# VISION-DRIVEN PREFERENCE SYNTHESIS FOR MITIGATING HALLUCINATIONS IN VLMS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Vision-Language Models (VLMs) have shown strong performance in visual understanding, yet they still suffer from hallucinations, generating content that is not grounded in the image. Preference alignment is a promising approach to improve visual faithfulness, but its success depends heavily on how preference pairs are constructed. Existing methods exhibit two key limitations; (a) intervention-based methods often introduce significant deviation from the policy distribution, and (b) sampling-based methods often underuse visual information during the construction. In this paper, we propose ViPSy (**Vi**sion-driven **P**reference **Sy**nthesis), a framework for constructing preference data that are both policy-aligned and visually grounded. Our framework consists of two stages; in the first stage, ViPSy derives a visual cue from recurring object-level content across semantically aligned image variants, so preference construction can rely on visual information rather than language priors. In the second stage, ViPSy conditions the policy's own rollouts on this cue, allowing candidates to be guided by visually grounded content while staying close to the policy's response distribution. The resulting candidates remain close to the policy's response distribution while better leveraging visual information from the image. Experiments show that the resulting VLM, preference-aligned with ViPSy-constructed preference pairs, achieves a new state-of-the-art in hallucination mitigation. Compared with the previous state-of-the-art method, it reduces hallucination rates on AMBER and Object HalBench by 35.7% and 24.5%, respectively. The resulting model further improves on general visual grounding benchmarks, e.g., MMStar, MMVP, and CV-Bench, while also yielding gains in semantic segmentation and ImageNet linear probing, underscoring the effectiveness of our framework in enhancing the model's visual capabilities.

## 1 INTRODUCTION

Recent progress in Vision-Language Models (VLMs) (Liu et al., 2024a; Alayrac et al., 2022; Dai et al., 2023; Bai et al., 2025; OpenAI, 2023) has led to notable improvements in visual understanding. By combining pretrained vision encoders (Radford et al., 2021; Zhai et al., 2023) with Large Language Models (LLMs) (Chiang et al., 2023; Hui et al., 2024), VLMs can jointly process images and text, supporting vision-language applications across diverse domains, including education (Lu et al., 2023; Lee et al., 2024b), medical applications (Li et al., 2023; Luo et al., 2023; Moor et al., 2023), industrial inspection (Jeong et al., 2023; Gu et al., 2024; Jiang et al., 2024b), and robotics (Xiong et al., 2024; Liu et al., 2024b; Li et al., 2024). Despite these capabilities, VLMs still suffer from a fundamental limitation known as *hallucination* (Gunjal et al., 2024; Guan et al., 2024), *i.e.*, generating content inconsistent with the given image, often introducing objects that do not actually exist. This limitation raises concerns about the reliability of VLMs in real-world applications, making hallucinations a critical challenge to address.

Hallucination in VLMs often arises from imbalances between the language and vision modalities, where strong language priors make models rely on text instead of the image (Wu et al., 2022; Han et al., 2022; Zhibo et al., 2023; Lee et al., 2024a; Leng et al., 2024; Cheng et al., 2026). Hallucination is also associated with statistical biases in pre-trained data (Agrawal et al., 2016; Goyal et al., 2017; Agarwal et al., 2020; Huang et al., 2025), which can cause models to favor frequent but visually unsupported associations. To address these issues, preference-based alignment (Zhao et al., 2023b; Peng et al., 2025; Yang et al., 2025; Zhang et al., 2026a) has emerged as a promising direction for



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

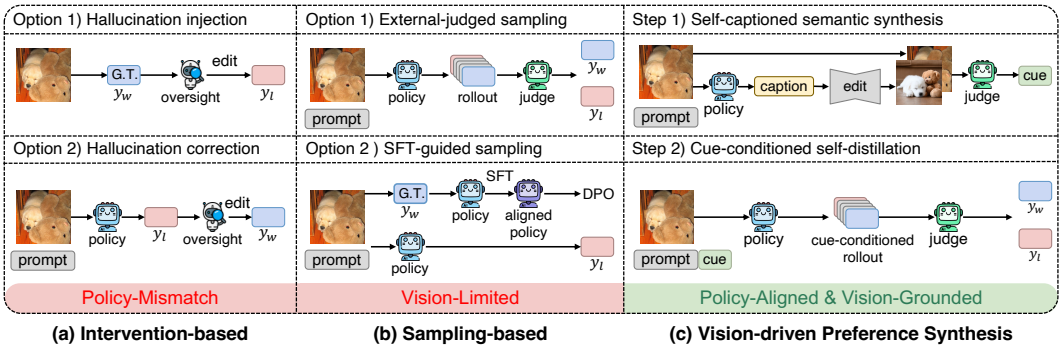


Figure 1: Comparison of preference construction strategies for hallucination mitigation in VLMs. (a) Intervention-based methods can create preference pairs that deviate from the policy distribution. (b) Sampling-based methods reduce this mismatch, but underuse visual information. (c) ViPSy addresses both issues through self-captioned semantic synthesis for consistent cue extraction and cue-conditioned self-distillation for preference synthesis.

improving the visual faithfulness of VLMs. In particular, hallucination mitigation can be naturally formulated as a preference optimization, where hallucination-free responses are treated as preferred outputs over hallucinated ones under the same image and prompt. For hallucination mitigation in VLMs, Direct Preference Optimization (DPO) (Rafailov et al., 2023) has become a practical choice for preference alignment, as it directly optimizes the model from preference pairs without explicit reward modeling or online rollouts. By avoiding these requirements, DPO sidesteps the challenges of reinforcement learning (RL), which updates a policy to maximize preference-derived rewards (Ziegler et al., 2019; Stiennon et al., 2020; Ouyang et al., 2022; Bai et al., 2022a). Specifically, reliable image-grounded rewards are difficult to define due to vision-language misalignment (Sun et al., 2024), and RL-based methods further require reward modeling and repeated online rollouts (Rafailov et al., 2023; Casper et al., 2023; Sidahmed et al., 2024; Li et al., 2025; Hou et al., 2026).

Nevertheless, the effectiveness of DPO depends on the quality of the constructed preference pairs (Ivison et al., 2024; Pan et al., 2025). As shown in Figure 1, existing methods for hallucination mitigation differ mainly in how they construct such pairs and can be grouped into (a) intervention-based and (b) sampling-based methods. Intervention-based methods include hallucination injection methods (Zhou et al., 2024; Sarkar et al., 2024; Xu, 2026), which insert hallucinated content into correct responses, and hallucination correction methods (Sun et al., 2024; Zhao et al., 2023b; Xiao et al., 2025), which revise model-generated responses. Although these methods provide hallucinated and hallucination-free pairs, external intervention can introduce distribution mismatch with the policy. Sampling-based methods instead reduce distribution mismatch by constructing preference data closer to the policy (Yu et al., 2024b; Yang et al., 2025; Peng et al., 2025). External-judged sampling selects preferred and dispreferred responses from policy rollouts using a judge (Yu et al., 2024b; Peng et al., 2025; Yu et al., 2026). Supervised fine-tuning (SFT)-guided sampling first performs SFT on hallucination-free responses and then aligns the resulting model with pairs that favor SFT-aligned outputs over pre-SFT outputs (Yang et al., 2025). More recently, Lim et al. (2026) and Zhang et al. (2026b) support policy-aligned preference construction through self-correction and modified visual inputs, while Zhao & Li (2025) uses external reflective hints to guide policy regeneration. Although policy-aligned preference data can improve preference optimization (Agarwal et al., 2024; Yang et al., 2024; Zhao et al., 2026), existing methods often underuse visual information during preference construction. A discussion of related work on preference alignment, hallucination mitigation, and policy-aligned self-generation is provided in Appendix B.

