Black-Box Visual Prompt Engineering for Mitigating Object Hallucination in Large Vision Language Models

Sangmin Woo[♡♠*] **Kang Zhou**[♡] **Yun Zhou**[♡] **Shuai Wang**[♡] **Sheng Guan**[♡]
Haibo Ding[♡✉] **Lin Lee Cheong**[♡]
[♡]Amazon AWS AI [♠]KAIST
{sangminw,zhoukang,yunzzhou,wshui,shguan,hbding,lcheong}@amazon.com

Abstract

Large Vision Language Models (LVLMs) often suffer from object hallucination, which undermines their reliability. Surprisingly, we find that simple object-based visual prompting—overlaying visual cues (*e.g.*, bounding box, circle) on images—can significantly mitigate such hallucination; however, different visual prompts (VPs) vary in effectiveness. To address this, we propose **Black-Box Visual Prompt Engineering (BBVPE)**, a framework to identify optimal VPs that enhance LVLM responses without needing access to model internals. Our approach employs a pool of candidate VPs and trains a router model to dynamically select the most effective VP for a given input image. This *black-box* approach is model-agnostic, making it applicable to both open-source and proprietary LVLMs. Evaluations on benchmarks such as POPE and CHAIR demonstrate that BBVPE effectively reduces object hallucination.

1 Introduction

LVLMs (Tong et al., 2024; Bai et al., 2023) demonstrate impressive capabilities but often suffer from object hallucination, where they describe objects not present in the image. Addressing this issue is vital for real-world deployment, particularly in critical areas like healthcare and assistive technologies (Hu et al., 2024; Xu et al., 2024).

Existing methods try to mitigate object hallucination by collecting datasets (Lu et al., 2024), re-training or fine-tuning (Zhao et al., 2023), modifying decoding methods (Leng et al., 2023; Favero et al., 2024; Woo et al., 2024a,b), or using costly feedback loops (Lee et al., 2023). However, they often require access to model internals (*e.g.*, attention, logits), making them impractical for proprietary LVLMs (OpenAI, 2024; Anthropic, 2024).

A promising yet under-explored direction is visual prompting, which overlays visual cues like bounding boxes or circles on images to guide model outputs (Yao et al., 2024; Shtedritski et al., 2023; Yang et al., 2023b,c,a). While visual prompting has shown potential in improving visual grounding (Yang et al., 2023c,a), its role in reducing object hallucination remains unclear. This raises two key questions: **(Q1)** Can visual prompting mitigate object hallucination in LVLMs? **(Q2)** If so, can we systematically learn the optimal VPs?

Our preliminary experiments show that simple object-based VPs can significantly reduce object hallucination. Interestingly, their effectiveness varies across images and is particularly notable in an *Oracle* scenario, where the best-performing VP for each image is assumed to be known. This finding effectively answers Q1 (see Fig. 1) and suggests the need for a systematic method to identify the optimal VP for each image.

To answer Q2, we introduce **BBVPE**, a novel framework designed to systematically identify and apply optimal VPs to reduce object hallucination in LVLMs. Our approach treats LVLMs as "black boxes", relying solely on input-output pairs without modifying the model itself. The framework has three key components: (1) a pool of predefined VPs, (2) a scoring function to evaluate the effectiveness of each prompt, and (3) a router model that dynamically selects the best prompt based on observed input-output behavior. Our method requires no access to model internals, making it applicable to both open-source and proprietary LVLMs.

Our key contributions are: **1)** We find that Oracle VPs exist for images given an LVLM, which, when identified, can greatly reduce object hallucination. **2)** We propose a novel framework, BBVPE, for systematically identifying these optimal VPs. **3)** In standard benchmarks like POPE and CHAIR, our approach significantly reduces object hallucination in both open-source and proprietary LVLMs.

[*]Work done during an internship at Amazon.
[✉]Corresponding author.

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers), pages 529–538

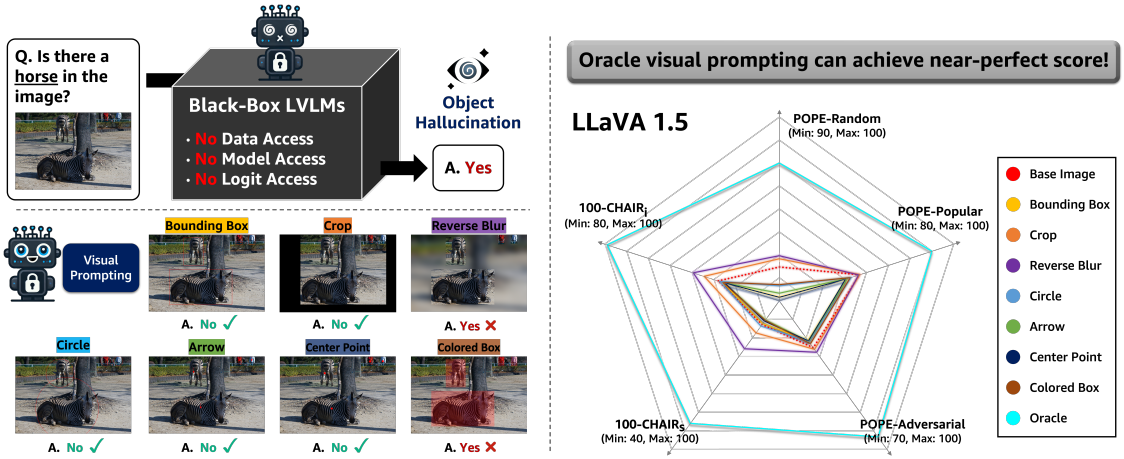

Figure 1: **Motivation. (left)** An LVLM misidentifies a zebra as a horse, demonstrating object hallucination. Various VPs elicit different responses, but their effectiveness depends on the specific characteristics of the image. To remove randomness and solely see the impact of visual prompting, all responses are generated using greedy decoding. **(right)** While most VPs yield comparable performances, an *Oracle*—which adaptively applies the best-performing VP per image—dramatically boosts results.

