged visual evidence. To analyze this, we construct **DocHallu**, a benchmark of 2,671 original–damaged image pairs across three tasks, available at `https://huggingface.co/datasets/IB99/DocHallu`. Experiments reveal that perceptual hallucination occurs across all models, with higher rates for numerical content than textual content. Activation patching analysis suggests that hallucinations are strongly associated with errors introduced in the vision encoder, which can subsequently propagate and become amplified through the text decoding process. We also demonstrate that LLM-based post-hoc filtering can reduce hallucination exposure by 36% on average, with reductions of up to 88%. This work extends VLM hallucination research by defining, analyzing, and verifying perceptual hallucination in document understanding.

1 Introduction

Recent advances in large language models (LLMs) have dramatically improved text understanding and generation capabilities. Vision-Language Models (VLMs), which combine these advances with visual information, have achieved strong performance across document understanding, visual question answering, and multimodal reasoning tasks (Zhang et al., 2023), emerging as powerful alternatives to traditional optical character recognition (OCR)-based pipelines (Kim et al., 2021).

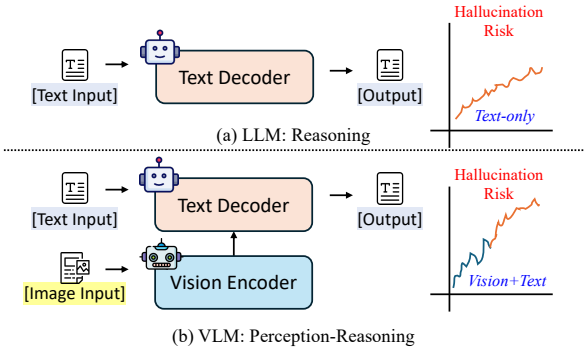


Figure 1: In VLMs, uncertainty introduced during visual perception can propagate into the reasoning process, resulting in divergent inference and hallucinated outputs despite identical reasoning mechanisms.

However, VLMs inherit the *hallucination* problem inherent to LLMs (Bai et al., 2024), generating outputs that are unfaithful to the input or factually incorrect. This poses **serious risks in real-world applications** where reliability is critical. While prior hallucination research has primarily focused on reasoning errors in text-only LLMs, VLMs adopt a more complex architecture that combines visual perception with linguistic reasoning (Radford et al., 2021). As a result, hallucinations in VLMs differ fundamentally from those in LLMs. However, most prior studies analyze them in much the same way, focusing solely on text generation without isolating the role of visual perception (Gunjal et al., 2024; Wang et al., 2023).

As illustrated in Figure 1, VLMs introduce an additional source of uncertainty at the visual perception stage, which can propagate into the reasoning process and result in divergent outputs.

This problem is particularly critical in tasks requiring precise visual recognition, such as mathematical information extraction (Anitei et al., 2021) or document understanding involving calculations (Huang et al., 2022). Misrecognition of tables, layouts, numbers, or symbols can lead to plausible but incorrect responses. Despite this, the

*Equal contribution.
†Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 24710–24725

importance of perception-stage hallucinations has been relatively overlooked (Tonmoy et al., 2024).

In this work, we define **perceptual hallucination** as the phenomenon where VLMs generate information as if perceived from the image, despite the visual evidence being absent or damaged.

To analyze this, we construct **DocHallu**, a benchmark covering three document understanding tasks: Mathematical Expression Recognition (MER), Key Information Extraction (KIE), and Document VQA (DVQA). Each sample consists of an original–damaged image pair, where the damaged image has the answer-relevant visual information removed, creating conditions that can induce hallucination under controlled settings.

We design experiments around the following research questions:

- **RQ1.** Is **perceptual hallucination** observable in document-based VLM tasks?
- **RQ2.** How do the **vision encoder** and **text decoder** contribute to perceptual hallucination in document understanding?
- **RQ3.** Can LLMs be used to automatically detect perceptual hallucination?

Our main contributions are as follows: (1) We construct **DocHallu**, a document-based original–damaged benchmark for systematically analyzing perceptual hallucination in VLMs. (2) We **quantitatively analyze** how visual information loss affects hallucination and examine the relative contributions of the vision encoder and text decoder. (3) We evaluate **LLM-based post-hoc hallucination verification**, demonstrating practical mitigation potential in document understanding.

2 Related Work

2.1 Hallucination in LLMs and VLMs

Hallucination refers to the generation of plausible but unfounded content and has been extensively studied in LLMs from the perspectives of factuality and faithfulness (Rawte et al., 2023). Recent surveys systematically categorize hallucination types, causes (e.g., data bias, decoding), and mitigation strategies (Huang et al., 2025; Tonmoy et al., 2024).

In VLMs, hallucination often manifests as modality unfaithfulness to visual inputs. For natural image VQA, benchmarks and evaluation protocols have been proposed to quantify the generation of non-existent objects, attributes, or text (Li et al., 2023; Sun et al., 2024). Decoding-based approaches that improve visual faithfulness without additional training have also been explored (Park et al., 2025), demonstrating that hallucination can be mitigated through reasoning-stage control alone.

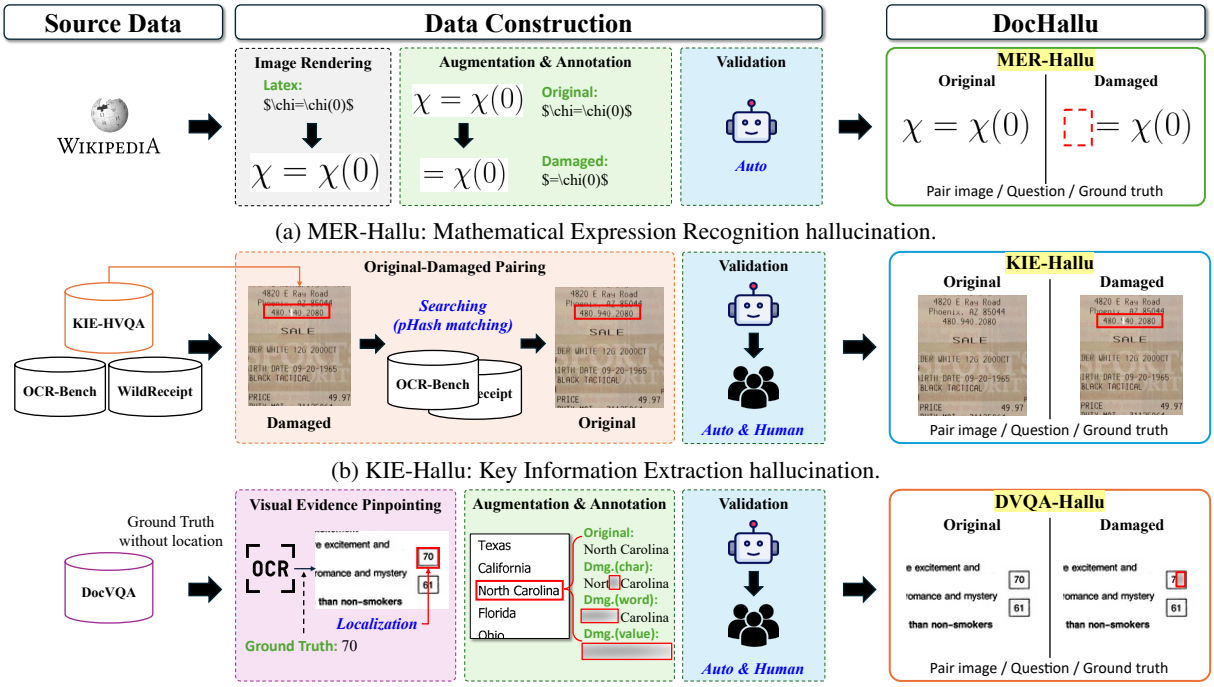
Recent work has proposed to decompose VLM hallucination by cause (Liu et al., 2023): *perceptual hallucination* involves generating text, symbols, or attributes as if perceived despite their absence, while *cognitive hallucination* arises from reasoning errors such as incorrect combinations or commonsense leaps (Liu et al., 2024a). However, existing VLM hallucination research has primarily focused on natural images, leaving hallucination in document inputs—where precise character and number recognition is critical—relatively underexplored. In document settings, hallucination cannot be fully explained by traditional OCR errors alone, as perceptual hallucination in VLMs arises from implicit visual representations learned end-to-end. In this context, our study focuses on measuring perceptual hallucination in document understanding under controlled conditions.

