# Cure or Poison? Embedding Instructions Visually Alters Hallucination in Vision-Language Models

Anonymous ACL submission

## Abstract

Vision-Language Models (VLMs) often suffer from hallucination, partly due to challenges in aligning multimodal information. We propose Prompt-in-Image, a simple method that embeds textual instructions directly into images. This removes the need for separate text inputs and forces the model to process all content through the visual channel. We evaluate this method on three popular open-source VLMs: Qwen2.5-VL, LLaVA-1.5, and InstructBLIP. The results reveal sharp differences. Prompt-in-Image improves Qwen2.5-VL's performance, increasing POPE accuracy by 4.1% (from 80.2% to 84.3%) and also reducing hallucination rates on MS-COCO. In contrast, LLaVA-1.5 and InstructBLIP experience a severe performance drop, with accuracy falling from around 84% to near-random levels. Through detailed analysis, we found that CLIP-based encoders in LLaVA and InstructBLIP exhibit excessive attention bias toward embedded text regions, disrupting visual understanding. In contrast, Qwen's vision encoder handles text-embedded images robustly. Crucially, Prompt-in-Image reduces Qwen's modality gap, enhancing cross-modal alignment by unifying information processing through a single modality.

## 1 Introduction

Most modern Vision-Language Models (VLMs) follow a standard architecture: a visual encoder (typically ViT), a projector, and a language model (LLM decoder). Since visual and textual components are pre-trained separately (Rabinovich et al., 2023), this approach introduces inherent cross-modal alignment challenges, significantly hampering the model's overall performance. One prominent manifestation of these alignment issues is the language bias phenomenon, where VLMs disproportionately rely on textual information while ignoring visual information (Niu et al., 2021; Wang et al., 2024a, 2025). To address these cross-modal alignment challenges, previous approaches have focused on improving cross-modal fusion. However, instead of enhancing cross-modal integration, we ask whether we can avoid cross-modal alignment challenges entirely by relying solely on single-modality information.

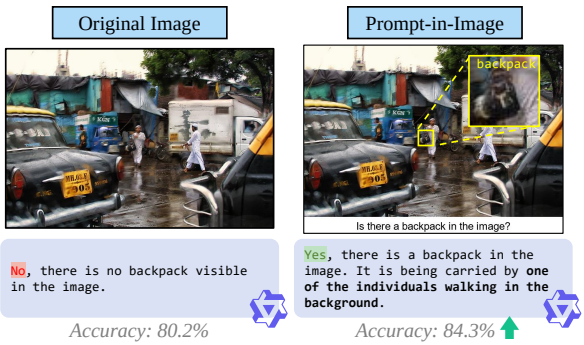


Figure 1: An example of Prompt-in-Image: the text instruction is directly embedded into the image. Using Prompt-in-Image, Qwen2.5-vl performance improves.

We propose Prompt-in-Image (Figure 1), which embeds textual instructions directly into images. By forcing models to process all information through the visual channel, this approach may enhance fusion and reduce alignment issues.

We use hallucination as our primary evaluation task for two key reasons: (1) hallucination represents a major challenge in VLM development, where models describe non-existent objects or miss key visual details, seriously affecting VLM performance and reliability; and (2) hallucination is highly correlated with modality alignment issues (Liu et al., 2024), making it an ideal testbed for evaluating our approach.

We use POPE, a representative hallucination benchmark, to extensively test three popular VLMs: Qwen2.5-VL, InstructBLIP and LLaVA-1.5. Surprisingly, we observe opposite effects. On POPE, Qwen2.5-VL's accuracy improves by 4–5%, while LLaVA-1.5's performance drops dramatically from

84% to 55% (a near-complete collapse). Similarly, InstructBLIP shows consistent behavior with LLaVA-1.5, declining from 74.4% to 54%. Our analysis reveals two key findings. First, LLaVA and InstructBLIP's collapse comes from its CLIP-based encoder's strong text bias, which gives too much attention to embedded text regions. This creates severe hallucinations. In contrast, Qwen's vision encoder handles text-embedded images much better. Second, Prompt-in-Image effectively reduces Qwen's modality gap, improving cross-modal alignment and boosting performance.

In summary, our contributions are threefold:

1. We propose *Prompt-in-Image*, a novel input strategy that embeds text into images to improve modality integration.

2. We conduct systematic evaluations on Qwen-VL, InstructBLIP and LLaVA-1.5, revealing divergent effects of Prompt-in-Image on hallucination performance.