**Contributions.** In this paper, we propose ViPSy (**Vi**sion-driven **P**reference **Sy**nthesis), a framework for hallucination mitigation that constructs visually grounded preference pairs while keeping them close to the policy's response distribution. The framework addresses two challenges in preference construction: limited use of visual information and distribution mismatch with the policy. First, we introduce *self-captioned semantic synthesis* for consistent cue extraction. Using the policy's own self-captions as T2I prompts, we synthesize semantically aligned image variants. Comparing these variants with the original image reveals recurring object-level evidence, which is aggregated

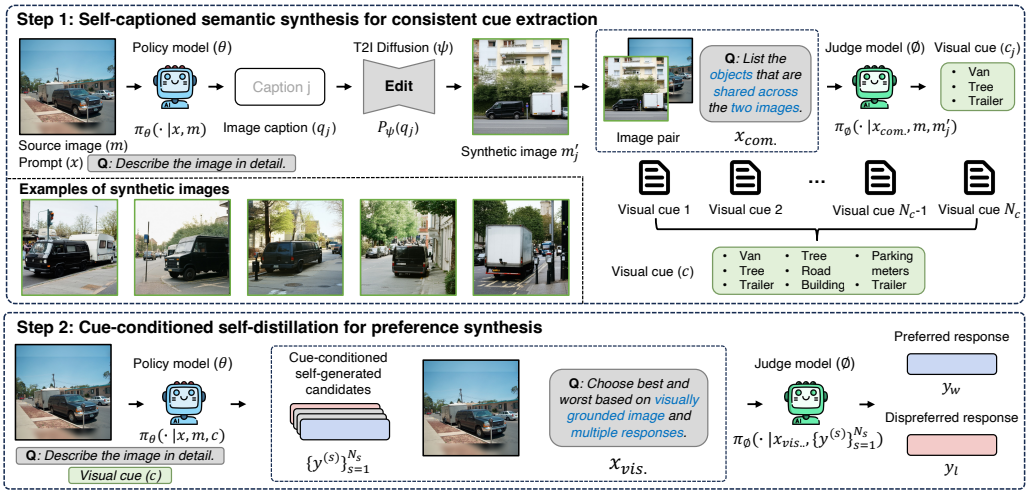


Figure 2: **Overview of ViPSy.** ViPSy constructs visually grounded and policy-aligned preference pairs through two stages. (1) *Self-captioned semantic synthesis for consistent cue extraction* uses the policy's own self-captions as T2I prompts to generate semantically aligned synthetic variants. Each variant is compared with the original image, and recurring object-level evidence across comparisons is aggregated into a consistent visual cue. (2) *Cue-conditioned self-distillation for preference synthesis* uses the consistent visual cue to guide the policy's own rollouts, producing cue-conditioned self-generated candidate responses. A judge model then selects preferred and dispreferred responses according to their visual grounding with respect to the original image.

into a consistent visual cue. This cue provides explicit visual guidance for preference construction, reducing reliance on language priors. Second, we introduce *cue-conditioned self-distillation* for preference synthesis. The consistent visual cue guides the policy's own rollouts, producing cue-conditioned self-generated candidate responses. We then select visually grounded and less grounded responses from these candidates. Thus, the cue encourages visual grounding, while the policy's own rollouts keep the resulting pairs close to its distribution.

Experimental results show that ViPSy enables a new state-of-the-art in mitigating hallucinations of VLMs when used for DPO. Compared to the previous state-of-the-art results, our method significantly reduces AMBER CHAIR and Object HalBench response-level hallucination by 35.7% and 24.5%, respectively. Interestingly, we observe that our alignment scheme further improves general vision-language benchmarks that require visual grounding, with gains of 6.0% on MMStar, 31.3% on MMVP, and 4.0% on CV-Bench. Moreover, the optimized vision encoder shows gains in patch-level segmentation probing and ImageNet linear probing, suggesting improvements in both response-level visual faithfulness and visual representations.

## 2 ViPSy: Vision-driven Preference Synthesis

We consider a vision-language model that generates a response $\boldsymbol{y}$ conditioned on a prompt $\boldsymbol{x}$ and an input image $\boldsymbol{m}$, given a policy $\pi_{\boldsymbol{\theta}}$ parameterized by $\boldsymbol{\theta}$:

$$\pi_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{m}) = \prod_{t=1}^{T} \pi_{\boldsymbol{\theta}}(\boldsymbol{y}_t \mid \boldsymbol{x}, \boldsymbol{m}, \boldsymbol{y}_{<t}). \tag{1}$$

Preference construction for hallucination mitigation requires pairs that contrast visually faithful responses with image-unsupported ones. Intervention-based methods create such contrasts by externally modifying responses, but this can move the data away from the policy distribution. Sampling from the policy reduces this mismatch; however, even when the policy is conditioned on the image $\boldsymbol{m}$ in Eq. 1, its rollouts may still be driven by language priors or modality biases rather than image-specific evidence. We therefore aim to construct preference pairs that remain close to the policy distribution while using object-level visual cues extracted from the image as additional conditioning for the policy's own rollouts.

As summarized in Figure 2, ViPSy achieves this goal by extracting a consistent visual cue from semantically aligned image variations, as detailed in Section 2.1. The key intuition is that visual content present in the original image should recur across synthetic images generated from diverse captions describing the same image. This consistent visual cue is then used to guide cue-conditioned self-rollouts from the policy, as described in Section 2.2. Rather than constructing preferences from externally generated responses, we sample cue-conditioned self-generated candidate responses from the policy itself. Thus, the visual cue encourages visual grounding, while the policy's own rollouts keep the candidates close to its response distribution.

### 2.1 Self-captioned semantic synthesis for visual cue extraction

To construct semantically aligned synthetic images, we use the policy's captions of images as T2I prompts. Concretely, we define the captioning prompt as $\boldsymbol{x}_{\text{cap.}}$ = "*Describe the image in detail.*". We sample $N_c$ captions and synthesize their corresponding images using a T2I diffusion model $P_\psi$:

$$\{\boldsymbol{q}_j\}_{j=1}^{N_c} \sim \pi_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{x}_{\text{cap.}}, \boldsymbol{m}), \qquad \{\boldsymbol{m}'_j\}_{j=1}^{N_c} = \{P_\psi(\boldsymbol{q}_j)\}_{j=1}^{N_c}. \tag{2}$$

Although the sampled captions are grounded in the same image, they emphasize different aspects of the image. Since these images share the same underlying semantic content, comparing them against the original image helps identify image-supported evidence.

**Visual cue extraction.** From the self-captioned synthetic images $\boldsymbol{m}'_j$, we extract a pairwise cue $\boldsymbol{c}_j$ for each $\boldsymbol{m}'_j$ by comparing it with the original image. Specifically, for the synthetic image indexed by $j$, the judge model extracts object-level visual evidence that is consistent between the original image $\boldsymbol{m}$ and the corresponding synthetic image $\boldsymbol{m}'_j$:

$$\boldsymbol{c}_j = \pi_{\boldsymbol{\phi}}(\boldsymbol{x}_{\text{comp.}}, \boldsymbol{m}, \boldsymbol{m}'_j), \tag{3}$$

where $\boldsymbol{\phi}$ denotes a judge model and $\boldsymbol{x}_{\text{comp.}}$ is the pairwise comparison prompt provided in Appendix O. The pairwise cues are then aggregated into a consistent visual cue:

$$\boldsymbol{c} = \pi_{\boldsymbol{\phi}}(\boldsymbol{x}_{\text{agg.}}, \{\boldsymbol{c}_j\}_{j=1}^{N_c}), \tag{4}$$

where $\boldsymbol{x}_{\text{agg.}}$ is the aggregation prompt provided in Appendix O. By default, we retain cues supported by all eight pairwise comparisons. Since T2I introduces variations in visual appearance, cues that recur consistently across multiple pairwise comparisons are more likely to reflect content grounded in the original image. The aggregation combines these pairwise consistent cues to produce the consistent visual cue $\boldsymbol{c}$, which serves as guidance for preference construction.