2.2 Document Understanding and Hallucination

Document understanding requires precise recognition of localized, high-resolution visual cues, unlike natural image understanding. Representative tasks include DVQA (Mathew et al., 2021), KIE (Jaume et al., 2019), and MER (Gervais et al., 2025). These tasks share a critical characteristic: performance depends on exact value/entity restoration rather than semantically approximate generation. Consequently, document hallucination poses substantial risk, potentially generating non-existent values such as amounts, dates, or identifiers.

Despite these concerns, hallucination research in document domains remains limited, particularly with respect to controlled analyses of visual information loss. Although recent work on KIE-HVQA has studied hallucination in degraded documents (He et al., 2025), it focuses primarily on KIE tasks. Consequently, unified experimental frameworks that construct original–damaged pairs across diverse document tasks are still lacking.

To address this gap, we propose **DocHallu**, encompassing MER-Hallu, KIE-Hallu, and DVQA-Hallu. Using original–damaged pairs, we measure hallucination under controlled conditions and analyze how document-specific factors, such as corruption granularity and answer type, influence hallucination patterns and how such effects are distributed across vision and language components.



(a) MER-Hallu: Mathematical Expression Recognition hallucination.

(b) KIE-Hallu: Key Information Extraction hallucination.

(c) DVQA-Hallu: Document Visual Question Answering hallucination.

Figure 2: Overview of the DocHallu dataset construction pipeline for three document understanding tasks. Damaged images are generated by removing visual evidence corresponding to the ground truth.

3 DocHallu: A Dataset for Perceptual Hallucination

3.1 Definition of Perceptual Hallucination

VLMs encode input images through a vision encoder and pass the resulting representations to a language decoder for output generation. The vision encoder recognizes visual elements such as text, objects, and layouts within the image. We define **perceptual hallucination** as the phenomenon in which errors in the *visual perception stage* cause the model to generate predictions based on non-existent or distorted visual information.

Conventional hallucination research has primarily focused on factual inconsistencies or reasoning errors during language generation. In contrast, perceptual hallucination concerns inaccuracies introduced earlier, during visual information interpretation. Such hallucinations are particularly difficult to detect, as they can be **amplified** through subsequent language reasoning, producing outputs that appear coherent despite lacking visual evidence.

3.2 Document Hallucination Analysis Tasks

As illustrated in Figure 3, these tasks span a spectrum from perception-dominant to perception–reasoning stages and are well suited for analyzing perceptual hallucination, since minor visual corruption can directly alter model predictions.

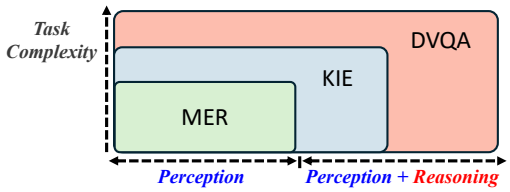


Figure 3: Document understanding tasks positioned by perception–reasoning dependency and task complexity.

We construct **damaged images** from **original images** by selectively corrupting answer-relevant visual information and compare VLM predictions under identical textual conditions. Observed output differences can therefore be attributed to information loss in the visual perception stage rather than reasoning variability.

3.3 Dataset Construction

DocHallu comprises three document understanding datasets constructed as original–damaged image pairs to measure perceptual hallucination.

MER-Hallu targets pure visual perception via verbatim transcription of mathematical expressions. **KIE-Hallu** focuses on extracting structured fields (e.g., phone numbers and dates) from documents. **DVQA-Hallu** addresses document visual question answering involving information localization and interpretation. Figure 2 illustrates the dataset construction pipeline, and Table 1 summarizes the dataset composition.

3.3.1 MER-Hallu

MER-Hallu is a synthetic dataset designed to analyze perceptual hallucination in mathematical expression recognition. We collect LaTeX expressions from Wikipedia and render 1,469 original–damaged image pairs.

Damaged images are generated by selectively removing portions from either end of the original expressions. Rendering errors and non-standardized expressions are filtered through automated validation (Figure 2a).

3.3.2 KIE-Hallu

KIE-Hallu is constructed to evaluate perceptual hallucination in real-world document scenarios, including receipts and identification cards. It is derived from KIE-HVQA, a document-oriented dataset used in prior VLM hallucination studies.

As illustrated in Figure 2b, KIE-HVQA contains only damaged document images. To construct original–damaged pairs, we retrieve the corresponding original images from the source datasets, including WildReceipt[1] and OCRBench (Liu et al., 2024b; Fu et al., 2024). We apply quality filtering to ensure data validity and proper alignment between the original and damaged images.

3.3.3 DVQA-Hallu

DVQA-Hallu is constructed to evaluate perceptual hallucination in document visual question answering. We select table- and list-type questions from the DocVQA benchmark that primarily require accurate visual recognition rather than complex reasoning (Figure 2c).

From an initial pool of 1,504 questions, we filter out question–answer mismatches and samples for which answer-relevant regions cannot be reliably identified due to low image quality, resulting in 919 final question–image pairs.

To pinpoint the answer-relevant visual evidence, we first localize the corresponding regions using a commercial OCR model[2] and selectively corrupt the localized regions. To analyze hallucination patterns under varying degrees of visual information loss, we apply three corruption granularities: character-level, word-level, and value-level.

For each task, detailed data structures and representative samples are provided in Appendix A.

[1] https://github.com/Ikomia-hub/dataset_wildreceipt

[2] https://www.upstage.ai/

Dataset	Granularity	Source	Pairs
MER-Hallu	Char	–	1,469
KIE-Hallu	Char	429	283
DVQA-Hallu	Char	1,504	318
DVQA-Hallu	Word	1,504	276
DVQA-Hallu	Value	1,504	325
Total			**2,671**

Table 1: DocHallu dataset statistics. **Source**: original data; **Pairs**: verified original–damaged image pairs. DVQA-Hallu totals 919 pairs across three corruption granularities (318 Char + 276 Word + 325 Value).

4 Experimental Setup

Using the DocHallu benchmark, we design experiments around three research questions to evaluate, analyze, and detect perceptual hallucination in VLMs:

- **RQ1.** Is **perceptual hallucination** observable in document-based VLM tasks?
- **RQ2.** How do the **vision encoder** and **text decoder** contribute to perceptual hallucination in document understanding?
- **RQ3.** Can LLMs be used to automatically detect perceptual hallucination?

4.1 Models

We evaluate both closed-source and open-source VLMs. The closed-source models include GPT-5.2 (OpenAI, 2025), Claude Sonnet 4.5 (Anthropic, 2025), and Gemini 2.5-Flash (DeepMind, 2025). The open-source models include Gemma3-4B (Google DeepMind, 2025) and Qwen3-VL-4B (Alibaba Group, 2025). This selection allows us to compare hallucination patterns across models with different architectures and training paradigms.

4.2 Inference Settings

To isolate perceptual hallucination originating from the vision component, we set the temperature to 0 for all experiments. This eliminates stochastic variation during inference and ensures reproducible outputs for identical inputs. GPT-5.2 and Claude Sonnet 4.5 are accessed via the OpenAI and Anthropic APIs, respectively. Gemini 2.5-Flash is accessed through the OpenRouter API[3]. Gemma3-4B and Qwen3-VL-4B are executed locally using the HuggingFace framework.

[3] https://openrouter.ai/

Task (Hallu. Rate ↓)	GPT-5.2		Claude Sonnet 4.5		Gemini 2.5-Flash		Gemma3-4B		Qwen3-VL-4B	
	Orig.	Dmg.	Orig.	Dmg.	Orig.	Dmg.	Orig.	Dmg.	Orig.	Dmg.
MER-Hallu	0.022	0.120	0.014	0.086	0.005	0.147	0.369	0.563	0.090	0.230
KIE-Hallu	–	0.496	–	0.444	–	0.438	–	0.315	–	0.459
DVQA-Hallu	–	0.594	–	0.571	–	0.472	–	0.719	–	0.708

Table 2: **Hallucination rate** on original (Orig.) and damaged (Dmg.) inputs. Shaded columns correspond to Dmg. inputs, which are the primary focus of comparison. For KIE-Hallu and DVQA-Hallu, hallucination is measured only under Dmg. where the answer region is removed.