3. We conduct an in-depth analysis of the performance gap and explain how Prompt-in-Image improves performance by reducing modality gaps and enhancing alignment

The rest of this paper is organized as follows. Section 2 reviews related work on VLM architectures, hallucination problems, and existing mitigation methods. Section 3 presents our Prompt-in-Image method, including the design details, evaluation benchmarks (POPE and MS-COCO), tested models, and experimental configurations. Section 4 reports our experimental results, showing Prompt-in-Image's contrasting effects on different models. Section 5 provides an in-depth analysis to explain these divergent outcomes. Finally, Section 6 concludes the paper and discusses future directions.

## 2 Related Works

### 2.1 Vision-Language Models and Hallucination

Vision-Language Models (VLMs) have rapidly evolved in recent years, with many powerful models like GPT-4o (OpenAI, 2024), LLaVA (Liu et al., 2023b), and Qwen-VL (Bai et al., 2025) achieving impressive performance. These models typically share similar architectures: a vision encoder to process images, a projection layer, and a language model to generate text. Despite their success in various tasks, hallucination remains a critical challenge for VLMs. Hallucinations can be categorized into two types. Judgement hallucination occurs when the model's response to a user's query is in disagreement with the actual visual data. Description hallucination is a failure to faithfully depict the visual information (Liu et al., 2024). To comprehensively evaluate and quantify this problem, researchers have proposed various VLM hallucination evaluation methods and benchmarks, including POPE (Li et al., 2023), NOPE (Lovenia et al., 2023), CHAIR (Rohrbach et al., 2018), MMHal-Bench (Sun et al., 2023), and AMBER (Wang et al., 2023).

### 2.2 Hallucination Causes and Mitigation Methods

The causes of hallucination in VLMs are complex and multifaceted. Several important factors contribute to this problem, such as data bias (Liu et al., 2023a), limitations in vision encoders (Li et al., 2024; Cho et al., 2022; Gong et al., 2024), poor modality alignment (Sun et al., 2023), and the inherent hallucination of LLMs. To address these issues, various methods have been proposed. Among them, training-free contrastive decoding (CD) strategies have shown effectiveness in reducing hallucination. Contrastive decoding reduces hallucination by comparing model outputs from original and perturbed inputs—such as visually noised images or modified instructions—to suppress over-reliance on language priors. Representative examples include Visual Contrastive Decoding (VCD) (Leng et al., 2024) and Instruction Contrastive Decoding (ICD) (Wang et al., 2024b). However, these methods also come with limitations, including slower inference speed and limited performance gains. Moreover, some recent studies (Yin et al., 2025) have argued that such decoding strategies may be entirely unrelated to the original objective of hallucination mitigation.

## 3 Method

### 3.1 Prompt-in-Image Design

In traditional VQA tasks, users provide both an image and textual instructions (prompt), and VLMs process both visual and textual inputs to complete the task. To enable models to rely on single-modality information, we adopt a straightforward approach: directly embedding instructions into the image, similar to movie subtitles. We then provide