### 2.2 Cue-conditioned self-distillation for preference synthesis

The visual cue $\boldsymbol{c}$ guides the policy's own rollouts. We sample $N_s$ cue-conditioned self-generated candidate responses from the policy:

$$\boldsymbol{y}^{(s)} \sim \pi_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{x}_{\text{cue-cap.}}, \boldsymbol{m}, \boldsymbol{c}), \tag{5}$$

where $s$ indexes the sampled candidate responses and $\boldsymbol{x}_{\text{cue-cap.}}$ denotes the cue-conditioned captioning prompt provided in Appendix O. Repeated sampling under the same prompt, image, and visual cue produces diverse cue-conditioned candidate responses. Since these candidates are sampled from the policy itself, they remain close to the policy's response distribution, while the visual cue encourages them to leverage visual information from the image, resulting in varying degrees of visual grounding across candidates.

**Preference selection by visual grounding.** From the cue-conditioned candidate responses $\{\boldsymbol{y}^{(s)}\}_{s=1}^{N_s}$, we select a preferred response $\boldsymbol{y}_w$ and a dispreferred response $\boldsymbol{y}_l$ based on visual grounding. Specifically, we query the judge model with a visual grounding prompt:

$$(\boldsymbol{y}_w, \boldsymbol{y}_l) = \pi_{\boldsymbol{\phi}}\left(\boldsymbol{x}_{\text{vis.}}, \boldsymbol{m}, \{\boldsymbol{y}^{(s)}\}_{s=1}^{N_s}\right), \tag{6}$$

where $\boldsymbol{x}_{\text{vis.}}$ is the visual grounding prompt provided in Appendix O. It asks the judge model to compare the candidate responses with respect to the original image $\boldsymbol{m}$. The selected pair contrasts a more visually grounded response with a less grounded one, forming a policy-aligned preference pair from the policy's own cue-conditioned generations. We provide the ViPSy algorithm in Appendix G.

Table 1: Comparison of hallucination and general benchmarks using LLaVA-1.5-7B & LLaVA-1.5-13B. **Bold** and underline values indicate the best and second-best results, respectively.

| Model | Method | Hallucination benchmarks | | | | | | General benchmarks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Object HalBench | | AMBER | | | HallusionBench | MMStar | MMVP | CV-Bench |
| | | Resp.↓ | Ment.↓ | CHAIR↓ | Hal.↓ | Cog.↓ | aAcc.↑ | Acc.↑ | Acc.↑ | Acc.↑ |
| LLaVA-1.5-7B | Vanilla | 51.3 | 25.9 | 7.9 | 36.8 | 4.3 | 46.8 | 33.4 | 23.3 | 61.9 |
| | VCD (Leng et al., 2024) | 50.6 | 25.6 | 8.1 | 37.9 | 4.0 | 47.2 | 34.5 | 25.3 | 56.0 |
| | DOLA (Chuang et al., 2023) | 44.3 | 24.9 | 6.5 | 28.5 | 3.1 | 46.7 | 33.5 | 22.6 | 62.0 |
| | OPERA (Huang et al., 2024) | 42.0 | 22.6 | 5.8 | 27.7 | 3.0 | 45.7 | 33.4 | 22.6 | 61.9 |
| | EFUF (Xing et al., 2024) | 41.6 | 22.6 | 5.8 | 28.2 | 3.0 | 47.0 | 33.6 | 24.6 | 62.2 |
| | LLaVA-RLHF (Sun et al., 2024) | 52.6 | 26.9 | 9.8 | 46.4 | 5.4 | 41.8 | 20.1 | 12.0 | 27.8 |
| | HALVA (Sarkar et al., 2024) | 41.3 | 20.5 | 6.8 | 33.3 | 3.5 | **48.8** | 32.7 | 23.3 | 61.3 |
| | HA-DPO (Zhao et al., 2023b) | 39.0 | 19.6 | 6.6 | 30.9 | 3.2 | 48.1 | 32.7 | 22.6 | 61.4 |
| | POVID (Zhou et al., 2024) | 31.6 | 16.3 | 5.4 | 28.7 | 3.0 | 46.5 | 33.6 | 25.3 | 62.6 |
| | TPR (He et al., 2024) | 11.0 | 5.8 | 3.6 | 20.4 | 1.7 | 39.8 | 33.2 | 29.3 | 63.6 |
| | RLAIF-V (Yu et al., 2024b) | 8.6 | 4.1 | 2.9 | 15.8 | 0.9 | 35.1 | 34.0 | 24.0 | 42.4 |
| | OPA-DPO (Yang et al., 2025) | 8.6 | 5.1 | 2.7 | 13.7 | 1.1 | 47.9 | 32.8 | 28.0 | 59.5 |
| | SENTINEL (Peng et al., 2025) | 5.3 | 3.0 | 2.8 | 13.6 | 1.2 | 47.6 | 33.6 | 24.6 | 61.9 |
| | $P^2$-DPO (Zhang et al., 2026b) | 5.0 | 3.2 | 5.8 | 26.7 | 2.4 | 47.9 | 33.3 | 23.3 | 62.3 |
| | ViPSy (**Ours**) | **4.0** | **2.6** | **1.8** | **9.6** | **0.6** | 48.1 | **35.4** | **30.6** | **64.4** |
| LLaVA-1.5-13B | Vanilla | 44.6 | 22.1 | 5.9 | 28.1 | 4.1 | 46.5 | 33.2 | 32.0 | 61.7 |
| | HALVA (Sarkar et al., 2024) | 47.3 | 12.8 | 5.8 | 28.8 | 4.1 | 47.6 | 33.4 | 32.6 | 61.4 |
| | LLaVA-RLHF (Sun et al., 2024) | 36.3 | 11.8 | 6.8 | 34.1 | 5.2 | 45.9 | 20.5 | 21.3 | 37.6 |
| | HSA-DPO (Xiao et al., 2025) | 5.0 | 3.0 | 2.2 | 12.2 | 1.6 | 46.1 | 34.5 | 33.3 | 60.1 |
| | OPA-DPO (Yang et al., 2025) | 6.0 | 3.7 | 2.1 | 9.4 | 1.4 | 46.4 | 34.6 | 31.3 | 61.3 |
| | SENTINEL (Peng et al., 2025) | 3.3 | 1.9 | 2.4 | 11.9 | 1.2 | 47.4 | 32.4 | **35.3** | 60.5 |
| | ViPSy (**Ours**) | **2.6** | **1.3** | **1.9** | **9.2** | **1.1** | **48.0** | **35.8** | **35.3** | **62.6** |

### 2.3 Preference optimization with ViPSy

Once ViPSy constructs the preference dataset $\mathcal{D}_{\text{pref}} = \{(\boldsymbol{x}_i, \boldsymbol{m}_i, \boldsymbol{y}_{w,i}, \boldsymbol{y}_{l,i})\}_{i=1}^{N}$, the pairs can be used with a range of pairwise preference optimization objectives (Zhao et al., 2023a; Azar et al., 2024; Hong et al., 2024; Meng et al., 2024). We instantiate this step with DPO (Rafailov et al., 2023) to match the optimization objective used in recent hallucination mitigation methods, enabling a fair comparison while isolating the effect of our preference construction. We optimize:

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(\boldsymbol{x},\boldsymbol{m},\boldsymbol{y}_w,\boldsymbol{y}_l)\sim\mathcal{D}_{\text{pref}}}\left[\log\sigma\left(\beta\log\frac{\pi_{\boldsymbol{\theta}}(\boldsymbol{y}_w \mid \boldsymbol{x},\boldsymbol{m})}{\pi_{\text{ref}}(\boldsymbol{y}_w \mid \boldsymbol{x},\boldsymbol{m})} - \beta\log\frac{\pi_{\boldsymbol{\theta}}(\boldsymbol{y}_l \mid \boldsymbol{x},\boldsymbol{m})}{\pi_{\text{ref}}(\boldsymbol{y}_l \mid \boldsymbol{x},\boldsymbol{m})}\right)\right], \quad (7)$$

where $\pi_{\text{ref}}$ denotes the reference policy and $\beta$ is a scaling parameter. The consistent visual cue $\boldsymbol{c}$ and self-captioned synthetic images $\{\boldsymbol{m}'_j\}_{j=1}^{K}$ are used only for constructing the preference pairs, not as inputs to the policy during DPO. During optimization, the policy and reference model evaluate the constructed responses $(\boldsymbol{y}_w, \boldsymbol{y}_l)$ conditioned only on the original prompt-image pair $(\boldsymbol{x}, \boldsymbol{m})$. This design enables cue-conditioned self-distillation by using preference optimization to distill cue-guided visual grounding into the image-conditioned policy. Consequently, ViPSy improves visual grounding without requiring cue extraction or synthetic image generation at inference time.