4.3 Evaluation Protocol

Evaluating hallucination requires assessing the consistency between model predictions and the visual evidence provided in the input. Since hallucination is an inherent limitation of LLMs, we rely on **human evaluation** rather than LLM-based automatic assessment. We employ 9 annotators, with 3 annotators assigned per task, and determine the final label by unanimous agreement. Annotators are presented with the original image, the damaged image, the model output, and the ground truth, and are asked to judge whether hallucination occurs.

A sample is labeled as hallucinated if it exhibits any of the following behaviors: (i) generating visually unverifiable symbols, numbers, or text from the damaged image; (ii) reconstructing complete values from only partial visual evidence; or (iii) combining values from different document regions to produce a plausible answer. Based on this definition, we compute the hallucination rate (Hallu. Rate) as the proportion of samples judged as hallucinated, as formalized in Eq. 1:

$$\text{Hallu. Rate} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\left(\sum_{k=1}^{3} y_i^{(k)} = 3\right) \quad (1)$$

$\mathbb{I}(\cdot)$ denotes the indicator function, which equals 1 if the condition is satisfied and 0 otherwise.

MER-Hallu involves minimal reasoning beyond visual perception; thus, we report hallucination rates under both original (Orig.) and damaged (Dmg.) conditions. In contrast, KIE-Hallu and DVQA-Hallu require contextual reasoning, making it difficult to disentangle perceptual errors from reasoning failures under original inputs. Accordingly, for these tasks, we measure hallucination only on Dmg. outputs, where answer-relevant visual evidence is removed.

5 Experiments and Results

5.1 RQ1: Do VLMs Hallucinate in Documents?

RQ1 aims to verify whether VLMs actually exhibit **perceptual hallucination**, defined as generating information that is visually unobservable.

5.1.1 Datasets and Models

We use all three datasets from the DocHallu benchmark and evaluate all five models.

5.1.2 Results and Analysis

As shown in Table 2, all models exhibit substantial hallucination under damaged image conditions. In MER-Hallu, minor hallucination is observed even on original images (0.014–0.369), while hallucination rates increase markedly when images are damaged. For KIE-Hallu and DVQA-Hallu, hallucination is measured only on damaged images where the answer-relevant visual evidence is removed. Despite the absence of visual evidence, models frequently generate answers (KIE-Hallu: 0.315–0.496; DVQA-Hallu: 0.472–0.719).

These results suggest that, when visual evidence is incomplete, models tend to rely on internal knowledge or linguistic priors, thereby inducing hallucination.

Appendix B provides additional results on general model performance based on human evaluation and quantitative metrics, along with the prompts used for evaluation.

Hallucination Rate by Answer Type As illustrated in Figure 4, numerical answers exhibit substantially higher hallucination rates than textual answers in DVQA-Hallu and KIE-Hallu. Specifically, hallucination rates reach 84.8% in DVQA-Hallu and 61.0% in KIE-Hallu for numerical content. In contrast, MER-Hallu shows comparable hallucination rates for text and numbers, indicating a smaller gap between information types in pure perception tasks.

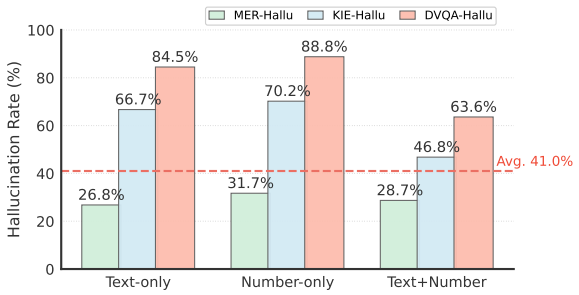


Figure 4: Hallucination rates by answer type across MER, KIE, and DVQA.

These results suggest that in reasoning-involved tasks, models tend to *plausibly complete* numerical information rather than *accurately read* it.

Hallucination Rate by Corruption Granularity As shown in Figure 5, character-level corruption consistently yields the highest hallucination rates in DVQA-Hallu. On average, character-level corruption results in 42.5% hallucination, followed by value-level (39.9%) and word-level (17.6%).

This pattern indicates that partial visual corruption encourages over-completion from limited evidence, increasing hallucination.

5.1.3 Human Evaluation Reliability

To verify the reliability of human evaluation, we measured inter-annotator agreement among 3 annotators using Fleiss' κ (Fleiss, 1971). All tasks showed κ values ranging from 0.55 to 0.74, indicating moderate to substantial agreement.

5.2 RQ2: How Is Hallucination Related to Vision and Language Components?

Having established the prevalence of perceptual hallucination (RQ1), we examine which internal components of VLMs contribute to perceptual hallucination in document understanding. To analyze how predictions change under visual information loss, we compare internal representations between original and damaged images. We employ activation patching (Meng et al., 2022) to examine how hidden representations at specific layers influence prediction outcomes.

5.2.1 Datasets and Models

We restrict our analysis to samples unanimously labeled as hallucinated by all annotators in RQ1, reducing uncertainty and enabling clearer analysis. Furthermore, because our analysis requires access to internal hidden states, we use only open-source VLMs, **Qwen3-VL-4B** and **Gemma3-4B**.

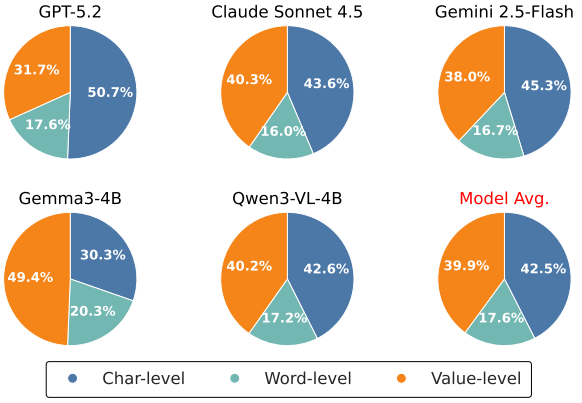


Figure 5: Hallucination rates by corruption granularity (per model and average).

5.2.2 Activation Patching

Activation patching assesses layer importance by injecting hidden representations from one input condition into another. During inference on damaged images, we replace hidden representations at specific layers with those from the corresponding original images and examine whether the model recovers the correct answer.

We perform layer-wise activation patching sequentially across all vision and text layers. For a given layer l, the hidden state from the damaged image is substituted with that from the original image, after which forward propagation proceeds through subsequent layers. If the correct answer is recovered, the patched layer is considered to be strongly associated with the hallucination observed under damaged conditions.

5.2.3 Results and Analysis

As shown in Figure 6, patching vision layers consistently leads to the recovery of correct answers across nearly all layer positions. This suggests that injecting correct visual representations at different stages of the vision encoder can enable downstream language processing to recover correct outputs.

In contrast, patching text layers exhibits different behaviors. Patching lower and middle text layers rarely restores correct answers, whereas patching from approximately the 10th layer from the top gradually recovers correct outputs. This suggests that lower text layers are more actively involved in reasoning transformations, while upper layers primarily refine or format outputs.

Influence of Vision Layers on Hallucination These results indicate that **vision layers are highly sensitive to hallucination** in document understanding tasks that require precise visual perception.