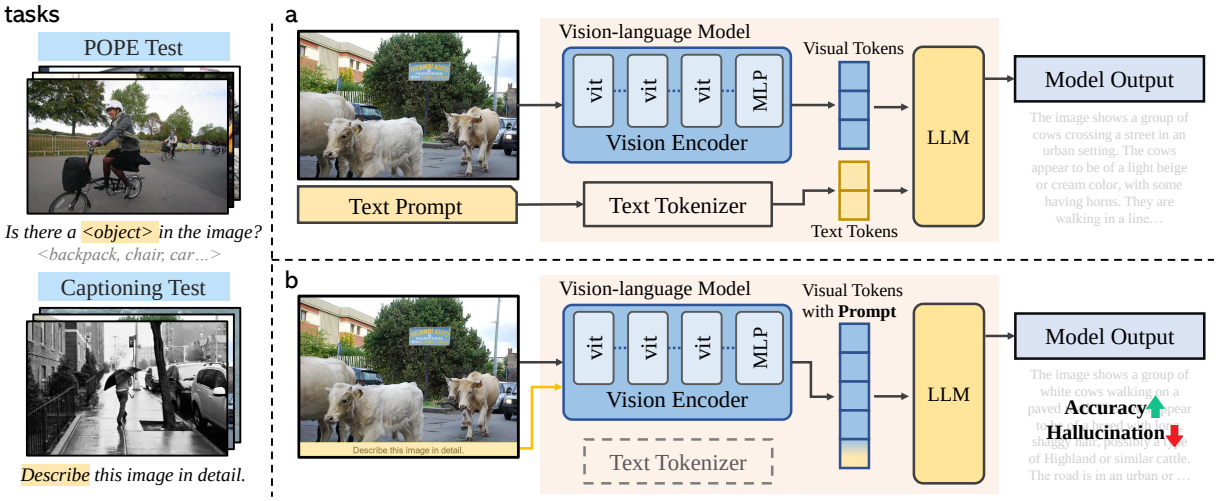


Figure 2: Comparison of interaction paradigms. (a) Traditional VLM interaction requires both an image and a separate text prompt as input. (b) Prompt-in-Image embeds the instruction directly into the image, allowing users to provide only the image without any extra textual input.

only the image to the VLM, eliminating separate textual input. We call this approach "Prompt-in-Image."

To create Prompt-in-Image, we render the question text at the bottom of each image. To avoid occluding image content, we add a separate white rectangular area below each image. The question is rendered in black Arial font (26pt), ensuring machine readability. This text region covers only about 5% of the total image height, minimizing interference with the original visual content. To control for visual changes unrelated to text, we include a white blank box condition: a control group where the same white rectangular area is added without any text. The image size in this group is kept identical to that in the Prompt-in-Image condition.

### 3.2 Benchmarks and Evaluation Metrics

We evaluate Prompt-in-Image on two complementary benchmarks:

**POPE (Polling-based Object Probing Evaluation)** (Li et al., 2023): This dataset evaluates object existence detection using binary questions such as "*Is there a <object> in the image?*", where *<object>* is selected from three splits: random (randomly selected objects), popular (frequently occurring objects), and adversarial (objects closely related to those in the image). We focus on the adversarial subset, which is the most challenging and closely reflects real-world hallucination scenarios. The complete adversarial dataset consists of 3,000 questions (500 images × 6 questions per image). To balance comprehensive evaluation with computational efficiency, we randomly selected 1,000 questions for testing. Model performance is measured using accuracy and F1 score.

**MS-COCO Caption**: We randomly select 500 images from the MS-COCO 2017 (Lin et al., 2014) validation set and prompt the VLMs with "*Describe this image in detail.*" Hallucination is evaluated using the **CHAIR** metric (Rohrbach et al., 2018), which compares generated captions against ground-truth object labels. We report two scores: **CHAIRi**, the proportion of hallucinated objects among all mentioned objects; and **CHAIRs**, the proportion of captions containing at least one hallucinated object.

These two datasets are well-established benchmarks and provide a comprehensive evaluation framework, covering both binary question answering and open-ended generation tasks.

## 4 Experimental Results

### 4.1 Models

We evaluate our method on two widely used open-source VLMs: **Qwen2.5-VL-7B** (Bai et al., 2025), **InstructBLIP-vicuna-7B** (Dai et al., 2023) and **LLaVA-v1.5-7B** (Liu et al., 2023b). All three models follow similar transformer-based architectures and demonstrate strong performance across multimodal tasks. Importantly, all three are capable of accurately recognizing questions embedded in images, making them suitable for evaluating the effectiveness of Prompt-in-Image.

| Setting | Qwen2.5-VL-7B | | | InstructBLIP-7B | | | LLaVA-v1.5-7B | | |
|---|---|---|---|---|---|---|---|---|---|
| | Acc. | F1 | Yes Ratio | Acc. | F1 | Yes Ratio | Acc. | F1 | Yes Ratio |
| Baseline | 80.2 | 0.76 | 0.32 | 74.4 | 0.72 | 0.53 | 84.0 | 0.86 | 0.62 |
| Control | 80.5(+0.3) | 0.77 | 0.33 | 75.0(+0.6) | 0.77 | 0.53 | 84.0(±0) | 0.86 | 0.62 |
| Hybrid | 83.0(+2.8) | 0.81 | 0.38 | 63.2(-11.2) | 0.69 | 0.68 | 64.0(-20.0) | 0.74 | 0.82 |
| Prompt-in-Image | **84.3**(+4.1) | **0.82** | 0.38 | – | – | – | – | – | – |
| Prompt-in-Image† | 82.1(+1.9) | 0.81 | 0.43 | 54.0(-20.4) | 0.70 | 0.99 | 55.0(-29.0) | 0.70 | 0.99 |

Table 1: Performance of different input settings on three VLMs. Prompt-in-Image (without instruction) is only applicable to Qwen2.5-VL. †With explicit instruction "*Answer the questions in the image*".

| Setting | CHAIRs (%) | CHAIRi (%) |
|---|---|---|
| Baseline | 32.3 | 8.8 |
| Hybrid | 34.2(+1.9) | 9.9(+1.1) |
| Prompt-in-Image | 24.7(-7.6) | 6.7(-2.1) |

Table 2: Hallucination rates on MS-COCO using CHAIR metrics. Prompt-in-Image reduces both CHAIRs and CHAIRi compared to other input modes.