2 Related Work

Hallucinations in LVLMs. Efforts to address hallucination in LVLMs (Dai et al., 2023; Liu et al., 2023c,b) have focused on three primary areas: *(i) Data.* Improving data quality is a key to reducing hallucinations (Wang et al., 2023), using negative (Liu et al., 2023a) and counterfactual data (Yu et al., 2023), as well as dataset cleansing to reduce noise and errors (Yue et al., 2024). *(ii) Training.* Training-based methods (Jiang et al., 2023; Zhai et al., 2023) utilize supervision from external datasets (Chen et al., 2023), reinforcement learning or preference optimization (Zhao et al., 2023; Gunjal et al., 2024) to better align model outputs with visual content. *(iii) Decoding.* Decoding-based methods (Leng et al., 2023; Favero et al., 2024; Woo et al., 2024b,a) refine generation by incorporating additional guidance into the output probability distribution. Alternatively, post-hoc correction methods (Lee et al., 2023; Wu et al., 2024; Yin et al., 2023) iteratively improve responses through self-feedback loops to identify and correct errors. Most of these approaches assume a *white-box* setting with access to model internals (*e.g.*, data, parameters, prediction logits). In contrast, our work addresses hallucinations in *black-box* scenarios.

Automated Prompt Engineering. Prompt engineering refines input prompts (x) to yield better outputs (y^*) without modifying model parameters (θ). While traditionally a manual process, APE automates this refinement and has been widely applied in LLMs (Shin et al., 2020; Zhou et al., 2022; Pryzant et al., 2023) to improve text prompts. In the vision-language domain, research has also focused on optimizing textual prompts for CLIP (Liu et al., 2024a) or text-to-image diffusion models (Mañas et al., 2024; Liu et al., 2024b). With LLMs evolve into multimodal system, capable of handling both text and visual data, APE's application to visual inputs is still largely unexplored. To our knowledge, this work is the first to extend APE to visual inputs, aiming to reduce hallucinations in LVLMs.

3 Black-Box Visual Prompt Engineering

Applying prompt engineering to the visual domain is challenging due to the vast combinatorial complexity of image space. Also, direct optimization over pixel values risks distorting the semantic content of the images. To circumvent this, we use a discrete selection approach, choosing from a predefined VPs that enhance images without altering their original meaning. A lightweight router model selects the most suitable VP, which is then applied before input to LVLMs, reducing hallucinations. Our black-box approach mitigates hallucinations without accessing internal LVLM values (*e.g.*, attention, logits), making it compatible with proprietary models. An overview is shown in Fig. 2.

Oracle. The Oracle represents an ideal scenario where the optimal VP for each image is known during evaluation, setting an upper bound on performance (see Fig. 1 right). It is equivalent to adaptively selecting the VP with minimal hallucination per image. Our goal is to train the router model to approximate this behavior.

Object localization. To identify relevant objects within an image I, we first utilize an object localization model $\mathcal{L}$. The model detects and outputs a set of object coordinates $O = \{o_1, o_2, \ldots, o_m\}$.

Visual prompt pool. We define a pool of candidate VPs $P = \{p_1, p_2, \ldots, p_n\}$, which includes