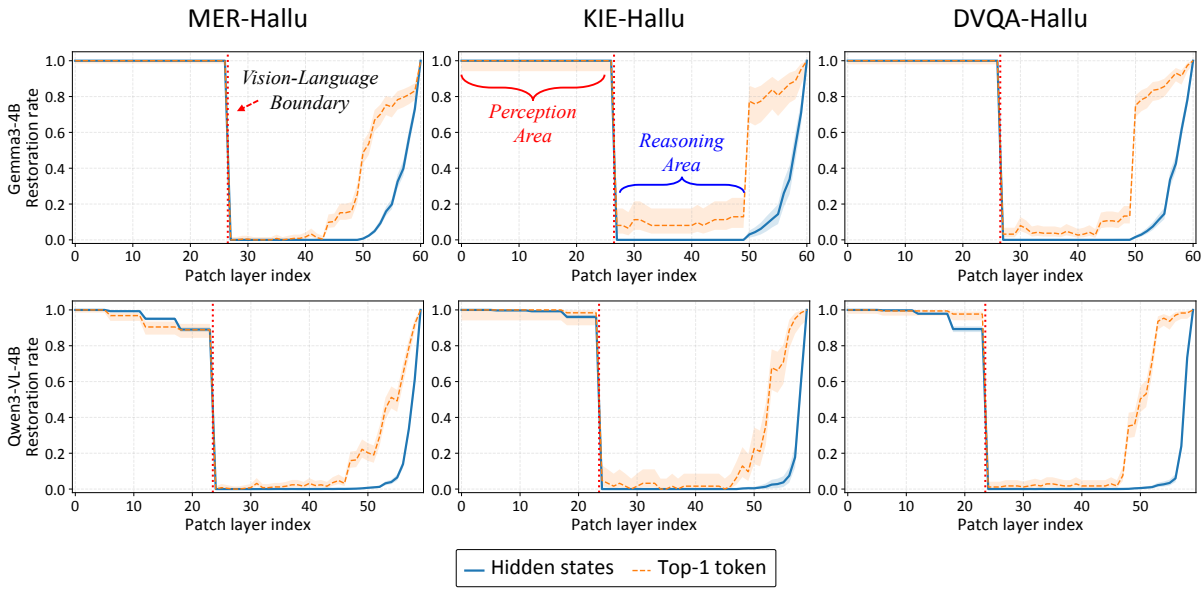


Figure 6: Activation patching results on hallucinated samples from MER-Hallu, KIE-Hallu, and DVQA-Hallu. We report both the recovery rate of the final-layer hidden states and the corresponding prediction recovery rate when patching individual layers. The red vertical line indicates the vision–language boundary.

Disruptions to visual representations at any vision layer can lead to prediction errors that are difficult to correct in subsequent language processing stages. Conversely, restoring correct visual representations enables stable downstream reasoning, underscoring the importance of accurate visual perception.

Influence of Text Layers on Hallucination For text layers, upper layers exhibit limited influence on hallucination, as they mainly operate on already-formed representations. Lower and middle text layers, which actively integrate visual information during reasoning, show greater sensitivity to representation instability. Disruptions in these layers can interfere with the integration of perception and reasoning, potentially contributing to hallucination.

Overall, our results indicate that VLM outputs are strongly associated with the visual perception stage, while hallucination related to language processing is more closely tied to lower text layers involved in active reasoning. Detailed experimental statistics are reported in Appendix C.

5.3 RQ3: Can LLM-based Hallucination Filtering Be Effective?

Given that hallucination primarily arises from the vision encoder (RQ2), we examine whether LLM-based verifiers can effectively reduce hallucination exposure when applied as post-hoc filters to VLM outputs on damaged images. We analyze the trade-off between hallucination reduction and response preservation by applying LLM-based verification to model outputs and filtering responses identified as hallucinated.

Datasets and Models Experiments are conducted on damaged images from the DocHallu benchmark. Because responses to damaged images are prone to hallucination due to missing visual evidence, post-hoc filtering represents a practical mitigation strategy. We use the same five VLMs from RQ1 as both predictors and verifiers, comparing *self-check* (same model) and *cross-check* (different model) configurations.

5.3.1 Metrics

Filtering performance is evaluated using two metrics for hallucination reduction and response coverage.

Hallucination Reduction Rate on Damaged Images (RRd)**.** Let H be the number of hallucinated samples and FN those remaining unfiltered. The hallucination reduction rate is defined as:

$$\mathrm{RRd} = 1 - \frac{\mathrm{FN}}{H}. \tag{2}$$

Higher RRd indicates more effective hallucination filtering.

Response Coverage (Cov)**.** Let N denote the total number of evaluated samples, and Filtered the number of responses blocked by the verifier. Response coverage is defined as:

$$\mathrm{Cov} = \frac{N - \mathrm{Filtered}}{N}, \tag{3}$$

which measures the proportion of responses preserved after filtering (Manakul et al., 2023).

5.3.2 Results and Analysis

Table 3 reports self-check results and representative cross-check verifier combinations that achieve the highest RRd for each task (see Appendix D for all combinations).

Self-check vs. Cross-check The results reveal a clear trade-off between hallucination reduction and response preservation. Self-check maintains relatively high response coverage (Cov $= 0.39$–0.97) but provides limited hallucination reduction (RRd $= 0.05$–0.60). In contrast, cross-check substantially improves hallucination reduction (up to RRd $= 0.88$), at the cost of lower coverage (Cov $= 0.23$–0.85).

Overall, self-check is suitable when response preservation is prioritized, whereas cross-check is preferable when minimizing hallucination exposure is critical. Accordingly, the proposed post-hoc filter should be interpreted as a mitigation mechanism rather than a complete prevention solution, with the configuration chosen based on application-specific safety and latency requirements.

Pred.	Verifier	Cov ↑	RRd ↑	Lat.(s)↓
MER-Hallu				
GPT-5.2	GPT-5.2	0.96	0.05	1.00
	Gemini 2.5-Flash	0.83	0.48 (+0.43)	2.28
Claude Sonnet 4.5	Claude Sonnet 4.5	0.91	0.12	3.03
	Gemini 2.5-Flash	0.85	0.54 (+0.43)	2.24
Gemini 2.5-Flash	Gemini 2.5-Flash	0.92	0.43	2.40
	GPT-5.2	0.82	0.61 (+0.17)	0.62
Gemma3-4B	Gemma3-4B	0.84	0.25	0.59
	Gemini 2.5-Flash	0.41	**0.88 (+0.63)**	2.32
Qwen3-VL-4B	Qwen3-VL-4B	0.81	0.25	0.46
	Gemini 2.5-Flash	0.74	0.67 (+0.42)	2.19
KIE-Hallu				
GPT-5.2	GPT-5.2	0.93	0.10	1.83
	Gemini 2.5-Flash	0.65	0.49 (+0.39)	2.74
Claude Sonnet 4.5	Claude Sonnet 4.5	0.90	0.10	2.80
	Gemini 2.5-Flash	0.50	**0.54 (+0.44)**	2.67
Gemini 2.5-Flash	Gemini 2.5-Flash	0.97	0.05	2.66
	Gemma3-4B	0.69	0.37 (+0.31)	0.87
Gemma3-4B	Gemma3-4B	0.73	0.21	1.07
	Gemini 2.5-Flash	0.51	0.48 (+0.27)	2.57
Qwen3-VL-4B	Qwen3-VL-4B	0.91	0.12	0.59
	Gemini 2.5-Flash	0.64	0.44 (+0.32)	2.66
DVQA-Hallu				
GPT-5.2	GPT-5.2	0.85	0.20	2.93
	Gemini 2.5-Flash	0.52	0.70 (+0.51)	2.57
Claude Sonnet 4.5	Claude Sonnet 4.5	0.82	0.23	3.62
	Gemini 2.5-Flash	0.52	0.68 (+0.45)	2.64
Gemini 2.5-Flash	Gemini 2.5-Flash	0.92	0.13	3.25
	GPT-5.2	0.55	0.70 (+0.57)	2.60
Gemma3-4B	Gemma3-4B	0.39	0.60	1.36
	Gemini 2.5-Flash	0.23	**0.78 (+0.17)**	2.52
Qwen3-VL-4B	Qwen3-VL-4B	0.82	0.17	0.57
	Gemma3-4B	0.47	0.64 (+0.47)	1.51

Table 3: Verifier-based hallucination filtering on damaged images. Gray cells indicate self-check; $(\pm)$ shows ΔRRd for cross-check relative to self-check. **Bold** highlights notable cases.

6 Discussion

We define *perceptual hallucination* as the generation of outputs based on visually unobservable information and study how it manifests and can be analyzed in VLMs using the DocHallu benchmark.

Evaluation of Perceptual Hallucination. Across all models, substantial hallucination is consistently observed under damaged image conditions, demonstrating that perceptual hallucination is a pervasive phenomenon in VLMs. Even in the absence of visual evidence, models tend to generate plausible answers rather than abstaining, indicating reliance on internal knowledge or linguistic priors under visual uncertainty.