### 4.2 Experimental Configuration

We evaluate four input configurations to test the effect of Prompt-in-Image:

- **Baseline:** Original image + text prompt.
- **Prompt-in-Image:** Image with embedded question + no text prompt (system messages like "You are a helpful assistant." may still be present).
- **Hybrid:** Image with embedded question + text prompt.
- **Control:** Image with a blank white box (no text) + text prompt. This controls for the visual change introduced by the prompt region.

All experiments are conducted with identical inference parameters and fixed random seeds to ensure reproducibility. The decoding temperature is set to 0.7 (default). These configurations allow us to isolate the effects of visual input, textual input, and their interaction.

VLMs behave very differently when processing our Prompt-in-Image samples. Qwen2.5-VL works quite naturally with embedded questions—it can directly read and answer questions that are placed in the image without needing any extra instructions from us. LLaVA-1.5 and InstructBLIP, however, act differently. When we don't give them specific text instructions, they tend to treat embedded text as just another part of the image to describe. Instead of answering the embedded question, they often give general descriptions of what they see.

To ensure fair comparison, we give InstructBLIP and LLaVA-1.5 a clear instruction for all Prompt-in-Image tests: "*Answer the questions in the image.*" This prevents LLaVA from just describing the image and ensures both models are actually trying to answer the embedded questions. Additionally, for experimental rigor, we conducted the same experiments with the Qwen model (see Table 1). We will analyze these results in detail in the next section.

### 4.3 POPE Evaluation

Table 1 summarizes the performance comparison between three VLMs on POPE. The results reveal striking differences between models:

**Qwen2.5-VL** demonstrates consistent improvements with Prompt-in-Image. Accuracy increases by 4.1% (80.2% → 84.3%). The Hybrid configuration also shows gains (+2.8%), though slightly lower than Prompt-in-Image. We examine specific examples to better understand how Prompt-in-Image changes Qwen's behavior (Figure 3, Top). The results show clear differences between baseline and Prompt-in-Image responses. Surprisingly, we find that Prompt-in-Image helps the model detect small, unusual, or partially hidden objects that it missed before. More importantly, Qwen doesn't just identify these challenging objects—it can also tell us exactly where they are located in the image (e.g., "on the right side of the image"). This suggests that Prompt-in-Image actually improves Qwen's visual understanding and ability to ground objects in the scene, rather than simply making it say "yes" more often to questions.

**LLaVA-v1.5** and **InstructBLIP** exhibit catastrophic performance degradation. LLaVA-v1.5 drops dramatically from 84.0% baseline accuracy to 55.0%, while InstructBLIP shows very similar behavior, declining from 74.4% to 54.0%. More


Figure 3: Case studies on Qwen2.5-VL. Top: POPE examples comparing baseline and Prompt-in-Image responses for object presence detection. Bottom: An MS-COCO captioning example. Prompt-in-Image helps the model generate more detailed answers and with lower hallucination.

concerning, both models show Yes Ratios jumping to 0.99, indicating they default to "yes" for nearly all queries. This represents a complete loss of discriminative capability for both models.

In all models, the control groups perform nearly identically to the baseline, confirming that the performance changes are caused by the presence of text in the image, rather than layout or formatting changes.

### 4.4 MS-COCO Validation

Given Qwen2.5-VL's promising results on POPE, we conducted further evaluation using MS-COCO Caption to assess performance on open-ended generation tasks. We tested 500 images, and evaluated model outputs using the CHAIR metric. Prompt-in-Image yields consistent improvements across both hallucination metrics. In Table 2, CHAIRs decreases by 7.6% (32.3% $\rightarrow$ 24.7%) and CHAIRi by 2.1% (8.8% $\rightarrow$ 6.7%), indicating reduced hallucination at both sentence and instance levels. Figure 3 (Bottom) shows a good example of this improvement. With Prompt-in-Image, the model generates much more detailed captions, including specific details like clothing colors and small text visible in the image. This is interesting because the model actually says more and gives more details, but makes fewer mistakes. Prompt-in-Image helps the model see and describe what's really there, rather than just making up information.