## 3 Experiments

Our experiments evaluate whether ViPSy addresses the two limitations identified above: (a) policy-distribution mismatch in intervention-based methods and (b) limited use of visual information in sampling-based preference construction. To validate it, we assess whether the constructed preference pairs reduce hallucination while enhancing general vision-language capability, and further examine whether self-derived visual cues strengthen the vision encoder through visual representation and segmentation evaluations rather than merely suppressing language-prior-driven hallucination.

### 3.1 Setups

**Models and Baselines.** Following prior work, we use LLaVA-1.5-7B and LLaVA-1.5-13B (Liu et al., 2024a) as the policy models for direct comparison with existing hallucination-mitigation methods, Qwen2.5-VL-72B (Team, 2025a) as the default judge, and SD-3.5-L (Esser et al., 2024) as the default T2I diffusion model for semantic synthesis. We compare ViPSy with decoding methods

Table 2: Average hallucination and general benchmark scores across policy models.

| Method | Hal. Avg. ↓ | Gen. Avg. ↑ |
|---|---|---|
| **Qwen2.5-VL-7B** | | |
| Vanilla | 9.48 | 64.90 |
| ViPSy | **8.16** | **66.10** |
| **mPLUG-Owl3-7B** | | |
| Vanilla | 6.66 | 48.87 |
| ViPSy | **5.84** | **50.10** |
| **InternVL3.5-8B** | | |
| Vanilla | 8.66 | 69.07 |
| ViPSy | **6.78** | **69.77** |
| **Kimi-VL-A3B** | | |
| Vanilla | 11.52 | 57.97 |
| ViPSy | **8.70** | **59.53** |

Table 3: Ablation study on model configurations based on average scores across hallucination and general benchmarks, respectively. "Self" denotes sampling from LLaVA-1.5-7B.

| ViPSy Configuration | | | Benchmarks | |
|---|---|---|---|---|
| I→T | T→I | Rollout | Hal. Avg. ↓ | Gen. Avg. ↑ |
| *Vanilla (LLaVA-1.5-7B)* | | | 25.24 | 39.53 |
| Self | SD-3.5-L | Self | **3.72** | **43.47** |
| GPT-5 mini | SD-3.5-L | Self | 5.12 | 42.57 |
| Qwen3-VL-30B | SD-3.5-L | Self | 4.70 | 42.13 |
| Qwen2.5-VL-72B | SD-3.5-L | Self | 4.26 | 42.40 |
| Self | PixelDiT | Self | 4.50 | 42.70 |
| Self | Flux-1.0-dev | Self | 4.36 | 42.50 |
| Self | SD-3.5-M | Self | 3.86 | 42.87 |
| Self | SD-3.5-L | GPT-5 mini | 15.64 | 41.07 |
| Self | SD-3.5-L | Qwen3-VL-30B | 16.00 | 42.07 |
| Self | SD-3.5-L | Qwen2.5-VL-72B | 11.22 | 41.80 |

Table 4: Vision encoder evaluation with 2-, 4-, and 8-shot linear probing accuracy on ImageNet-1k (IN-1k), ImageNet-O (IN-O), and Rendered-SST2 (R-SST2), and MS COCO segmentation recall.

| Method | Linear Probing | | | | | | | | | Seg. |
|---|---|---|---|---|---|---|---|---|---|---|
| | 2-shot | | | 4-shot | | | 8-shot | | | |
| | IN-1k Acc.↑ | IN-O Acc.↑ | R-SST2 Acc.↑ | IN-1k Acc.↑ | IN-O Acc.↑ | R-SST2 Acc.↑ | IN-1k Acc.↑ | IN-O Acc.↑ | R-SST2 Acc.↑ | MS COCO Recall ↑ |
| Vanilla | 35.4 | 18.0 | 48.9 | 48.6 | 29.5 | 50.4 | 59.3 | 38.0 | 50.1 | 34.7 |
| ViPSy (**Ours**) | **36.1** | **18.3** | **49.5** | **49.0** | **30.0** | **51.0** | **60.0** | **38.5** | **50.9** | **35.9** |

(Leng et al., 2024; Chuang et al., 2023; Huang et al., 2024), which operate only at inference time, and unlearning-based methods (Xing et al., 2024), which rely on curated data. Intervention-based methods (Zhao et al., 2023b; Zhou et al., 2024; Sarkar et al., 2024; He et al., 2024; Xiao et al., 2025) can deviate from the policy distribution, while sampling-based methods (Sun et al., 2024; Yu et al., 2024b; Yang et al., 2025; Peng et al., 2025) often underuse visual information during preference construction. We also compare with $P^2$-DPO (Zhang et al., 2026b), which constructs on-policy pairs from modified visual inputs. Model and baseline details are provided in Appendices D and E.

**Datasets and evaluation.** We construct preference data from Visual Genome images (Krishna et al., 2017), following Peng et al. (2025), with implementation and training details in Appendix C. We evaluate hallucination on Object HalBench (Rohrbach et al., 2018), AMBER (Wang et al., 2023a), and HallusionBench (Guan et al., 2024), and general vision-language capability on MM-Star (Chen et al., 2024), MMVP (Tong et al., 2024b), and CV-Bench (Tong et al., 2024a). For vision encoder evaluation, we use ImageNet 1k (Russakovsky et al., 2015), ImageNet-O (Hendrycks et al., 2021), and Rendered-SST2 (Socher et al., 2013; Radford et al., 2021) for linear probing, and MS COCO (Lin et al., 2014) for segmentation. Dataset and metric details are provided in Appendix F.

### 3.2 Main Results

**Mitigating hallucination.** Table 1 shows that ViPSy achieves state-of-the-art hallucination mitigation on LLaVA-1.5-7B and remains effective on LLaVA-1.5-13B. On LLaVA-1.5-7B, against TPR (He et al., 2024), the strongest intervention-based method, our method reduces Object HalBench Resp./Ment. by 63.6%/55.2% and AMBER CHAIR/Hal./Cog. by 50.0%/52.9%/64.7%. Against the strongest sampling-based method, SENTINEL (Peng et al., 2025), it improves Object HalBench Resp./Ment. by 24.5%/13.3% and AMBER CHAIR/Hal./Cog. by 35.7%/29.4%/50.0%. These gains support policy-aligned, visually grounded preference construction. Although it ranks second on HallusionBench, it achieves the best results on the remaining benchmarks, showing broader gains in faithful visual understanding. ViPSy also outperforms $P^2$-DPO (Zhang et al., 2026b), which exploits visual information via modified visual inputs, suggesting that cue-guided preference construction better exploits visual information. On the larger LLaVA-1.5-13B model, ViPSy outperforms all baselines across all hallucination benchmarks. In particular, compared with

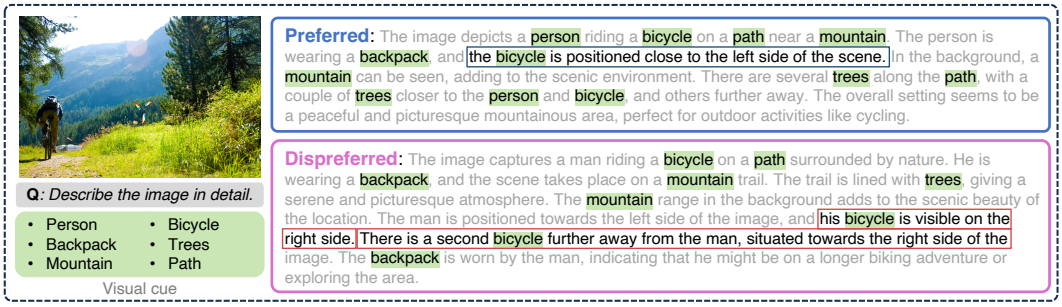


Figure 3: Qualitative examples of cue-conditioned preference pairs. Given the same image, prompt, and visual cue, preferred responses remain image-grounded, while dispreferred ones hallucinate. Additional examples are shown in Appendix N.1, N.2, and N.3: preference pairs, aligned-policy image descriptions, and self-captioned semantic synthesis, respectively.