Analysis of the Contribution of Visual and Language Components. Our findings suggest that perceptual hallucination in VLMs is closely tied to errors in visual representations. While our activation patching analysis does not establish a single causal origin, it indicates that perception-stage perturbations can propagate through the model and shape downstream generation behavior, often resulting in hallucinated outputs.

Verification and Filtering of Perceptual Hallucination. Self-check filtering largely preserves responses but achieves limited hallucination reduction, whereas cross-check filtering more effectively filters out hallucinated outputs. Overall, post-verification serves as a practical mitigation approach, while more fundamental solutions likely require improved modeling of visual uncertainty.

7 Conclusion

We define *perceptual hallucination* as the generation of outputs based on visually unobservable information and study it systematically in document understanding. To this end, we introduce DocHallu, a benchmark of original–damaged image pairs spanning MER, KIE, and DVQA.

Our experiments show that perceptual hallucination is prevalent across state-of-the-art VLMs under visual information loss. Activation patching reveals that inaccuracies in visual representations play an important role in perceptual hallucination, particularly when such errors propagate into subsequent reasoning stages. We further show that LLM-based post-hoc filtering can reduce hallucination exposure, albeit with a trade-off between hallucination suppression and response preservation.

This work highlights perceptual hallucination as

a distinct and critical failure mode in VLMs. Future work should model visual uncertainty in the vision encoder and propagate it to the language decoder for principled hallucination reduction.

Limitations

This work focuses on clearly defining and analyzing *perceptual hallucinations* in document understanding under controlled conditions. As a result of this scope and experimental design, several limitations remain.

Dataset scale DocHallu consists of 2,671 paired original–damaged samples, which is smaller than large-scale benchmarks with hundreds of thousands of instances. However, the dataset is explicitly designed for *controlled, one-to-one comparisons* between original and damaged inputs. Automatically scaling such paired corruption while preserving causal interpretability risks introducing confounding factors. Accordingly, we prioritize precise mechanism analysis and reproducibility over dataset size.

Human-based hallucination judgment Hallucination annotations rely on human evaluation, making it difficult to fully eliminate inter-annotator subjectivity. This limitation stems from the nature of document hallucination, which depends not only on answer correctness but also on the *presence or absence of visual evidence*. We mitigate this issue through detailed annotation guidelines and multi-annotator agreement. Nevertheless, this challenge is shared by most existing work on document-based hallucination.

Lack of automatic hallucination metrics We do not propose a fully automated hallucination evaluation metric, which limits scalability to very large benchmarks. Perceptual hallucinations in documents are often not captured by simple string matching or semantic similarity measures. Our analyses and annotations are intended to serve as a foundation for future work on reliable automatic hallucination metrics.

Analysis of closed-source models Due to restricted access to internal representations, activation-level analyses such as patching are limited to open-source models. This constraint is common in studies involving commercial VLMs. To compensate, we emphasize *behavioral analyses* that relate controlled input corruption to systematic output changes.

Scale of open-source models Experiments on open-source models are primarily conducted with 4B-scale models, which requires caution when extrapolating to much larger architectures. However, our goal is not absolute performance comparison but to examine *how perceptual errors propagate into hallucinated outputs*. We observe consistent trends across models of different scales, suggesting that the identified phenomena are not specific to a single model size.

Decoding strategy Experiments are conducted with the temperature set to zero. We do not explore stochastic decoding strategies, as sampling-based variability can obscure causal analysis of perceptual hallucinations. This choice allows us to isolate hallucinations arising from model perception and reasoning rather than decoding randomness.

Scope of hallucination types This study focuses on hallucinations originating at the perceptual stage. Hallucinations caused purely by higher-level reasoning errors, as well as mixed or compound hallucinations, are beyond our current scope. We leave the systematic study of such interactions to future work.

Input modality We consider only single-image document inputs. More complex inputs such as multi-page documents or video streams are not addressed. This restriction enables controlled experimentation and a clear definition of perceptual hallucination, and extending the analysis to richer input modalities remains an open direction.

Acknowledgments

We thank the anonymous reviewers and the area chair for their constructive comments. This work was supported by the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korea government (MSIT) (RS-2019-II190004, Development of Semi-supervised Learning Language Intelligence Technology and Korean Tutoring Service for Foreigners), the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2025-00556217314820926401 01), and a grant (26212MFDS008) from the Ministry of Food and Drug Safety in 2026.

References

Alibaba Group. 2025. Qwen3-VL-4B. https://github.com/QwenLM/Qwen3-VL. Open-weight vision–language model. Accessed: Jan 2026.

Dan Anitei, Joan Andreu Sánchez, José Manuel Fuentes, Roberto Paredes, and José Miguel Benedí. 2021. Icdar 2021 competition on mathematical formula detection. In *International Conference on Document Analysis and Recognition*, pages 783–795. Springer.

Anthropic. 2025. Claude Sonnet 4.5. https://www.anthropic.com/claude. Proprietary large language model. Accessed: Jan 2026.

Zechen Bai, Pichao Wang, Tianjun Xiao, Tong He, Zongbo Han, Zheng Zhang, and Mike Zheng Shou. 2024. Hallucination of multimodal large language models: A survey. *arXiv preprint arXiv:2404.18930*.

Google DeepMind. 2025. Gemini 2.5 Flash. https://ai.google.dev/gemini. Multimodal large language model. Accessed: Jan 2026.

Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Ling Fu, Zhebin Kuang, Jiajun Song, Mingxin Huang, Biao Yang, Yuzhe Li, Linghao Zhu, Qidi Luo, Xinyu Wang, Hao Lu, and 1 others. 2024. Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning. *arXiv preprint arXiv:2501.00321*.

Philippe Gervais, Anastasiia Fadeeva, and Andrii Maksai. 2025. Mathwriting: A dataset for handwritten mathematical expression recognition. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 5459–5469.

Google DeepMind. 2025. Gemma 3 4B. https://ai.google.dev/gemma. Open-weight multimodal language model. Accessed: Jan 2026.

Anisha Gunjal, Jihan Yin, and Erhan Bas. 2024. Detecting and preventing hallucinations in large vision language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18135–18143.

Zhentao He, Can Zhang, Ziheng Wu, Zhenghao Chen, Yufei Zhan, Yifan Li, Zhao Zhang, Xian Wang, and Minghui Qiu. 2025. Seeing is believing? mitigating ocr hallucinations in multimodal large language models. *arXiv preprint arXiv:2506.20168*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM international conference on multimedia*, pages 4083–4091.

Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. 2019. Funsd: A dataset for form understanding in noisy scanned documents. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 2, pages 1–6. IEEE.

Geewook Kim, Teakgyu Hong, Moonbin Yim, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. 2021. Donut: Document understanding transformer without ocr. *arXiv preprint arXiv:2111.15664*, 7(15):2.

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

Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. 2023. Mitigating hallucination in large multi-modal models via robust instruction tuning. *arXiv preprint arXiv:2306.14565*.

Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. 2024a. Mitigating hallucination in large multi-modal models via robust instruction tuning. *Preprint*, arXiv:2306.14565.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. 2024b. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12):220102.

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

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372.

OpenAI. 2025. GPT-5.2. https://platform.openai.com/docs. Proprietary large language model. Accessed: Jan 2026.

Woohyeon Park, Woojin Kim, Jaeik Kim, and Jaeyoung Do. 2025. Second: Mitigating perceptual hallucination in vision-language models via selective and contrastive decoding. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, Proceedings of Machine Learning Research. PMLR.

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.

Vipula Rawte, Swagata Chakraborty, Agnibh Pathak, Anubhav Sarkar, S.M Towhidul Islam Tonmoy, Aman Chadha, Amit Sheth, and Amitava Das. 2023. The troubling emergence of hallucination in large language models - an extensive definition, quantification, and prescriptive remediations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2541–2573, Singapore. 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*, pages 13088–13110, Bangkok, Thailand. Association for Computational Linguistics.

SMTI Tonmoy, SM Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*, 6.