## 5 Divergent Effects of Prompt-in-Image

Our experiments reveal contradictory effects of Prompt-in-Image across different models. This section addresses two critical questions:

- Why does InstructBLIP and LLaVA's performance degrade with Prompt-in-Image?
- How does Prompt-in-Image enhance Qwen's performance?

We examine both questions in detail in this section.

### 5.1 Why Prompt-in-Image Hurts LLaVA

We analyze CLIP ViT-L/14 (Radford et al., 2021), which serves as the visual encoder in LLaVA-v1.5 and is also closely related to the visual encoder in InstructBLIP. We visualize the patch-level attention weights across different layers (Figure 4), using the average attention across all heads. Specifically,

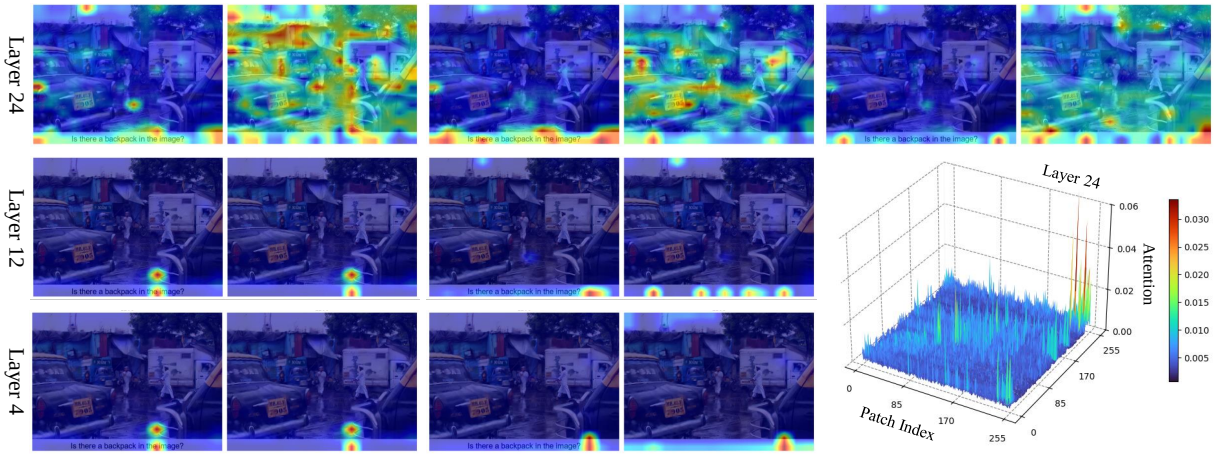


Figure 4: CLIP attention visualization across layers 4, 12, and 24 on the example image. Each row shows patch-level attention weights for Prompt-in-Image (with embedded text) versus Control Group (text-free images). Deep layers (layer 24) exhibit strong attention bias toward text regions.

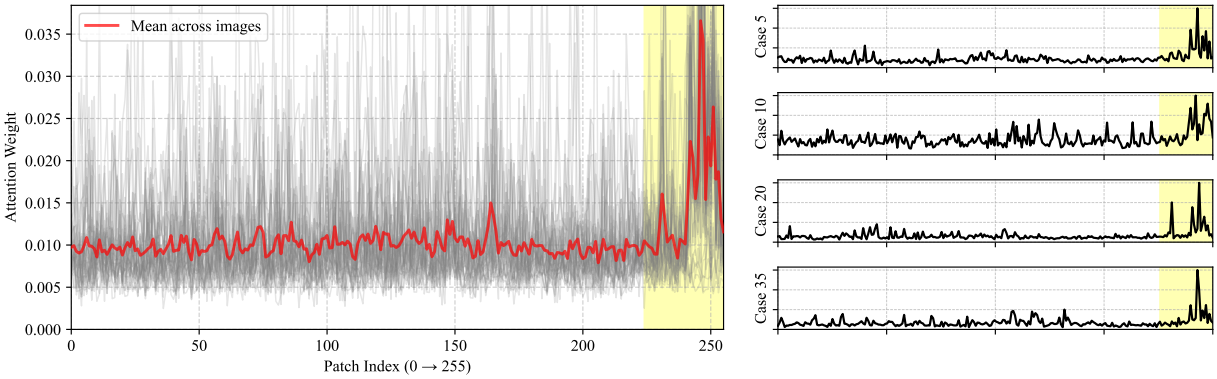


Figure 5: Self-attention analysis of CLIP's final layer across 35 Prompt-in-Image images. The red line shows average attention weights across all images, while gray lines represent individual samples. The yellow region corresponds to text-related patches, which consistently receive much higher self-attention scores.

we examine three representative layers: a shallow layer (layer 4), a middle layer (layer 12), and a deep layer (layer 24). The goal is to compare how CLIP processes two types of input: images with embedded text (Prompt-in-Image) and their text-free counterparts (Control Group).