the previous state-of-the-art, it achieves notable improvements of 21.2%/31.6% on Object HalBench Resp./Ment. and 20.8%/22.7%/8.3% on AMBER CHAIR/Hal./Cog. These results indicate that ViPSy remains effective as the model scales. Qualitative examples in Appendix N.2 show that our preference pairs reduce unsupported details and produce image-grounded descriptions. Additional evaluation beyond object-presence hallucination is provided in Appendix L, where Table 20 shows consistent improvements. Response analysis after preference alignment is presented in Appendix K, and Table 18 shows that the improvement is not simply due to shorter or less detailed generations. Under the same judge, ViPSy also achieves the lowest hallucination scores, showing that its gains do not stem from the choice of judge. Details are provided in Appendix J, Table 15.

**General performance.** On general benchmarks with LLaVA-1.5-7B, our preference construction improves over vanilla on all three benchmarks, with gains of 6.0% on MMStar, 31.3% on MMVP, and 4.0% on CV-Bench, achieving the best overall results among compared methods. Unlike sampling-based methods that degrade or merely maintain CV-Bench, our method improves it, suggesting that visual cues enhance general vision-language capability. Together with the hallucination results, these gains support policy-aligned, visually grounded preference pairs. The results on LLaVA-1.5-13B further demonstrate scalability, as our method achieves the best performance across all benchmarks and improves over vanilla by 7.8% on MMStar and 10.3% on MMVP. The results under the same judge models also show that ViPSy preserves the strongest general benchmark performance, with details provided in Appendix J, Table 15.

**Generalization across policy models.** The effectiveness of ViPSy is not specific to LLaVA-1.5-7B. Table 2 shows that our method generalizes across diverse policy models, consistently improving average hallucination and general benchmark performance. Compared with the vanilla policy, it reduces the hallucination average, computed over two hallucination benchmarks (Rohrbach et al., 2018; Wang et al., 2023a), by 13.9% on Qwen2.5-VL-7B, 12.3% on mPLUG-Owl3-7B, 21.7% on InternVL3.5-8B, and 24.5% on Kimi-VL-A3B. At the same time, it improves the general average across three general benchmarks (Chen et al., 2024; Tong et al., 2024b;a). These results demonstrate that policy-aligned, visually grounded preference construction provides consistent benefits beyond a single policy model. Full results are provided in Appendix H.1.

### 3.3 Analysis

**Effect on vision encoder.** To examine whether our approach improves the vision encoder beyond response-level hallucination mitigation, we use few-shot linear probing and segmentation. We update the vision encoder during preference alignment and freeze the resulting encoder for linear probing and segmentation. Following Liu et al. (2025b), we freeze the resulting vision encoder and train dataset-specific linear classifiers on frozen image features under 2-, 4-, and 8-shot settings. For segmentation, we train a two-layer transformer output head to predict patch-level semantic labels, following Covert et al. (2024). As shown in Table 4, our method improves ImageNet-1k, ImageNet-O, and Rendered-SST2 accuracy as well as segmentation recall over vanilla. These gains suggest that ViPSy improves both visual representation quality and the ability to separate visual regions, rather than merely suppressing language-prior-driven hallucinations in responses.

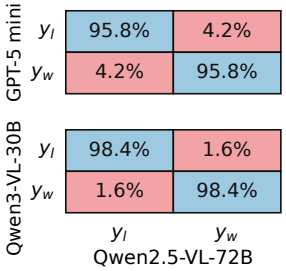


Figure 4: Agreement of preference selections for alternative judges compared upon preferences based on Qwen2.5-VL-72B.

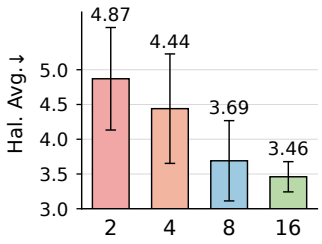


Figure 5: Effect of varying the number of cue-conditioned rollouts on average hallucination errors across five runs.

| Strategy | Hal. Avg. ↓ | Gen. Avg. ↑ |
|---|---|---|
| Vanilla | 25.24 | 39.53 |
| w/o cue | 7.14 | 42.50 |
| Translation | 5.10 | 43.13 |
| Resize | 4.82 | 42.47 |
| Mask | 4.82 | 42.53 |
| Crop | 4.76 | 42.43 |
| Video | 4.46 | 41.70 |
| Diffusion | **3.72** | **43.47** |

Table 5: Comparison of visual variations for cue extraction in average hallucination and general benchmark scores.

**Qualitative analysis of cue-conditioned rollouts.** We show how ViPSy obtains preferred and dispreferred responses, and we present examples of preference pairs selected from multiple cue-conditioned rollouts under the same image, prompt, and visual cue. Preferred responses use the visual cue to ground the image description, whereas dispreferred responses use the cue but add details unsupported by the image. As highlighted in Figure 3, the same visual cue can yield candidates with different grounding quality, e.g., in the bicycle scene, the preferred response correctly locates the bicycle close to the left side of the scene, while the dispreferred response states that the bicycle is on the right side and further hallucinates a second bicycle, although only one bicycle is visible. This demonstrates that preference selection is needed to separate faithful cue usage from unsupported extrapolation. More examples are provided in Appendix N.1. We additionally quantify the differences between preferred and dispreferred responses and the distributional shift of the constructed preference pairs in Appendix K, where Table 17 analyzes response quality and characteristics, and Table 16 evaluates distributional alignment with the vanilla policy.

**Judge agreement and robustness.** Since preference selection is performed by a judge over multiple cue-conditioned rollouts, one concern is whether the selected pairs depend on the judge model. We therefore evaluate the agreement between the default judge, Qwen2.5-VL-72B, and alternative judges. We also measure hallucination and general capabilities when each judge is used to construct preference pairs. As shown in Figure 4, Qwen3-VL-30B and GPT-5 mini agree with the default judge on 98.4% and 95.8% of preference selections, respectively, indicating that the selected preferences are highly consistent across judges. Furthermore, Table 10 shows that different judges yield similar results on both hallucination and general benchmarks. This suggests that the preference construction process is not dominated by judge-specific bias and that ViPSy reliably constructs visually grounded preference pairs. Detailed results are provided in Appendix H.4. We further evaluate robustness to the judge used for pairwise visual cue extraction, distinct from the preference-selection judge above. The improvements are largely maintained with smaller cue-extraction judges, indicating robustness to the choice of judge. Detailed results are provided in Appendix J, Table 14.

### 3.4 Ablation Study

**Visual cue construction.** We analyze visual-cue construction by varying the captioning and T2I models. The input image is captioned, and the caption is used to synthesize an image for cue extraction. As shown in Table 3, self-captioning achieves the best overall result, while alternative T2I models remain competitive with moderate performance differences. This suggests that using policy-generated captions is beneficial for cue construction, and that ViPSy is not overly sensitive to the T2I model. Full results are in Appendix H.2, and Qualitative examples in Appendix N.3 show that self-captioned synthetic images preserve the main visual context across T2I models.