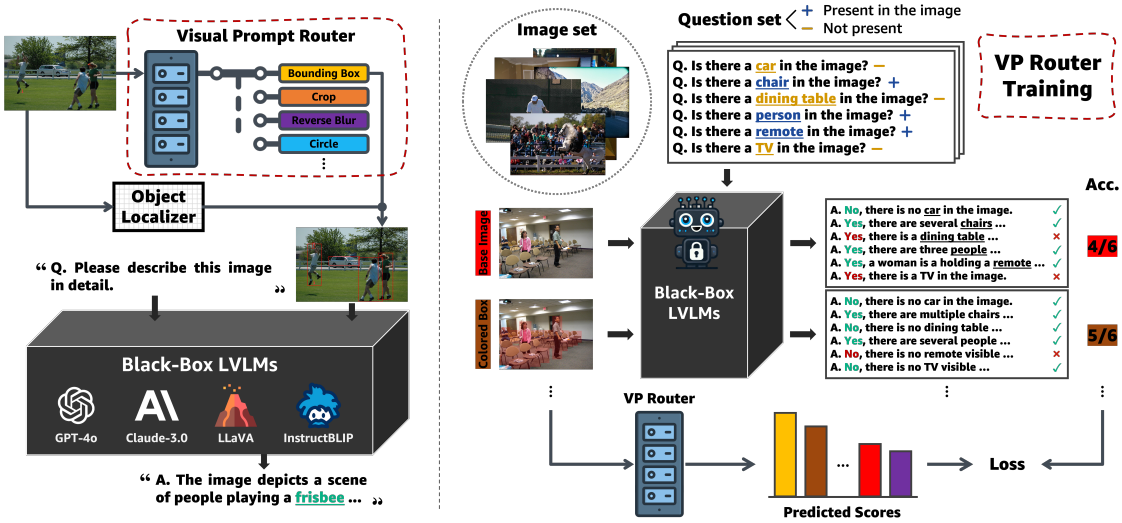

Figure 2: **Overview. (left)** BBVPE utilizes a VP router and object localizer to mitigate object hallucinations in LVLMs. VP router dynamically selects the optimal VP for a given image. **(right)** During its training phase, a set of images with various VPs and a series of object-related questions are posed to the LVLMs. The question set includes both objects that are present and not present in the image. LVLM responses are then evaluated based on accuracy. The VP router predicts scores for each VP, optimizing the selection process to identify the most effective prompt for a given image.

visual markers like circles and arrows. Each VP $p_i \in P$ modifies the image I by highlighting localized objects O, producing I_{p_i}. The image-text pair (I_{p_i}, T), where T is a textual prompt, is then fed into the LVLM $\mathcal{M}$ to produce a response.

Quantifying object hallucination. To evaluate a model's robustness to object hallucination, we define a scoring function S that measures response accuracy regarding object presence:

$$S = \frac{|\text{correct responses}|}{|\text{total presence questions}|} \tag{1}$$

Dataset construction. For a given image I, the optimal VP p^* is chosen to maximize S:

$$p^* = \arg\max_{p_i \in P} S\left(\mathcal{M}\left(I_{p_i}, T\right)\right) \tag{2}$$

To ensure uniqueness, cases where multiple VPs achieve the highest score are excluded. This results in a training dataset D_{train} that maps images to unique optimal prompts, including the option of not applying any VP:

$$D_{\text{train}} = \left\{\left(I_j, p_j^*\right) \mid \text{unique } p_j^*\right\} \tag{3}$$

Training a router model. The router model $\mathcal{R}_\theta$ is trained on D_{train} to predict the optimal VP p^* for a given image I. It assigns a score $\hat{s}_{p_i}$ to each VP:

$$\hat{s}_{p_i} = \mathcal{R}_\theta\left(I, p_i\right) \tag{4}$$

These scores are converted into probabilities via softmax:

$$\hat{P}(p_i \mid I) = \frac{\exp(\hat{s}_{p_i})}{\sum_{p_j \in P} \exp(\hat{s}_{p_j})} \tag{5}$$

The router model is trained using cross-entropy loss between the predicted probability distribution $\hat{P}(p_i \mid I)$ and the one-hot encoded ground-truth optimal VP p^*:

$$\mathcal{L} = -\sum_{p_i \in P} \mathbb{1}_{p_i = p^*} \log \hat{P}(p_i \mid I) \tag{6}$$

The trained router model enables efficient VP selection without directly querying the LVLM.

LVLM inference. At inference, the trained router model $\mathcal{R}_\theta$ predicts the optimal VP $\hat{p}$:

$$\hat{p} = \arg\max_{p_i \in P} \hat{s}_{p_i} \tag{7}$$

Applying $\hat{p}$ to the localized objects O in I produces $I_{\hat{p}}$, which, along with the textual prompt T, is fed into LVLM $\mathcal{M}$ to obtain a response with reduced object hallucination.

4 Experiments

In all tables, *baseline* refers to not using visual prompting. We compare our approach against three baselines: (1) selecting *random VP* for each image, (2) consistently using a fixed *best VP* that delivers the highest overall performance for the model, and (3) an *Oracle* that adaptively selects the optimal VP per image. Responses are generated via greedy decoding to eliminate randomness.[1]

Evaluation setup. We evaluate using POPE (Li et al., 2023) and CHAIR (Rohrbach et al., 2018) on the COCO (Lin et al., 2014) val split. POPE assesses hallucination by asking binary Yes/No

[1]Implementation details are in Appendix A.