We observe that the attention distributions in shallow (layer 4) and middle layers (layer 12) are similar across both image types, suggesting limited sensitivity to embedded text at early stages. However, in the deep layer (layer 24), the difference becomes more pronounced: CLIP shows a clear tendency to focus attention heavily on the text region.

To further confirm CLIP's text bias, we quantify this effect by analyzing 35 randomly selected Prompt-in-Image images from the POPE dataset. We examine the self-attention patterns in the final layer (layer 24) of the CLIP encoder, focusing on the diagonal of the attention matrix, which shows

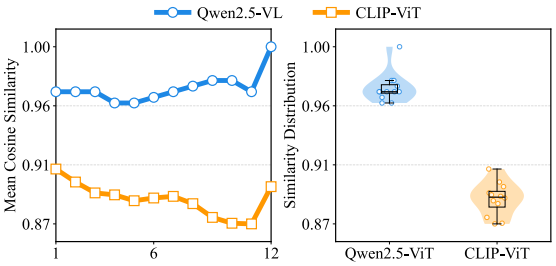


Figure 6: Layer-wise similarity comparison between Qwen2.5-ViT and CLIP on an example image pair.

how much attention each patch gives to itself. Figure 5 shows that the last 32 patches, corresponding to the text region, consistently have very high self-attention scores. This confirms that CLIP is overly sensitive to embedded text, giving too much attention to text regions in deeper layers.Many previous works (Darcet et al., 2023; Gong et al., 2024; Zhang et al., 2024) suggest that excessive attention

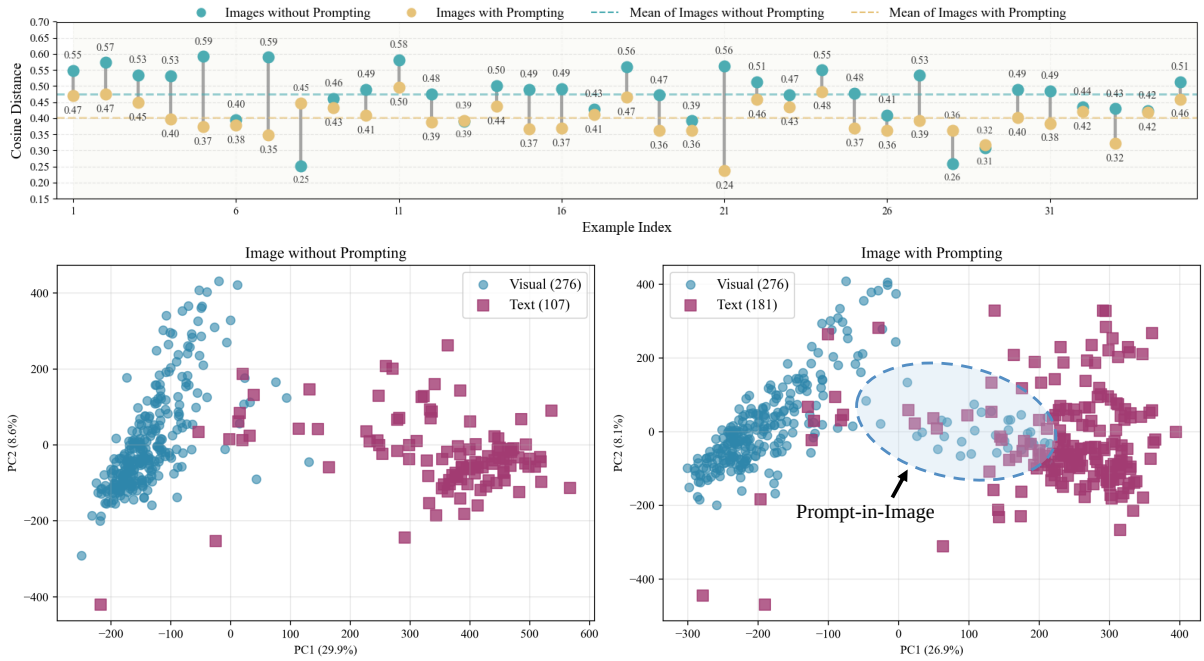


Figure 7: Modality gap analysis for Qwen2.5-VL. Top: Comparison of average cosine distances between image and caption embeddings across 35 samples. Most Prompt-in-Image samples (yellow) show lower modality gaps compared to baseline (blue). Bottom: PCA visualization of image (blue) and text (purple) embeddings for one example. Prompt-in-Image (right) brings the two modalities closer together compared to the baseline condition (left).