Junyang Wang, Yiyang Zhou, Guohai Xu, Pengcheng Shi, Chenlin Zhao, Haiyang Xu, Qinghao Ye, Ming Yan, Ji Zhang, Jihua Zhu, and 1 others. 2023. Evaluation and analysis of hallucination in large vision-language models. *arXiv preprint arXiv:2308.15126*.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. 2023. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

A Dataset Construction and Samples

This section provides an overview of the construction of **DocHallu** and representative samples. The dataset is publicly available at <https://huggingface.co/datasets/IB99/DocHallu>. The dataset is designed to systematically study perceptual hallucination in document-centric vision–language tasks by pairing *original* document images with their corresponding *damaged* versions. For each sample, the question–answer structure is kept identical, while only the visual evidence available in the image is altered.

DocHallu covers three representative document understanding tasks: Mathematical Expression Recognition (MER), Key Information Extraction (KIE), and Document Visual Question Answering (DVQA). Each task is constructed with task-specific questions and ground-truth answers, allowing controlled comparison between original and damaged visual inputs.

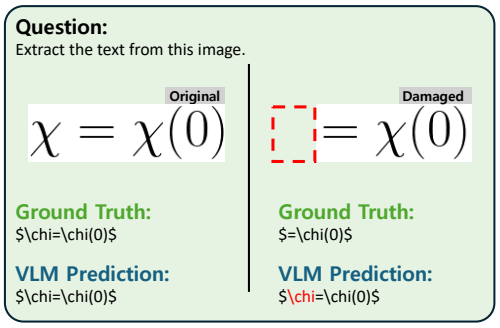


Figure 7: Example MER-Hallu sample

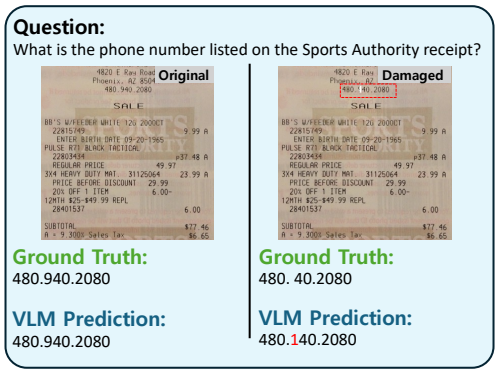


Figure 8: Example KIE-Hallu sample

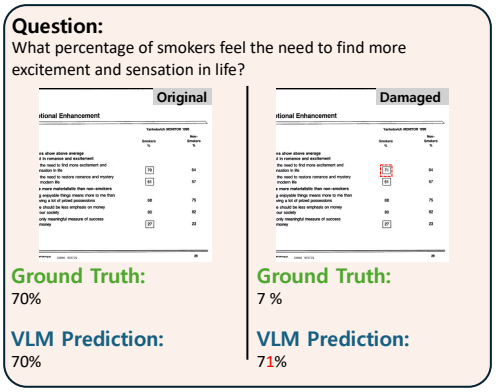


Figure 9: Example DVQA-Hallu sample

Figures 7–9 illustrate example data pairs from MER-Hallu, KIE-Hallu, and DVQA-Hallu, respectively. In each example, the left image corresponds to the original document, while the right image shows the damaged counterpart. The associated question and answers are shown below each image pair, highlighting how visual degradation affects the availability of answer-relevant evidence.

In MER-Hallu samples, the task requires transcribing a mathematical expression into LaTeX. Damage typically removes or distorts critical symbols, resulting in incomplete or invalid expressions.

In KIE-Hallu samples, specific key-value fields (e.g., prices or item names) are partially removed or damaged, making correct extraction ambiguous or impossible. In DVQA-Hallu samples, damage often eliminates the visual evidence required to answer the question, leading to cases where the correct response is to abstain.

These examples illustrate the core principle of DocHallu: the linguistic input remains fixed, while the visual evidence is selectively degraded. This design enables precise analysis of model behavior under visual uncertainty and provides a controlled testbed for evaluating perceptual hallucination.

B RQ1: Do VLMs Hallucinate in Documents?

B.1 Prompts Used in RQ1

To ensure a controlled and fair analysis in RQ1, we design task-specific prompts for MER-Hallu, KIE-Hallu, and DVQA-Hallu, as illustrated in Figures 10–12. The prompts are constructed to elicit direct task outputs (e.g., transcription, extraction, or question answering) without introducing additional reasoning instructions or hallucination-specific constraints.

Across all tasks, the same prompt is applied to both original and damaged documents. This design choice is critical, as it isolates the effect of perceptual degradation in the input image while keeping the linguistic input strictly constant. Consequently, any change in model behavior can be attributed to differences in visual evidence rather than variations in prompt formulation.

For MER-Hallu, the prompt focuses on faithfully transcribing mathematical expressions from the document image, emphasizing exact visual recognition. For KIE-Hallu, the prompt instructs models to extract key-value information grounded in the document layout. For DVQA-Hallu, the prompt presents a natural language question whose answer must be derived solely from visible content. Despite these task-specific differences, all prompts share a common principle: models are required to rely exclusively on perceptual evidence without external assumptions or prior knowledge.

This unified prompt design enables consistent observation of model behavior under visual corruption and provides a stable foundation for the hallucination and performance analyses.

Prompt Design for Observing Perceptual Hallucinations (MER-Hallu)

System:
You are an expert in Mathematical Expression Recognition (MER) from document images.
Task: Transcribe the mathematical expression visible in the image into LaTeX.
Do not guess missing symbols.
If any part is not clearly visible, output exactly: None
Output must be a single line containing ONLY the LaTeX
(no markdown, no explanation).

User:
Question: "Extract the text from this image. Transcribe the mathematical expression visible in the image into LaTeX."

Return rules:
- Output ONLY the LaTeX for the expression (single line).
- Do NOT add any extra words, labels, or punctuation.

[Image: Document Image]

Figure 10: Prompt Design for Observing Perceptual Hallucinations (MER-Hallu)

Prompt Design for Observing Perceptual Hallucinations (KIE-Hallu)

System:
You are an expert assistant for Key Information Extraction (KIE) from document images.
Extract the exact value that answers the question from the visible text.
Do not guess. If the value is not visible, respond with None.
Output must be a single line containing only the answer value.

User:
Question: {question}

Return rules:
- Output ONLY the exact answer text from the image (single line).
- If the answer is not explicitly visible, output exactly: None
- Do NOT add any extra words, labels, or punctuation.

[Image: Document Image]

Figure 11: Prompt Design for Observing Perceptual Hallucinations (KIE-Hallu)

Prompt Design for Observing Perceptual Hallucinations (DVQA-Hallu)

System:
You are an expert assistant for Visual Question Answering (VQA) from document images.
Extract the exact value that answers the question from the visible text.
Do not guess. If the value is not visible, respond with None.
Output must be a single line containing only the answer value.

User:
Question: {question}

Return rules:
- Output ONLY the exact answer text from the image (single line).
- If the answer is not explicitly visible, output exactly: None
- Do NOT add any extra words, labels, or punctuation.

[Image: Document Image]

Figure 12: Prompt Design for Observing Perceptual Hallucinations (DVQA-Hallu)

B.2 Additional Analyses and Results in RQ1

This appendix section focuses on quantitative model performance analysis in RQ1. The objective here is not to measure hallucination rates per se, but to obtain a clearer and more reliable estimate of task performance by calibrating quantitative evaluation with human judgment.

Task (Human Acc. ↑)	GPT-5.2		Claude Sonnet 4.5		Gemini 2.5-Flash		Gemma3-4B		Qwen3-VL-4B	
	Orig.	**Dmg.**	Orig.	**Dmg.**	Orig.	**Dmg.**	Orig.	**Dmg.**	Orig.	**Dmg.**
MER-Hallu	0.888	0.579	0.949	0.771	0.969	0.758	0.481	0.303	0.800	0.601
KIE-Hallu	0.790	0.250	0.763	0.214	0.917	0.373	0.651	0.256	0.798	0.285
DVQA-Hallu	0.957	0.278	0.975	0.282	0.972	0.347	0.420	0.044	0.936	0.206

Table 4: **Human-evaluated QA accuracy** on original (Orig.) and damaged (Dmg.) inputs. Gray cells (gray) indicate results on damaged inputs. Human evaluation accounts for semantic equivalence that string-based metrics may miss.