Setup	Methods	Open-source LVLMs								Proprietary LVLMs							
		LLaVA 1.5				InstructBLIP				GPT-4o				Claude-3.0-Sonnet			
		Acc. ↑	Prec. ↑	Rec. ↑	F1 ↑	Acc. ↑	Prec. ↑	Rec. ↑	F1 ↑	Acc. ↑	Prec. ↑	Rec. ↑	F1 ↑	Acc. ↑	Prec. ↑	Rec. ↑	F1 ↑
Random	*baseline*	89.60	88.77	90.67	89.71	90.23	92.95	87.07	89.91	87.33	97.95	76.27	85.76	79.93	98.18	61.00	75.25
	random VP	89.46	89.07	89.95	89.51	89.75	91.76	87.35	89.50	87.02	96.63	76.75	85.53	78.91	97.74	59.18	73.71
	best VP†	90.40	90.67	90.07	90.37	89.97	91.89	87.67	89.73	88.07	98.47	77.33	86.63	80.10	97.78	61.60	75.58
	BBVPE	91.37	91.97	91.40	91.42	91.50	90.47	91.44	90.95	88.83	98.71	78.26	87.31	80.84	97.43	63.49	76.88
	Oracle	93.99	95.13	94.69	93.94	94.04	97.16	92.46	93.44	93.50	99.47	87.48	93.09	85.87	99.27	72.27	83.64
Popular	*baseline*	86.20	83.23	90.67	86.79	83.43	81.17	87.07	84.01	86.03	94.56	76.47	84.56	78.43	93.56	61.07	73.90
	random VP	86.20	83.68	89.96	86.70	83.12	80.54	87.35	83.80	85.26	92.38	76.91	83.92	77.48	93.24	59.24	72.44
	best VP†	86.70	84.38	90.07	87.13	84.13	81.88	87.67	84.67	86.37	94.31	77.40	85.02	78.70	93.90	61.60	74.40
	BBVPE	87.23	85.97	90.20	88.03	84.57	82.41	88.71	85.44	87.33	95.31	79.22	86.52	79.67	94.90	62.42	75.30
	Oracle	91.97	92.81	94.69	92.38	88.52	89.65	92.46	89.06	92.57	98.04	86.87	92.12	84.87	96.78	72.13	82.66
Adversarial	*baseline*	79.73	74.40	90.67	81.73	80.73	77.28	87.07	81.88	85.50	93.33	76.47	84.06	77.13	89.82	61.20	72.80
	random VP	79.56	74.48	89.95	81.49	79.87	75.99	87.35	81.27	84.49	90.76	76.85	83.20	75.90	88.83	59.25	71.07
	best VP†	80.30	75.35	90.07	82.05	80.20	76.28	87.67	81.58	85.73	93.07	77.00	84.28	76.90	88.76	61.60	72.73
	BBVPE	81.33	75.84	91.77	83.05	81.23	77.33	88.49	82.53	86.00	92.19	78.67	84.89	78.00	88.89	61.54	72.73
	Oracle	85.62	84.23	94.69	87.25	85.72	85.98	92.46	86.80	91.90	96.94	86.53	91.44	83.53	94.36	71.33	81.25

Table 1: **Results on POPE benchmark.** Our approach consistently outperforms baselines; yet, there is still a large gap compared to *Oracle*. † Best VPs are: `'reverse blur'` for LLaVA and InstructBLIP, `'crop'` for GPT-4o and Claude-3.0-Sonnet.

Methods	Open-source LMMs				Proprietary LMMs			
	LLaVA 1.5		InstructBLIP		GPT-4o		Claude-3.0	
	CH_S↓	CH_I↓	CH_S↓	CH_I↓	CH_S↓	CH_I↓	CH_S↓	CH_I↓
baseline	62.8	18.1	53.6	14.7	44.9	8.0	38.5	12.1
random VP	61.7	18.4	53.7	15.8	45.2	8.0	39.0	13.9
best VP†	56.3	17.0	48.5	14.4	36.5	5.9	33.9	11.4
BBVPE	46.3	14.9	41.5	12.5	32.0	4.9	31.7	10.7
Oracle	27.7	6.4	18.5	3.8	8.4	1.3	7.4	2.0

Table 2: **Results on CHAIR benchmark.** Black-Box VPE significantly reduces hallucinations in image descriptions. † Best VPs are: `'center point'` for LLaVA and InstructBLIP, `'reverse blur'` for GPT-4o, and `'arrow'` for Claude-3.0-Sonnet.

Methods	LLaVA 1.5					
	Acc ↑	Det ↑	Com ↑	Rel ↑	Rob ↑	Total ↑
baseline	7.08	6.63	6.67	7.35	7.51	35.24
random VP	6.38	6.21	6.25	6.85	6.84	32.52
best VP†	6.53	6.30	6.34	6.92	6.92	33.00
BBVPE	7.24	6.86	6.95	7.63	7.70	36.38
Oracle	7.59	7.27	7.30	8.03	8.10	38.29

Table 3: **Comprehensive image description evaluation by GPT-4o.** LLaVA is assessed based on 5 criteria: Accuracy, Detail, Comprehensiveness, Relevance, and Robustness. † The best VP is `'center point'`.

questions like "Is there a [object] in the image?" across various prompt setups (Random, Popular, and Adversarial). CHAIR measures the ratio of hallucinated objects in image descriptions, with two variants: CH_S (per sentence) and CH_I (per object), where lower scores indicate fewer hallucinations. Additionally, we use GPT-4o (OpenAI, 2024) for a more comprehensive evaluation.[2]

Model instantiation. While our framework is generic, we instantiate the components as follows:

- **Object Localizer** $\mathcal{L}$: SAM 2 (Ravi et al., 2024).
- **VP Router** $\mathcal{R}_\theta$: Frozen CLIP vision encoder (Radford et al., 2021) with a trainable MLP.
- **LVLMs** $\mathcal{M}$: We use two open-source models (LLaVA-1.5, InstructBLIP) and two proprietary models (GPT-4o, Claude-3.0-Sonnet).

During router training, all other model components are kept frozen.

4.1 Evaluation Results

POPE benchmark. Table 1 shows BBVPE consistently outperforms baselines across most metrics, prompt setups, and LVLMs. While *random VP* may not improve results over *baseline* (No VP applied), *best VP* generally performs better. BBVPE further enhances performance by properly routing the optimal VP for each image, though a gap remains to *Oracle*, suggesting room for improvement.