to certain visual tokens can lead to hallucinations in VLM outputs. When attention weights concentrate disproportionately on specific visual tokens (like those containing embedded text) the model loses its ability to balance local and global visual information. These dominant tokens can override both visual context and linguistic priors, causing the model to ignore actual image content and default to affirmative responses regardless of other visual evidence.

Additionally, we extract features from all transformer layers when processing Prompt-in-Image and Control image pairs, computing cosine similarity across patches at each layer. Figure 6 shows the similarity profiles for the final 12 layers, where high-level semantic representations emerge.

The results reveal a striking difference: Qwen-ViT maintains consistently high similarity (>0.95) between Prompt-in-Image and Control images throughout its deep layers, while CLIP-ViT shows declining similarity in the same layers. This indicates that Qwen's vision encoder preserves image semantics despite embedded text, whereas CLIP becomes increasingly sensitive to textual elements in deeper layers.

This robustness likely stems from Qwen's diverse pretraining regime, which includes not only standard image-caption pairs but also interleaved image-text documents and OCR data (Bai et al., 2025). By learning to process images with naturally embedded text during pretraining, Qwen-ViT develops representations that treat text as a normal visual element rather than a disruptive signal—explaining why it can successfully interpret visual prompts without catastrophic attention shifts.

### 5.2 Why Prompt-in-Image Helps Qwen

In contrast to InstructBLIP and LLaVA-1.5, Qwen2.5-VL shows consistent performance gains under Prompt-in-Image. On the POPE dataset, accuracy improves by 4–5%, and on the open-ended caption generation task for MS-COCO images, hallucination rates also decrease.

We propose a possible explanation: Prompt-in-Image unifies the input modality through the visual channel, thereby enhancing modality fusion and mitigating alignment issues. This hypothesis is supported by two observations.

**Text input disrupts Prompt-in-Image performance**: As shown in Section 4, the Hybrid setting (image with embedded question and separate text prompt) performs worse than the Prompt-in-Image

setting (image with embedded question only). This suggests that introducing an additional modality (i.e., text) does not help the model and instead degrades performance. In contrast, Prompt-in-Image consolidates all relevant information into a single modality, which helps the model focus and enhances overall performance.

**Prompt-in-Image reduces the modality gap**: Previous studies (Liang et al., 2022) have identified the "modality gap" phenomenon in vision-language models. While these models are designed to map images and text into a shared representation space, different modalities actually end up clearly separated in this space. This separation negatively affects performance across multiple downstream tasks, and reducing this gap has been shown to improve model performance (Role et al., 2025).

Ideally, image embeddings and their corresponding caption embeddings should overlap closely in the representation space, indicating good cross-modal alignment. To test whether Prompt-in-Image improves this alignment in Qwen, we examine the modality gap between image and caption embeddings.

We randomly selected 35 samples from the MS-COCO test 4 set and formed two groups:

- Image with a blank white box (no text) and its caption generated with explicit textual instruction "Describe this image in detail".
- Image with an embedded question (Prompt-in-Image) and its caption generated without any textual instruction.

We fed both the image and its corresponding caption into the Qwen2.5-VL model and extracted final-layer embeddings. We then computed the average cosine distance between the image tokens and text tokens in the shared semantic space. Results show that the Prompt-in-Image group consistently exhibits smaller cosine distances, with an average reduction of 12%. In Figure 7, we also present PCA visualizations of two sets of test samples. The right plot (Prompt-in-Image) shows a smaller modality gap, where visual and text tokens are more closely aligned. This suggests that Prompt-in-Image acts as a bridge between the two modalities, effectively reducing the gap and enhancing multimodal alignment.

## 6 Conclusion

In this work, we propose Prompt-in-Image, a simple yet effective strategy that embeds textual instructions directly into images to unify the input modality. Through systematic evaluations on three representative VLMs, Qwen2.5-VL, InstructBLIP and LLaVA-1.5, we observe divergent effects: while Prompt-in-Image consistently improves Qwen's performance and reduces hallucination, it significantly degrades InstructBLIP and LLaVA's output quality.