**Cue-guided preference construction.** We analyze how visual cues affect preference construction. Table 6 shows that cues from semantic synthesis provide more reliable guidance than direct extraction from the original image. Table 13 further shows that stronger aggregation consensus improves cue correctness and preference construction. Agree=$k$ retains cues supported by at least $k$ of eight comparisons, and cue correctness is the fraction matching Visual Genome annotations over the full

Table 6: Ablation on cue construction strategies with average hallucination and general benchmark scores. Cue correctness is measured against Visual Genome annotations.

| Method | Self-captioning | T→I | Cue extraction | Voting (8 samples) | Cue correctness (%) ↑ | Hal. Avg. ↓ | Gen. Avg. ↑ |
|---|---|---|---|---|---|---|---|
| No cue | X | X | X | X | – | 7.14 | 42.50 |
| Original image | X | X | O | X | 81 | 6.17 | 42.63 |
| Original image + Voting | X | X | O | O | 85 | 5.94 | 42.78 |
| Self-caption + Voting | O | X | O | O | 83 | 6.18 | 42.45 |
| ViPSy (**Ours**) | O | O | O | O | **95** | **3.72** | **43.47** |

dataset, where GPT-5 mini determines whether similar terms match the same annotation. These results highlight the importance of reliable cues from semantic comparison and consensus.

**Rollout model.** We compare models for cue-conditioned rollout generation while keeping the visual-cue construction fixed. Self-rollouts achieve the best hallucination score and general performance, whereas rollouts from external VLMs lead to degraded results. This shows that candidate responses should remain close to the policy distribution while being guided by visual cues. Full results are provided in Appendix H.2.

**Number of rollouts.** To assess how the number of rollouts affects preference construction, we vary the number of cue-conditioned rollouts. As shown in Figure 5, more rollouts improve reliability by enlarging the candidate pool and increasing grounding-quality diversity. Increasing the number of rollouts reduces variance in hallucination score, as the larger candidate pool makes the preferred and dispreferred responses more clearly distinguishable. Since more rollouts can improve preference construction but increase computational cost, we use 8 cue-conditioned rollouts as a trade-off between reliability and efficiency. Additional results are included in Appendix H.3.

**Cue conditioning and synthesis strategy.** Table 5 studies cue conditioning and visual variation strategies for cue extraction. Even without cues, policy-sampled candidates improve over the vanilla model, but cue-conditioned construction performs better, confirming the importance of visual cues for visually faithful rollouts. Among cue extraction strategies, simple transformations such as translation, resize, mask, and crop bring improvements but provide limited semantic diversity. Video-based variation using an image-to-video model (Team, 2025b) can introduce uncontrolled changes. Diffusion-based synthesis produces richer semantic variations and achieves the best overall result, with the lowest hallucination score and the highest general capability. Detailed results are reported in Appendix H.5. We also examine cue granularity in Appendix L, where Table 19 compares object, action, and spatial relationship cues. Spatial cues perform best on general benchmarks, while object cues achieve the strongest hallucination mitigation and remain competitive on general benchmarks.

**Offline computational efficiency.** We analyze the computational cost of offline preference construction in Appendix M. Table 21 compares ViPSy with TPR, a state-of-the-art intervention-based method, and shows that fewer synthetic variants or lighter synthesis backbones substantially lower construction cost with only modest performance changes. Table 22 provides a component-wise breakdown, showing that diffusion-based synthesis accounts for most of the computation.

## 4 Conclusion

In this paper, we proposed ViPSy, a vision-driven preference synthesis that constructs preference pairs leveraging visual information while remaining aligned with the policy distribution. Unlike intervention-based methods that externally modify responses and sampling-based methods that overlook visual information, our approach first extracts visual cues by comparing the original image with semantically synthesized image variations. These cues then guide the policy model's own rollouts, allowing candidate responses to remain close to the model's generation distribution while encouraging them to leverage visual information from the input image. Experiments show that our approach achieves state-of-the-art hallucination mitigation while also improving general visual grounding benchmarks, with additional gains observed in vision encoder evaluations.

### AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic data sets as part of our proposed framework. We did not use generative AI tools for other tasks requiring disclosure, and the remaining required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to edit the manuscript to improve readability. All AI-assisted content was reviewed by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

### REPRODUCIBILITY STATEMENT

Appendix C provides the training settings and compute infrastructure. Appendix D details the models, Appendix E describes the baseline implementations, and Appendix F provides the datasets and evaluation metrics. Appendix O lists all prompts used for visual cue extraction and preference construction.

## REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes. In *The twelfth international conference on learning representations*, 2024.

Vedika Agarwal, Rakshith Shetty, and Mario Fritz. Towards Causal VQA: Revealing and Reducing Spurious Correlations by Invariant and Covariant Semantic Editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9690–9698, 2020.

Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. Analyzing the Behavior of Visual Question Answering Models. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1955–1960, 2016.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a Visual Language Model for Few-Shot Learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A General Theoretical Paradigm to Understand Learning from Human Preferences. In *International Conference on Artificial Intelligence and Statistics*, pp. 4447–4455. PMLR, 2024.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL Technical Report. *arXiv preprint arXiv:2511.21631*, 2025.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. *arXiv preprint arXiv:2204.05862*, 2022a.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional AI: Harmlessness from AI Feedback. *arXiv preprint arXiv:2212.08073*, 2022b.

Garrick Brazil, Abhinav Kumar, Julian Straub, Nikhila Ravi, Justin Johnson, and Georgia Gkioxari. Omni3D: A Large Benchmark and Model for 3D Object Detection in the Wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13154–13164, 2023.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback. *arXiv preprint arXiv:2307.15217*, 2023.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are We on the Right Way for Evaluating Large Vision-Language Models? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024.

Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. YOLO-World: Real-Time Open-Vocabulary Object Detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16901–16911, 2024.

Zhe Cheng, Wenyu Chen, Fode Zhang, and Dehuan Shen. Mitigating hallucinations in large vision-language models via causal route gating. *arXiv preprint arXiv:2605.24024*, 2026.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality, March 2023. URL https://lmsys.org/blog/2023-03-30-vicuna/.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. DoLa: Decoding by Contrasting Layers Improves Factuality in Large Language Models. *arXiv preprint arXiv:2309.03883*, 2023.

Ian Covert, Tony Sun, James Zou, and Tatsunori Hashimoto. Locality Alignment Improves Vision-Language Models. *arXiv preprint arXiv:2410.11087*, 2024.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. *Advances in neural information processing systems*, 36:49250–49267, 2023.

Qingxiu Dong, Li Dong, Xingxing Zhang, Zhifang Sui, and Furu Wei. Self-Boosting Large Language Models with Synthetic Preference Data. *arXiv preprint arXiv:2410.06961*, 2024.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis. In *Forty-first international conference on machine learning*, 2024.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. KTO: Model Alignment as Prospect Theoretic Optimization. *arXiv preprint arXiv:2402.01306*, 2024.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6904–6913, 2017.

Zhaopeng Gu, Bingke Zhu, Guibo Zhu, Yingying Chen, Ming Tang, and Jinqiao Wang. AnomalyGPT: Detecting Industrial Anomalies Using Large Vision-Language Models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 1932–1940, 2024.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, et al. HallusionBench: An Advanced Diagnostic Suite for Entangled Language Hallucination and Visual Illusion in Large Vision-Language Models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14375–14385, 2024.

Anisha Gunjal, Jihan Yin, and Erhan Bas. Detecting and Preventing Hallucinations in Large Vision Language Models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 18135–18143, 2024.

Yudong Han, Liqiang Nie, Jianhua Yin, Jianlong Wu, and Yan Yan. Visual Perturbation-aware Collaborative Learning for Overcoming the Language Prior Problem. *arXiv preprint arXiv:2207.11850*, 2022.

Lehan He, Zeren Chen, Zhelun Shi, Tianyu Yu, Jing Shao, and Lu Sheng. Systematic Reward Gap Optimization for Mitigating VLM Hallucinations. *arXiv preprint arXiv:2411.17265*, 2024.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural Adversarial Examples. *CVPR*, 2021.

Jiwoo Hong, Noah Lee, and James Thorne. ORPO: Monolithic Preference Optimization without Reference Model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 11170–11189, 2024.

Xuehe Hou, Wenshuo Li, Yali Li, Han Shu, Yuan Wang, Xinghao Chen, and Shengjin Wang. Ves-rft: Rewarding visual evidence sensitivity to mitigate hallucinations in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4168–4177, 2026.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. OPERA: Alleviating Hallucination in Multi-Modal Large Language Models via Over-Trust Penalty and Retrospection-Allocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13418–13427, 2024.