CHAIR benchmark. As shown in Table 2, BBVPE significantly reduces object hallucinations in image descriptions at both instance (CH_I) and sentence (CH_S) levels across all LVLMs, though still below *Oracle* performance. While *random VP* often underperforms *baseline*, *best VP* consistently improves results, with BBVPE further enhancing performance.

GPT-4o evaluation. Table 3 shows GPT-4o's evaluation of image descriptions from LLaVA 1.5, scored from 0 to 10. GPT-4o receives the image and the generated descriptions, scoring each based on 5 criteria.[3] While naive visual prompting (*random VP*, *best VP*) degrade performance, BBVPE effectively improves scores. Notably, applying a fixed *best VP* to all images performs even worse than using no VP (*baseline*), but BBVPE outperforms both by optimally selecting VPs per image.

4.2 Key Observations

(1) Different LVLMs favor different VPs. For example, `'reverse blur'` and `'crop'` generally

[2]More details about evaluation setup are in Appendix B.

[3]Details on GPT-4o instruction are in Appendix C.

Methods	Latency (ms/token)	TFLOPs	
Baseline (LLaVA-1.5)	43.664	9.726	-
+ VCD (Liu et al., 2023a)	111.392	19.452	✗
+ M3ID (Favero et al., 2024)	84.49	19.452	✗
+ RITUAL (Woo et al., 2024a)	88.582	19.452	✗
+ AvisC (Woo et al., 2024b)	88.127	19.452	✗
+ OPERA (Huang et al., 2023)	159.615	48.628	✗
+ VOLCANO (Lee et al., 2023)	202.122	42.794	✗
+ BBVPE (Ours)	65.505	16.968	✓

Table 4: **Comparison of methods on latency, TFLOPs, and applicability to black-box LVLMs ( ).** All runs use a single NVIDIA A100 40GB GPU.

Methods (Model: LLaVA-1.5)	Random		Popular		Adversarial	
	Acc.	F1	Acc.	F1	Acc.	F1
baseline	81.23	83.16	72.43	77.31	69.07	75.37
random VP	80.97	82.95	72.07	77.00	68.70	74.94
best VP (reverse blur)	82.10	83.99	73.27	78.02	69.43	75.43
BBVPE (train dataset → test dataset)						
GQA → GQA	**83.47**	**84.89**	**74.37**	**78.56**	**71.73**	**76.87**
COCO → GQA	**82.73**	**84.17**	**73.83**	**78.28**	**70.30**	**75.90**
Oracle	92.93	93.05	82.27	84.00	76.87	80.21

Table 5: **Results on POPE benchmark using GQA dataset (Hudson and Manning, 2019).** Here, we also compare with cross-dataset evaluation setup (COCO → GQA).

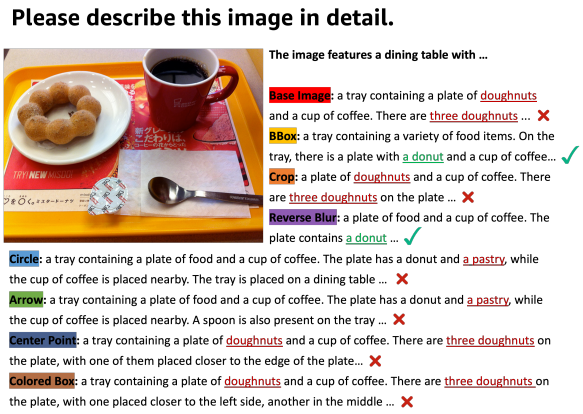

Figure 3: **Impact of different VPs on image description generation.** Different VPs produce varied results, but not all are equally effective. All responses are generated using greedy decoding to eliminate randomness and focus solely on the influence of visual prompting.

work well for LLaVA 1.5 (Fig. 1 (Right)).
(2) Surprisingly, proprietary LVLMs underperform compared to open-source LVLMs on POPE in terms of Accuracy and F1 score (Table 1). Proprietary LVLMs are cautious to say "yes"—indicated by high precision but low recall. It suggests a conservative response strategy, likely due to policy restrictions aimed at minimizing false positives.
(3) No single VP achieves optimal results across all LVLMs and metrics; the best VP varies by model and metric. (Tables 1 to 3)
(4) Learning an effective routing of VPs can significantly reduce hallucinations (Tables 1 to 3).

4.3 Analysis

Computational cost. We analyze the latency and computational overhead (TFLOPs) of recent methods for object hallucination mitigation in Table 4. VCD (Liu et al., 2023a), M3ID (Favero et al., 2024), RITUAL (Woo et al., 2024a), and AvisC (Woo et al., 2024b) require two forward passes, while OPERA (Huang et al., 2023) uses beam search with rollbacks, and VOLCANO (Lee et al., 2023) performs critique-revise-decide steps, needing three forward passes. BBVPE introduces some additional latency due to the use of an object localizer (*e.g.*, SAM2) and VP router (*e.g.*, CLIP+MLP). However, it is significantly more efficient than other methods. Unlike others relying on model internals (*e.g.*, weights, logits), BBVPE operates in a black-box manner, making it applicable to both open-source and proprietary models.