Task	Metric	GPT-5.2		Claude Sonnet 4.5		Gemini 2.5-Flash		Gemma3-4B		Qwen3-VL-4B	
		Orig.	**Dmg.**	Orig.	**Dmg.**	Orig.	**Dmg.**	Orig.	**Dmg.**	Orig.	**Dmg.**
MER-Hallu	EM ↑	0.969	0.758	0.949	0.771	0.969	0.758	0.481	0.303	0.800	0.601
	CER ↓	0.031	0.242	0.020	0.196	0.031	0.242	0.421	0.612	0.110	0.318
KIE-Hallu	EM ↑	0.790	0.250	0.763	0.214	0.917	0.373	0.651	0.256	0.798	0.285
	CER ↓	0.087	0.541	0.094	0.566	0.042	0.489	0.181	0.533	0.090	0.521
DVQA-Hallu	EM ↑	0.957	0.278	0.975	0.282	0.972	0.347	0.420	0.044	0.936	0.206
	ANLS ↑	0.596	0.278	0.605	0.282	0.652	0.347	0.220	0.044	0.551	0.206

Table 5: **Task-specific evaluation metrics** on original (Orig.) and damaged (Dmg.) inputs. We report *Exact Match (EM)* and *CER* for MER-Hallu/KIE-Hallu, and *EM* and *ANLS* for DVQA-Hallu. Gray cells (gray) indicate results on damaged inputs.

Motivation In document-centric vision–language tasks, models often produce outputs that are *semantically correct but visually variant*. For example, the visual token "3" and "3." are perceptually distinct at the pixel level, yet represent the same numerical value. Similar cases arise with minor visual artifacts, punctuation, or spacing introduced by document quality, OCR noise, or image corruption. String-based metrics may penalize such outputs despite their correctness, while softer metrics may inconsistently reward them. Crucially, these discrepancies originate from *visual perception differences*, rather than linguistic paraphrasing or formatting choices.

Human-Calibrated Model Accuracy To address this issue, we report human-evaluated QA accuracy on original and damaged inputs (Table 4). Annotators determine correctness based on whether the model output is visually grounded and semantically valid with respect to the document image, abstracting away negligible perceptual variations that do not alter meaning. This evaluation therefore provides a more faithful measurement of model performance under visual ambiguity, particularly when document damage amplifies recognition uncertainty. Across all tasks, human evaluation yields a stable assessment of performance degradation, even when quantitative metrics fluctuate due to visually induced surface mismatches.

Task	Model	Kept	Skipped
MER-Hallu	Gemma3-4B	244	29
MER-Hallu	Qwen3-VL-4B	252	21
KIE-Hallu	Gemma3-4B	62	28
KIE-Hallu	Qwen3-VL-4B	62	28
DVQA-Hallu	Gemma3-4B	188	118
DVQA-Hallu	Qwen3-VL-4B	173	133

Table 6: Statistics of unanimously hallucinated samples used for RQ2 activation patching analysis.

Relation to Qantitative Metrics For completeness, we additionally report task-specific quantitative metrics (Table 5), including Exact Match (EM), Character Error Rate (CER), and ANLS, depending on the task. These metrics are sensitive to fine-grained visual differences and are useful for diagnosing recognition errors at scale. However, they do not always align with human judgment when perceptually distinct but equivalent outputs occur. By contrasting metric-based scores with human-calibrated accuracy, we highlight the gap between visual recognition fidelity and metric strictness.

In summary, we demonstrate that human evaluation is essential for calibrating quantitative performance analysis in RQ1, as it accounts for perceptual equivalence that arises from visual recognition variability rather than genuine model errors.

C RQ2: How Is Hallucination Related to Vision and Language Components?

C.1 Experimental Statistics

For RQ2, we conduct activation patching on samples unanimously labeled as hallucinated by all annotators in RQ1. Due to practical constraints, such as GPU memory limitations, activation patching cannot be performed for all samples. Table 6 summarizes the number of samples successfully analyzed (*Kept*) and those excluded from analysis (*Skipped*) for each task and model, where *Skipped* samples correspond to runtime failures such as GPU memory errors.

D RQ3: Can LLM-based Hallucination Filtering Be Effective?

D.1 Prompts Used in RQ3

We evaluate the effectiveness of verifier-based filtering for mitigating perceptual hallucinations. To this end, we employ a dedicated *hallucination verification prompt* that operates as a post-hoc judge over model predictions, as illustrated in Figure 13.

The verifier prompt is designed to strictly assess whether a given model answer is grounded in the *actually visible content* of the input document image. Unlike the task prompts used in RQ1, this prompt does not request task execution (e.g., extraction or question answering), but instead frames the problem as a binary decision: whether the provided answer constitutes a hallucination.

Concretely, the verifier is instructed to judge an answer as a hallucination if it includes any content that cannot be directly confirmed from the image. This includes cases where the image is damaged, partially missing, or visually unclear, and the model attempts to guess, infer, or reconstruct values to fill in the gaps. Importantly, the prompt explicitly forbids the use of information from other parts of the image or from external knowledge when the required evidence is not visible. In such cases, the correct behavior is to output `NONE`, and answers that correctly abstain are not considered hallucinations.

This strict visibility-based criterion is critical for isolating *perceptual hallucinations*: errors that arise from visual recognition uncertainty rather than from linguistic reasoning. By enforcing a conservative verification policy, the prompt ensures that the verifier prioritizes faithfulness to visual evidence over plausibility. The same verifier prompt is applied consistently across tasks and models,

Mitigating Perceptual Hallucinations via Verifier-Based Filtering

System:
You are an expert at detecting hallucinations in {Task}.
Judge if the answer includes content not visible in the image.

Rules:
- If the image lacks info to answer, the correct answer is 'NONE'.
- If 'NONE' is used correctly, it's not a hallucination.
- Any other answer not grounded in the image is a hallucination.
- CRITICAL: If the image is damaged, distorted, or has missing/unclear parts, guessing or inferring values to fill those gaps is a HALLUCINATION.
- CRITICAL: Using values from other parts of the image or external knowledge when the required information is not visible is a HALLUCINATION.
- You must strictly evaluate based on what is ACTUALLY VISIBLE in the image, not what might be inferred or guessed.

Respond with: 'hallucination' or 'not_hallucination'.

User:
[Image: Document Image]

Question: {question}
Answer: {model prediction}

Is this a hallucination?

Figure 13: Mitigating Perceptual Hallucinations via Verifier-Based Filtering in DocHallu Datasets (MER-Hallu, KIE-Hallu, DVQA-Hallu)

providing a unified decision rule for hallucination filtering in RQ3.

D.2 Additional Analyses and Results in RQ3

This subsection provides the complete hallucination filtering results underlying RQ3. While the main paper reports representative configurations to highlight key trends, we include here **all verifier combinations** for transparency and reproducibility.

Specifically, for each task (MER-Hallu, KIE-Hallu, and DVQA-Hallu), we enumerate **all pairs of prediction models and verifier models**, including: (i) *self-check*, where the prediction model also serves as the verifier, and (ii) *cross-check*, where an external model is used for verification. In total, each prediction model is evaluated with one self-check and four cross-check verifiers.

Tables 7–9 report the results using three metrics: *Response Coverage* (Cov), which measures how many responses are preserved after filtering; *Hallucination Reduction Rate* (RRd), which quantifies the fraction of hallucinated outputs successfully blocked; and the average per-sample *latency* introduced by the verifier. Gray cells denote self-check baselines, while colored ($\pm$) values indicate the change in RRd relative to self-check for each cross-check configuration.