We further analyze this phenomenon and identify key differences between the two models.On the one hand, InstructBLIP and LLaVA (based on CLIP) exhibit excessive attention to embedded text regions, leading to over-reliance on local patterns and results in hallucination. In contrast, Qwen demonstrates stronger robustness. On the other hand, Prompt-in-Image helps Qwen by enhancing modality fusion and mitigating alignment issues. Empirical results confirm that Prompt-in-Image leads to a smaller modality gap and improved cross-modal coherence.

This work shows that how models are trained on multimodal data really matters. It also suggests that simpler, unified approaches to VLM architecture might be worth exploring further.

## Limitations

Our study has several limitations. First and foremost, due to computational resource constraints, we restricted our evaluation to three representative open-source models (Qwen2.5-VL, InstructBLIP, and LLaVA-1.5), all at the 7B parameter scale. While these models cover distinct vision encoding strategies (e.g., CLIP-based vs. non-CLIP), we have not verified whether our findings generalize to significantly larger models (e.g., 34B, 70B+) or proprietary commercial models (e.g., GPT-4o). It remains an open question whether larger models, with their stronger emergent capabilities, would exhibit the same sensitivity to embedded text or possess different mechanisms for handling visual prompts.

## References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Jaemin Cho, Seunghyun Yoon, Ajinkya Kale, Franck Dernoncourt, Trung Bui, and Mohit Bansal. 2022. Fine-grained image captioning with clip reward. *arXiv preprint arXiv:2205.13115*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Preprint*, arXiv:2305.06500.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. 2023. Vision transformers need registers. *arXiv preprint arXiv:2309.16588*.

Xuan Gong, Tianshi Ming, Xinpeng Wang, and Zhihua Wei. 2024. Damro: Dive into the attention mechanism of lvlm to reduce object hallucination. *arXiv preprint arXiv:2410.04514*.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. 2024. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13872–13882.

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*.

Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. 2024. Monkey: Image resolution and text label are important things for large multi-modal models. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26763–26773.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. 2022. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer.

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

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*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Holy Lovenia, Wenliang Dai, Samuel Cahyawijaya, Ziwei Ji, and Pascale Fung. 2023. Negative object presence evaluation (nope) to measure object hallucination in vision-language models. *arXiv preprint arXiv:2310.05338*.

Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-Sheng Hua, and Ji-Rong Wen. 2021. Counterfactual vqa: A cause-effect look at language bias. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12700–12710.

OpenAI. 2024. Gpt-4o: Openai's new flagship model. https://openai.com/index/gpt-4o. Accessed: 2025-05-23.

Ella Rabinovich, Samuel Ackerman, Orna Raz, Eitan Farchi, and Ateret Anaby-Tavor. 2023. Predicting question-answering performance of large language models through semantic consistency. *arXiv preprint arXiv:2311.01152*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

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

François Role, Sébastien Meyer, and Victor Amblard. 2025. Fill the gap: Quantifying and reducing the modality gap in image-text representation learning. *arXiv preprint arXiv:2505.03703*.

Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, Yiming Yang, and 1 others. 2023. Aligning large multimodal models with factually augmented rlhf. *arXiv preprint arXiv:2309.14525*.

Fei Wang, Wenxuan Zhou, James Y Huang, Nan Xu, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2024a. mdpo: Conditional preference optimization for multimodal large language models. *arXiv preprint arXiv:2406.11839*.

Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, and 1 others. 2023. Amber: An llm-free multi-dimensional benchmark for mllms hallucination evaluation. *arXiv preprint arXiv:2311.07397*.

Xintong Wang, Jingheng Pan, Liang Ding, and Chris Biemann. 2024b. Mitigating hallucinations in large vision-language models with instruction contrastive decoding. *arXiv preprint arXiv:2403.18715*.

Zhaochen Wang, Bryan Hooi, Yiwei Wang, Ming-Hsuan Yang, Zi Huang, and Yujun Cai. 2025. Text speaks louder than vision: Ascii art reveals textual biases in vision-language models. *arXiv preprint arXiv:2504.01589*.

Hao Yin, Gunagzong Si, and Zilei Wang. 2025. The mirage of performance gains: Why contrastive decoding fails to address multimodal hallucination. *arXiv preprint arXiv:2504.10020*.

Yudong Zhang, Ruobing Xie, Jiansheng Chen, Xingwu Sun, Yu Wang, and 1 others. 2024. Dhcp: Detecting hallucinations by cross-modal attention pattern in large vision-language models. *arXiv preprint arXiv:2411.18659*.

# A Example Appendix

This is an appendix.