Cross-dataset evaluation on POPE-GQA benchmark. Table 5 shows the results on POPE benchmark using GQA dataset. The overall performance trends are similar to the LLaVA-1.5 results in Table 1. Notably, the VP router trained on COCO performs effective VP selection even on unseen datasets like GQA, outperforming a fixed best VP and achieving results comparable to a VP router trained and tested on GQA. This demonstrates BBVPE's potential for cross-dataset generalization.

Visual prompting for image description generation. Fig. 3 analyzes the impact of VPs on image descriptions. While certain VPs, such as Bounding Box and Reverse Blur, enable the model to accurately identify existing items, others introduce errors by mentioning additional pastries or multiple donuts. This again confirms the variability in VPs' effectiveness and underscores the importance of selecting the right VP to mitigate hallucination.

5 Conclusion

In this work, we proposed **BBVPE** framework to systematically identify optimal VPs that mitigate object hallucinations in LVLMs. Our findings confirm that: **(A1)** carefully curated visual prompting can effectively reduce hallucinations in LVLMs, and **(A2)** optimal VPs can be systematically learned in a *black-box* setup. By dynamically selecting the most suitable VP from a predefined pool, guided by a trained router model based on LVLM preferences, our framework significantly enhances the performance of both open-source and proprietary LVLMs on hallucination benchmarks.

Limitations & Future Work

(1) Our current approach primarily focuses on natural images and does not extend to abstract and synthetic figures, such as those used in document VQA (Mathew et al., 2021), science VQA (Lu et al., 2022), or math VQA (Lu et al., 2023). The current design of our method may not be directly applicable to these synthetic images, which typically exhibit different visual characteristics.

(2) We currently use bounding box-based prompts from the Segment Anything Model (Kirillov et al., 2023). Transitioning to fine-grained, mask-based VPs could potentially enhance performance, as demonstrated in recent studies (Yang et al., 2023a,b).

(3) Our router model currently considers only image features and does not incorporate the question context. Our preliminary experiments suggest that incorporating question context could further improve results, pointing toward future work on exploring question-aware visual prompting.

(4) To simplify optimization, we focus on object-level visual prompting, but extending to patch-based or pixel-based VPs could potentially provide a richer set of design space.

(5) Exploring the synergy between visual and textual prompt optimization remains an open research direction that may offer valuable insights.

(6) While our method is specifically designed to address object hallucination, exploring how VP and our framework perform in addressing attribute and relation hallucination remains an intriguing challenge that we leave for future work.

(7) Object localization matters. We observed that better localization, such as using ground truth object coordinates, leads to improved results in our preliminary results.

(8) During router model training, we observed sensitivity to hyperparameters and occasional convergence instability, sometimes leading to overfitting. This highlights the subtle learning signal from LVLM preferences over VPs, requiring a carefully designed training process.

Despite these limitations, to the best of our knowledge, our study is the first black-box approach for mitigating object hallucination in LVLMs. We hope that our initial investigation into automated visual prompt engineering and black-box strategies inspires further research into broader vision-language challenges beyond object hallucination.

Ethical Considerations

In our current method, we use a predefined pool of VPs and have not observed any jail-breaking phenomena with visual prompting. However, we are uncertain whether more fine-grained visual prompt engineering, such as using diffusion models, could lead to adversarial attacks or jail-breaking scenarios. Rigorous testing is needed to ensure the robustness and safety of this approach. Further research should address these considerations, if present, and focus on identifying and mitigating potential risks associated with VP misuse.

References

Anthropic. 2024. Claude 3.0. Accessed: 2024-09-17.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond.

Zhiyang Chen, Yousong Zhu, Yufei Zhan, Zhaowen Li, Chaoyang Zhao, Jinqiao Wang, and Ming Tang. 2023. Mitigating hallucination in visual language models with visual supervision. *arXiv preprint arXiv:2311.16479*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36.

Alessandro Favero, Luca Zancato, Matthew Trager, Siddharth Choudhary, Pramuditha Perera, Alessandro Achille, Ashwin Swaminathan, and Stefano Soatto. 2024. Multi-modal hallucination control by visual information grounding. *arXiv preprint arXiv:2403.14003*.

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.

Yutao Hu, Tianbin Li, Quanfeng Lu, Wenqi Shao, Junjun He, Yu Qiao, and Ping Luo. 2024. Omnimed-vqa: A new large-scale comprehensive evaluation benchmark for medical lvlm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22170–22183.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. 2023. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. *arXiv preprint arXiv:2311.17911*.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Chaoya Jiang, Haiyang Xu, Mengfan Dong, Jiaxing Chen, Wei Ye, Ming Yan, Qinghao Ye, Ji Zhang, Fei Huang, and Shikun Zhang. 2023. Hallucination augmented contrastive learning for multimodal large language model. *arXiv preprint arXiv:2312.06968*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026.

Seongyun Lee, Sue Hyun Park, Yongrae Jo, and Minjoon Seo. 2023. Volcano: mitigating multimodal hallucination through self-feedback guided revision. *arXiv preprint arXiv:2311.07362*.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. 2023. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. *arXiv preprint arXiv:2311.16922*.

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*.

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. In *The Twelfth International Conference on Learning Representations*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023b. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*.