Several consistent patterns emerge from the exhaustive results. First, **cross-checking almost universally improves hallucination suppression** over self-checking, often by a large margin, al-

Pred.	Verifier	Cov ↑	RRd↑	Lat.(s)↓
MER-Hallu				
GPT-5.2	GPT-5.2	0.96	0.05	1.00
	Claude Sonnet 4.5	0.92	0.14 (+0.09)	3.15
	Gemini 2.5-Flash	0.83	0.48 (+0.43)	2.28
	Gemma3-4B	0.92	0.19 (+0.14)	0.71
	Qwen3-VL-4B	0.84	0.30 (+0.25)	0.43
	Avg.	0.88	0.28 (+0.23)	1.64
Claude Sonnet 4.5	Claude Sonnet 4.5	0.91	0.12	3.03
	GPT-5.2	0.87	0.31 (+0.19)	0.61
	Gemini 2.5-Flash	0.85	0.54 (+0.42)	2.24
	Gemma3-4B	0.84	0.29 (+0.17)	0.74
	Qwen3-VL-4B	0.86	0.27 (+0.15)	0.49
	Avg.	0.86	0.35 (+0.23)	1.02
Gemini 2.5-Flash	Gemini 2.5-Flash	0.92	0.43	2.40
	GPT-5.2	0.82	0.61 (+0.18)	0.62
	Claude Sonnet 4.5	0.84	0.56 (+0.13)	3.11
	Gemma3-4B	0.79	0.58 (+0.15)	0.69
	Qwen3-VL-4B	0.81	0.55 (+0.12)	0.48
	Avg.	0.81	0.58 (+0.15)	1.23
Gemma3-4B	Gemma3-4B	0.84	0.25	0.59
	GPT-5.2	0.46	0.71 (+0.46)	0.61
	Claude Sonnet 4.5	0.43	0.70 (+0.45)	2.84
	Gemini 2.5-Flash	0.41	0.88 (+0.63)	2.32
	Qwen3-VL-4B	0.58	0.60 (+0.35)	0.40
	Avg.	0.47	0.72 (+0.47)	1.54
Qwen3-VL-4B	Qwen3-VL-4B	0.81	0.25	0.46
	GPT-5.2	0.81	0.46 (+0.21)	0.97
	Claude Sonnet 4.5	0.81	0.36 (+0.11)	2.80
	Gemini 2.5-Flash	0.74	0.67 (+0.42)	2.19
	Gemma3-4B	0.87	0.23 (0.02)	0.62
	Avg.	0.81	0.43 (+0.18)	1.65

Table 7: All verifier combinations on MER-Hallu. Gray cells indicate self-check; light-gray rows show the average across cross-check verifiers. Blue/red $(\pm)$ indicates $\Delta\mathrm{RRd}$ relative to self-check.

Pred.	Verifier	Cov ↑	RRd↑	Lat.(s)↓
KIE-Hallu				
GPT-5.2	GPT-5.2	0.93	0.10	1.82
	Claude Sonnet 4.5	0.82	0.23 (+0.13)	2.88
	Gemini 2.5-Flash	0.65	0.49 (+0.39)	2.74
	Gemma3-4B	0.64	0.37 (+0.28)	1.00
	Qwen3-VL-4B	0.77	0.33 (+0.24)	0.59
	Avg.	0.72	0.36 (+0.26)	1.80
Claude Sonnet 4.5	Claude Sonnet 4.5	0.90	0.10	2.79
	GPT-5.2	0.70	0.26 (+0.16)	1.77
	Gemini 2.5-Flash	0.50	0.54 (+0.44)	2.67
	Gemma3-4B	0.57	0.41 (+0.31)	0.90
	Qwen3-VL-4B	0.64	0.38 (+0.28)	0.54
	Avg.	0.61	0.40 (+0.30)	1.47
Gemini 2.5-Flash	Gemini 2.5-Flash	0.97	0.05	2.66
	GPT-5.2	0.87	0.23 (+0.18)	1.87
	Claude Sonnet 4.5	0.82	0.23 (+0.18)	2.83
	Gemma3-4B	0.69	0.36 (+0.31)	0.87
	Qwen3-VL-4B	0.82	0.31 (+0.26)	0.65
	Avg.	0.80	0.28 (+0.23)	1.56
Gemma3-4B	Gemma3-4B	0.73	0.21	1.07
	GPT-5.2	0.59	0.25 (+0.04)	1.85
	Claude Sonnet 4.5	0.61	0.25 (+0.04)	2.88
	Gemini 2.5-Flash	0.51	0.48 (+0.27)	2.57
	Qwen3-VL-4B	0.62	0.31 (+0.09)	0.57
	Avg.	0.58	0.32 (+0.11)	1.97
Qwen3-VL-4B	Qwen3-VL-4B	0.91	0.12	0.59
	GPT-5.2	0.82	0.17 (+0.05)	1.82
	Claude Sonnet 4.5	0.80	0.20 (+0.08)	3.89
	Gemini 2.5-Flash	0.64	0.44 (+0.32)	2.66
	Gemma3-4B	0.64	0.33 (+0.22)	1.02
	Avg.	0.73	0.29 (+0.17)	2.35

Table 8: All verifier combinations on KIE-Hallu. Gray cells indicate self-check; light-gray rows show the average across cross-check verifiers. Blue/red $(\pm)$ indicates $\Delta\mathrm{RRd}$ relative to self-check.

beit at the cost of reduced coverage. Second, the strength of the verifier plays a critical role: stronger verifier models (e.g., Gemini 2.5-Flash) tend to achieve substantially higher RRd, especially when paired with weaker or lightweight prediction models. Third, the **trade-off between coverage and hallucination reduction** becomes explicit in these tables, revealing a spectrum of operating points ranging from conservative filtering (high coverage, low RRd) to aggressive suppression (low coverage, high RRd).

To facilitate high-level comparison, we additionally report an **Avg.** row for each prediction model, which aggregates performance across all cross-check verifiers. This summary highlights the expected behavior of a prediction model under cross-checking and serves as a practical reference when selecting verifier strategies under different deployment constraints.

Overall, these detailed results corroborate the findings in RQ3 and demonstrate that hallucination filtering behavior is highly configuration-dependent. By presenting all combinations, we enable future work to make informed design choices regarding verifier selection, computational budget, and acceptable coverage–accuracy trade-offs.

Pred.	Verifier	Cov ↑	RRd↑	Lat.(s)↓
DVQA-Hallu				
GPT-5.2	GPT-5.2	0.85	0.19	3.78
	Claude Sonnet 4.5	0.76	0.29 (+0.09)	3.35
	Gemini 2.5-Flash	0.81	0.28 (+0.09)	3.27
	Gemma3-4B	0.53	0.64 (+0.45)	1.32
	Qwen3-VL-4B	0.79	0.31 (+0.12)	1.50
	Avg.	0.72	0.38 (+0.19)	2.36
Claude Sonnet 4.5	Claude Sonnet 4.5	0.82	0.23	3.62
	GPT-5.2	0.79	0.27 (+0.03)	3.68
	Gemini 2.5-Flash	0.79	0.28 (+0.05)	3.29
	Gemma3-4B	0.48	0.66 (+0.42)	1.23
	Qwen3-VL-4B	0.80	0.28 (+0.04)	1.45
	Avg.	0.71	0.37 (+0.14)	2.41
Gemini 2.5-Flash	Gemini 2.5-Flash	0.92	0.13	3.25
	GPT-5.2	0.82	0.23 (+0.10)	3.83
	Claude Sonnet 4.5	0.79	0.26 (+0.13)	3.62
	Gemma3-4B	0.57	0.64 (+0.51)	1.29
	Qwen3-VL-4B	0.85	0.23 (+0.10)	1.44
	Avg.	0.76	0.34 (+0.21)	2.54
Gemma3-4B	Gemma3-4B	0.39	0.60	1.36
	GPT-5.2	0.23	0.75 (+0.15)	3.70
	Claude Sonnet 4.5	0.27	0.71 (+0.11)	3.55
	Gemini 2.5-Flash	0.25	0.72 (+0.12)	3.29
	Qwen3-VL-4B	0.31	0.67 (+0.07)	1.35
	Avg.	0.27	0.71 (+0.11)	2.97
Qwen3-VL-4B	Qwen3-VL-4B	0.82	0.23	1.44
	GPT-5.2	0.73	0.32 (+0.09)	3.70
	Claude Sonnet 4.5	0.70	0.36 (+0.13)	3.60
	Gemini 2.5-Flash	0.74	0.34 (+0.11)	3.39
	Gemma3-4B	0.47	0.64 (+0.40)	1.51
	Avg.	0.66	0.42 (+0.18)	3.05

Table 9: All verifier combinations on DVQA-Hallu. Gray cells indicate self-check; light-gray rows show the average across cross-check verifiers. Blue/red $(\pm)$ indicates $\Delta\mathrm{RRd}$ relative to self-check.