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

Shihong Liu, Samuel Yu, Zhiqiu Lin, Deepak Pathak, and Deva Ramanan. 2024a. Language models as black-box optimizers for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12687–12697.

Yilun Liu, Minggui He, Feiyu Yao, Yuhe Ji, Shimin Tao, Jingzhou Du, Duan Li, Jian Gao, Li Zhang, Hao Yang, et al. 2024b. What do you want? user-centric prompt generation for text-to-image synthesis via multi-turn guidance. *arXiv preprint arXiv:2408.12910*.

Jiaying Lu, Jinmeng Rao, Kezhen Chen, Xiaoyuan Guo, Yawen Zhang, Baochen Sun, Carl Yang, and Jie Yang. 2024. Evaluation and enhancement of semantic grounding in large vision-language models. In *AAAI-ReLM Workshop*.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2023. Mathvista: Evaluating math reasoning in visual contexts with gpt-4v, bard, and other large multimodal models. *arXiv e-prints*, pages arXiv–2310.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Oscar Mañas, Pietro Astolfi, Melissa Hall, Candace Ross, Jack Urbanek, Adina Williams, Aishwarya Agrawal, Adriana Romero-Soriano, and Michal Drozdzal. 2024. Improving text-to-image consistency via automatic prompt optimization. *arXiv preprint arXiv:2403.17804*.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. 2021. Docvqa: A dataset for vqa on document images. *arXiv preprint arXiv:2007.00398*.

OpenAI. 2024. GPT-4o system card.

Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. 2023. Automatic prompt optimization with" gradient descent" and beam search. *arXiv preprint arXiv:2305.03495*.

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

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. 2024. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*.

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

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with

automatically generated prompts. *arXiv preprint arXiv:2010.15980*.

Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. 2023. What does clip know about a red circle? visual prompt engineering for vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11987–11997.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. 2024. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*.

Bin Wang, Fan Wu, Xiao Han, Jiahui Peng, Huaping Zhong, Pan Zhang, Xiaoyi Dong, Weijia Li, Wei Li, Jiaqi Wang, et al. 2023. Vigc: Visual instruction generation and correction. *arXiv preprint arXiv:2308.12714*.

Sangmin Woo, Jaehyuk Jang, Donguk Kim, Yubin Choi, and Changick Kim. 2024a. Ritual: Random image transformations as a universal anti-hallucination lever in lvlms. *arXiv preprint arXiv:2405.17821*.

Sangmin Woo, Donguk Kim, Jaehyuk Jang, Yubin Choi, and Changick Kim. 2024b. Don't miss the forest for the trees: Attentional vision calibration for large vision language models. *arXiv preprint arXiv:2405.17820*.

Junfei Wu, Qiang Liu, Ding Wang, Jinghao Zhang, Shu Wu, Liang Wang, and Tieniu Tan. 2024. Logical closed loop: Uncovering object hallucinations in large vision-language models. *arXiv preprint arXiv:2402.11622*.

Dexuan Xu, Yanyuan Chen, Jieyi Wang, Yue Huang, Hanpin Wang, Zhi Jin, Hongxing Wang, Weihua Yue, Jing He, Hang Li, et al. 2024. Mlevlm: Improve multi-level progressive capabilities based on multimodal large language model for medical visual question answering. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 4977–4997.

Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. 2023a. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*.

Lingfeng Yang, Yueze Wang, Xiang Li, Xinlong Wang, and Jian Yang. 2023b. Fine-grained visual prompting. *Advances in Neural Information Processing Systems*, 36.

Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. 2023c. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv preprint arXiv:2309.17421*, 9(1):1.

Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. 2024. Cpt: Colorful prompt tuning for pre-trained vision-language models. *AI Open*, 5:30–38.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Tong Xu, Hao Wang, Dianbo Sui, Yunhang Shen, Ke Li, Xing Sun, and Enhong Chen. 2023. Woodpecker: Hallucination correction for multimodal large language models. *arXiv preprint arXiv:2310.16045*.

Qifan Yu, Juncheng Li, Longhui Wei, Liang Pang, Wentao Ye, Bosheng Qin, Siliang Tang, Qi Tian, and Yueting Zhuang. 2023. Hallucidoctor: Mitigating hallucinatory toxicity in visual instruction data. *arXiv preprint arXiv:2311.13614*.

Zihao Yue, Liang Zhang, and Qin Jin. 2024. Less is more: Mitigating multimodal hallucination from an eos decision perspective. *arXiv preprint arXiv:2402.14545*.

Bohan Zhai, Shijia Yang, Chenfeng Xu, Sheng Shen, Kurt Keutzer, Chunyuan Li, and Manling Li. 2023. Halle-control: Controlling object hallucination in large multimodal models. *arXiv preprint arXiv:2310.01779*.

Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. 2023. Beyond hallucinations: Enhancing lvlms through hallucination-aware direct preference optimization. *arXiv preprint arXiv:2311.16839*.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large language models are human-level prompt engineers. *arXiv preprint arXiv:2211.01910*.

Appendix

A Implementation Details

We use a frozen CLIP-ViT-L/14@336px[4] model with a trainable MLP head as our VP router. The router is trained on the COCO dataset (Lin et al., 2014) training split, where each image is paired with 6 questions: 3 positive (about objects present in the image) and 3 negative (about objects not present in the image), following the POPE protocol (Li et al., 2023). Each VP router is individually trained for each LVLM, as the preference for VPs varies across models, and we observed that these preferences do not transfer between models. The training configuration is outlined below.

config	value
image size	336×336
optimizer	AdamW
learning rate	1e-4
loss function	cross entropy loss
training epochs	20

Table 6: **Training configurations for the router model.**

For the object localizer, we use Segment Anything Model 2 (`sam2-hiera-large`)[5]. For LVLMs, we use two open-source models, LLaVA-1.5-7b[6] and InstructBLIP-vicuna-7b[7], and two proprietary models, GPT-4o (`gpt-4o-2024-08-06`)[8] and Claude-3.0-Sonnet (`claude-3-sonnet-20240229`)[9].

B More Details on Evaluation Setup

Benchmarks. We evaluate object hallucinations in LVLMs through discriminative and descriptive tasks on the COCO (Lin et al., 2014) validation split, using the POPE and CHAIR benchmarks, respectively.

(1) POPE (Li et al., 2023) frames hallucination assessment as a binary classification task, asking yes/no questions about the presence of both real and nonexistent objects in an image (*e.g.*, "*Is there a/an* `[OBJECT]` *in the image?*"). Questions for real objects are randomly selected from the actual objects present in the image. There are three prompt setups for selecting nonexistent objects:

- Random: Nonexistent objects are randomly selected from all object categories.
- Popular: Nonexistent objects are chosen from top-k most frequent objects in the dataset.
- Adversarial: Objects are chosen based on frequent co-occurrences with actual objects but are absent from the image.

We use Accuracy, Precision, Recall, and F1 score as evaluation metrics. Accuracy reflects the proportion of correctly answered questions. Precision and Recall indicate the correctness of "Yes" and "No" answers, respectively. F1 score is a harmonic mean of Precision and Recall.

(2) CHAIR (Rohrbach et al., 2018) evaluates the proportion of words in captions that correspond to actual objects in an image, based on ground-truth captions and object annotations. The metric has two variants:

- Per-sentence (CH_S): Proportion of sentences containing hallucinated objects, calculated as $CH_S = \frac{|\text{\# sentences with hallucinated objects}|}{|\text{\# all sentences}|}$.
- Per-instance (CH_I): Proportion of hallucinated objects relative to all mentioned objects, calculated as $CH_I = \frac{|\text{\# hallucinated objects}|}{|\text{\# all objects mentioned}|}$.

Captions are generated with the prompt, "*Please describe this image in detail.*" for evaluation.

C Instruction for GPT-4o Evaluation

Fig. 4 shows the instruction given to GPT-4o for evaluating 8 textual image descriptions of an image, based on 5 criteria: Accuracy, Detail, Comprehensiveness, Relevance, and Robustness. Each criterion is scored on a scale from 1 to 10, with higher scores reflecting better performance. Total scores are calculated for each description to evaluate their overall quality.

[4] https://huggingface.co/openai/clip-vit-large-patch14-336
[5] https://huggingface.co/facebook/sam2-hiera-large
[6] https://huggingface.co/liuhaotian/llava-v1.5-7b
[7] https://huggingface.co/Salesforce/instructblip-vicuna-7b
[8] https://platform.openai.com/docs/models
[9] https://docs.anthropic.com/en/docs/about-claude/models

Image Description Quality Assessment using GPT-4o

```
<SYSTEM_MESSAGE>
You are an expert in image description evaluation. Your task is to assess how well textual
descriptions capture the detailed visual information of images.

<INSTRUCTION>
Compare and evaluate the following 8 descriptions of the provided image.

Descriptions:
{description 1}
{description 2}
...
{description 7}
{description 8}

For each description, rate a score on a scale of 1 to 10, where a higher score indicates better
performance, for each of the 5 criteria:
1. Accuracy: How precisely does the description reflect the actual objects, details, and
attributes (such as color, shape, and number of objects) visible in the image?
2. Detail: How thoroughly does the description capture visual details of the objects, including
finer elements like positions, relative sizes, and relationships?
3. Comprehensiveness: How well does the description cover all key elements of the image, without
omitting important objects or details?
4. Relevance: Does the description focus on significant and pertinent details from the image. The
score decreases if the description includes unnecessary or unrelated information that distracts
from the core details of the image.
5. Robustness: Does the description avoid mentioning any objects or attributes that are not
present in the image? Descriptions without any false information score higher. If nonexistent
elements are included, the score decreases.

Only provide the numerical scores for each criterion and the total score, formatted as follows:
1. Accuracy: score1 | score2 | score3 | score4 | score5 | score6 | score7 | score8
2. Detail: score1 | score2 | score3 | score4 | score5 | score6 | score7 | score8
3. Comprehensiveness: score1 | score2 | score3 | score4 | score5 | score6 | score7 | score8
4. Relevance: score1 | score2 | score3 | score4 | score5 | score6 | score7 | score8
5. Robustness: score1 | score2 | score3 | score4 | score5 | score6 | score7 | score8
Total Score: total1 | total2 | total3 | total4 | total5 | total6 | total7 | total8
```

Figure 4: **GPT-4o evaluation instruction.**