Yiyang Huang, Liang Shi, Yitian Zhang, Yi Xu, and Yun Fu. SHIELD: Suppressing Hallucinations In LVLM Encoders via Bias and Vulnerability Defense. *arXiv preprint arXiv:2510.16596*, 2025.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2.5-Coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

Hamish Ivison, Yizhong Wang, Jiacheng Liu, Zeqiu Wu, Valentina Pyatkin, Nathan Lambert, Noah A Smith, Yejin Choi, and Hannaneh Hajishirzi. Unpacking DPO and PPO: Disentangling Best Practices for Learning from Preference Feedback. *Advances in neural information processing systems*, 37:36602–36633, 2024.

Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. WinCLIP: Zero-/Few-Shot Anomaly Classification and Segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19606–19616, 2023.

Songtao Jiang, Yan Zhang, Ruizhe Chen, Tianxiang Hu, Yeying Jin, Qinglin He, Yang Feng, Jian Wu, and Zuozhu Liu. Modality-Fair Preference Optimization for Trustworthy MLLM Alignment. *arXiv preprint arXiv:2410.15334*, 2024a.

Xi Jiang, Jian Li, Hanqiu Deng, Yong Liu, Bin-Bin Gao, Yifeng Zhou, Jialin Li, Chengjie Wang, and Feng Zheng. MMAD: A Comprehensive Benchmark for Multimodal Large Language Models in Industrial Anomaly Detection. *arXiv preprint arXiv:2410.09453*, 2024b.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International journal of computer vision*, 123(1):32–73, 2017.

Black Forest Labs. FLUX. https://github.com/black-forest-labs/flux, 2024.

Seongyun Lee, Sue Hyun Park, Yongrae Jo, and Minjoon Seo. Volcano: Mitigating Multimodal Hallucination through Self-Feedback Guided Revision. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 391–404, 2024a.

Unggi Lee, Minji Jeon, Yunseo Lee, Gyuri Byun, Yoorim Son, Jaeyoon Shin, Hongkyu Ko, and Hyeoncheol Kim. LLaVA-Docent: Instruction Tuning with Multimodal Large Language Model to Support Art Appreciation Education. *Computers and Education: Artificial Intelligence*, 7: 100297, 2024b.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. 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*, pp. 13872–13882, 2024.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. LLaVA-Med: Training a Large Language-and-Vision Assistant for Biomedicine in One Day. *Advances in Neural Information Processing Systems*, 36: 28541–28564, 2023.

Long-Fei Li, Yu-Yang Qian, Peng Zhao, and Zhi-Hua Zhou. Provably Efficient Online RLHF with One-Pass Reward Modeling. *arXiv preprint arXiv:2502.07193*, 2025.

Xiaoqi Li, Mingxu Zhang, Yiran Geng, Haoran Geng, Yuxing Long, Yan Shen, Renrui Zhang, Jiaming Liu, and Hao Dong. ManipLLM: Embodied Multimodal Large Language Model for Object-Centric Robotic Manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18061–18070, 2024.

Byeonggeuk Lim, JungMin Yun, Junehyoung Kwon, Kyeonghyun Kim, and YoungBin Kim. Aligning with your own voice: Self-corrected preference learning for hallucination mitigation in lvlms. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 35829–35849, 2026.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved Baselines with Visual Instruction Tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 26296–26306, 2024a.

Jinyi Liu, Yifu Yuan, Jianye Hao, Fei Ni, Lingzhi Fu, Yibin Chen, and Yan Zheng. Enhancing Robotic Manipulation with AI Feedback from Multimodal Large Language Models. *arXiv preprint arXiv:2402.14245*, 2024b.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024c.

Wenqi Liu, Xuemeng Song, Jiaxi Li, Yinwei Wei, Na Zheng, Jianhua Yin, and Liqiang Nie. Mitigating Hallucination Through Theory-Consistent Symmetric Multimodal Preference Optimization. *arXiv preprint arXiv:2506.11712*, 2025a.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pp. 216–233. Springer, 2024d.

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-RFT: Visual Reinforcement Fine-Tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2034–2044, 2025b.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating Mathematical Reasoning of Foundation Models in Visual Contexts. *arXiv preprint arXiv:2310.02255*, 2023.

Yizhen Luo, Jiahuan Zhang, Siqi Fan, Kai Yang, Yushuai Wu, Mu Qiao, and Zaiqing Nie. BioMedGPT: Open Multimodal Generative Pre-trained Transformer for BioMedicine. *arXiv preprint arXiv:2308.09442*, 2023.

Yu Meng, Mengzhou Xia, and Danqi Chen. SimPO: Simple Preference Optimization with a Reference-Free Reward. *Advances in Neural Information Processing Systems*, 37:124198–124235, 2024.

Michael Moor, Qian Huang, Shirley Wu, Michihiro Yasunaga, Yash Dalmia, Jure Leskovec, Cyril Zakka, Eduardo Pontes Reis, and Pranav Rajpurkar. Med-Flamingo: a Multimodal Medical Few-shot Learner. In *Machine learning for health (ML4H)*, pp. 353–367. PMLR, 2023.

OpenAI. GPT-4V(ision) System Card, 2023. URL https://api.semanticscholar.org/CorpusID:263218031.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Yu Pan, Zhongze Cai, Guanting Chen, Huaiyang Zhong, and Chonghuan Wang. What Matters in Data for DPO? *arXiv preprint arXiv:2508.18312*, 2025.

Shangpin Peng, Senqiao Yang, Li Jiang, and Zhuotao Tian. Mitigating Object Hallucinations via Sentence-Level Early Intervention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 635–646, 2025.

Renjie Pi, Tianyang Han, Wei Xiong, Jipeng Zhang, Runtao Liu, Rui Pan, and Tong Zhang. Strengthening Multimodal Large Language Model with Bootstrapped Preference Optimization. In *European Conference on Computer Vision*, pp. 382–398. Springer, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning Transferable Visual Models From Natural Language Supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

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

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Pritam Sarkar, Sayna Ebrahimi, Ali Etemad, Ahmad Beirami, Sercan Ö. Arık, and Tomas Pfister. Mitigating Object Hallucination in MLLMs via Data-augmented Phrase-level Alignment, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint arXiv:2402.03300*, 2024.

Hakim Sidahmed, Samrat Phatale, Alex Hutcheson, Zhuonan Lin, Zhang Chen, Zac Yu, Jarvis Jin, Simral Chaudhary, Roman Komarytsia, Christiane Ahlheim, et al. Parameter Efficient Reinforcement Learning from Human Feedback. *arXiv preprint arXiv:2403.10704*, 2024.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. OpenAI GPT-5 System Card. *arXiv preprint arXiv:2601.03267*, 2025.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize from human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, et al. Aligning Large Multimodal Models with Factually Augmented RLHF. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13088–13110, 2024.

Fahim Tajwar, Anikait Singh, Archit Sharma, Rafael Rafailov, Jeff Schneider, Tengyang Xie, Stefano Ermon, Chelsea Finn, and Aviral Kumar. Preference Fine-Tuning of LLMs Should Leverage Suboptimal, On-Policy Data. *arXiv preprint arXiv:2404.14367*, 2024.

Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, Congcong Wang, Dehao Zhang, Dikang Du, Dongliang Wang, Enming Yuan, Enzhe Lu, Fang Li, Flood Sung, Guangda Wei, Guokun Lai, Han Zhu, Hao Ding, Hao Hu, Hao Yang, Hao Zhang, Haoning Wu, Haotian Yao, Haoyu Lu, Heng Wang, Hongcheng Gao, Huabin Zheng, Jiaming Li, Jianlin Su, Jianzhou Wang, Jiaqi Deng, Jiezhong Qiu, Jin Xie, Jinhong Wang, Jingyuan Liu, Junjie Yan, Kun Ouyang, Liang Chen, Lin Sui, Longhui Yu, Mengfan Dong, Mengnan Dong, Nuo Xu, Pengyu Cheng, Qizheng Gu, Runjie Zhou, Shaowei Liu, Sihan Cao, Tao Yu, Tianhui Song, Tongtong Bai, Wei Song, Weiran He, Weixiao Huang, Weixin Xu, Xiaokun Yuan, Xingcheng Yao, Xingzhe Wu, Xinxing Zu, Xinyu Zhou, Xinyuan Wang, Y. Charles, Yan Zhong, Yang Li, Yangyang Hu, Yanru Chen, Yejie Wang, Yibo Liu, Yibo Miao, Yidao Qin, Yimin Chen, Yiping Bao, Yiqin Wang, Yongsheng Kang, Yuanxin Liu, Yulun Du, Yuxin Wu, Yuzhi Wang, Yuzi Yan, Zaida Zhou, Zhaowei Li, Zhejun Jiang, Zheng Zhang, Zhilin Yang, Zhiqi Huang, Zihao Huang, Zijia Zhao, and Ziwei Chen. Kimi-VL Technical Report, 2025. URL https://arxiv.org/abs/2504.07491.

Qwen Team. Qwen2.5-VL, January 2025a. URL https://qwenlm.github.io/blog/qwen2.5-vl/.

Wan Team. Wan: Open and Advanced Large-Scale Video Generative Models, 2025b.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai C Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A Fully Open, Vision-Centric Exploration of Multimodal LLMs. *Advances in Neural Information Processing Systems*, 37:87310–87356, 2024a.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes Wide Shut? Exploring the Visual Shortcomings of Multimodal LLMs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9568–9578, 2024b.

Fei Wang, Wenxuan Zhou, James Y Huang, Nan Xu, Sheng Zhang, Hoifung Poon, and Muhao Chen. mDPO: Conditional Preference Optimization for Multimodal Large Language Models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8078–8088, 2024.

Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, et al. AMBER: An LLM-free Multi-dimensional Benchmark for MLLMs Hallucination Evaluation. *arXiv preprint arXiv:2311.07397*, 2023a.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing Open-Source Multimodal Models in Versatility, Reasoning, and Efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 13484–13508, 2023b.

Yike Wu, Yu Zhao, Shiwan Zhao, Ying Zhang, Xiaojie Yuan, Guoqing Zhao, and Ning Jiang. Overcoming Language Priors in Visual Question Answering via Distinguishing Superficially Similar Instances. In *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 5721–5729, 2022.

Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-Play Preference Optimization for Language Model Alignment. *arXiv preprint arXiv:2405.00675*, 2024.

Wenyi Xiao, Ziwei Huang, Leilei Gan, Wanggui He, Haoyuan Li, Zhelun Yu, Fangxun Shu, Hao Jiang, and Linchao Zhu. Detecting and Mitigating Hallucination in Large Vision Language Models via Fine-Grained AI Feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 25543–25551, 2025.

Yuxi Xie, Guanzhen Li, Xiao Xu, and Min-Yen Kan. V-DPO: Mitigating Hallucination in Large Vision Language Models via Vision-Guided Direct Preference Optimization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 13258–13273, 2024.

Shangyu Xing, Fei Zhao, Zhen Wu, Tuo An, Weihao Chen, Chunhui Li, Jianbing Zhang, and Xinyu Dai. EFUF: Efficient Fine-grained Unlearning Framework for Mitigating Hallucinations in Multimodal Large Language Models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1167–1181, 2024.

Chuyan Xiong, Chengyu Shen, Xiaoqi Li, Kaichen Zhou, Jeremy Liu, Ruiping Wang, and Hao Dong. AIC MLLM: Autonomous Interactive Correction MLLM for Robust Robotic Manipulation. *arXiv preprint arXiv:2406.11548*, 2024.

Qinwu Xu. Reducing hallucination in vision-language models via stage-wise preference optimization under distribution shift. *arXiv preprint arXiv:2605.16411*, 2026.

Zhaorui Yang, Tianyu Pang, Haozhe Feng, Han Wang, Wei Chen, Minfeng Zhu, and Qian Liu. Self-Distillation Bridges Distribution Gap in Language Model Fine-Tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1028–1043, 2024.

Zhihe Yang, Xufang Luo, Dongqi Han, Yunjian Xu, and Dongsheng Li. Mitigating Hallucinations in Large Vision-Language Models via DPO: On-Policy Data Hold the Key. *arXiv preprint arXiv:2501.09695*, 2025.

Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mPLUG-Owl3: Towards Long Image-Sequence Understanding in Multi-Modal Large Language Models. *arXiv preprint arXiv:2408.04840*, 2024.

Chengzhi Yu, Yifan Xu, Yifan Chen, and Wenyi Zhang. Optimizing lvlms with on-policy data for effective hallucination mitigation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 4733–4743, March 2026.

Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, et al. RLHF-V: Towards Trustworthy MLLMs via Behavior Alignment from Fine-grained Correctional Human Feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13807–13816, 2024a.

Tianyu Yu, Haoye Zhang, Yuan Yao, Yunkai Dang, Da Chen, Xiaoman Lu, Ganqu Cui, Taiwen He, Zhiyuan Liu, Tat-Seng Chua, et al. RLAIF-V: Aligning MLLMs through Open-Source AI Feedback for Super GPT-4V Trustworthiness. *arXiv preprint arXiv:2405.17220*, 2(3):8, 2024b.

Yongsheng Yu, Wei Xiong, Weili Nie, Yichen Sheng, Shiqiu Liu, and Jiebo Luo. PixelDiT: Pixel Diffusion Transformers for Image Generation. *arXiv preprint arXiv:2511.20645*, 2025.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-Rewarding Language Models. *arXiv preprint arXiv:2401.10020*, 2024.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid Loss for Language Image Pre-Training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

Huatian Zhang, Zhendong Mao, Lei Zhang, and Yongdong Zhang. Uncertainty-aware exploratory direct preference optimization for multimodal large language models. *arXiv preprint arXiv:2605.04874*, 2026a.

Ruipeng Zhang, Zhihao Li, Haozhang Yuan, CL Chen, and Tong Zhang. $P^2$-DPO: Grounding hallucination in perceptual processing via calibration direct preference optimization. *arXiv preprint arXiv:2606.03376*, 2026b.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-Distilled Reasoner: On-Policy Self-Distillation for Large Language Models. *arXiv preprint arXiv:2601.18734*, 2026.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. SLiC-HF: Sequence Likelihood Calibration with Human Feedback. *arXiv preprint arXiv:2305.10425*, 2023a.

Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. Beyond Hallucinations: Enhancing LVLMs through Hallucination-Aware Direct Preference Optimization. *arXiv preprint arXiv:2311.16839*, 2023b.

Zihui Zhao and Zechang Li. Reflective preference optimization (rpo): Enhancing on-policy alignment via hint-guided reflection. *arXiv preprint arXiv:2512.13240*, 2025.

Ren Zhibo, Wang Huizhen, Zhu Muhua, Wang Yichao, Xiao Tong, and Zhu Jingbo. Overcoming Language Priors with Counterfactual Inference for Visual Question Answering. In *Proceedings of the 22nd Chinese National Conference on Computational Linguistics*, pp. 600–610, 2023.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene Parsing through ADE20K Dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 633–641, 2017.

Yiyang Zhou, Chenhang Cui, Rafael Rafailov, Chelsea Finn, and Huaxiu Yao. Aligning Modalities in Vision Large Language Models via Preference Fine-tuning. *arXiv preprint arXiv:2402.11411*, 2024.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-Tuning Language Models from Human Preferences. *arXiv preprint arXiv:1909.08593*, 2019.

## A Limitations and broader impact

Although ViPSy achieves strong hallucination mitigation, it relies on external models for preference construction, including a text-to-image diffusion model and a judge VLM. Consequently, the resulting preference pairs may inherit biases or failure modes from these models when synthesized variants or judge decisions do not faithfully reflect image-grounded evidence. Improving or domain-adapting these external models could further enhance preference quality. This study also focuses mainly on object-level visual grounding with cues from Visual Genome images. While effective for reducing object hallucination, broader scenarios such as complex reasoning, spatial understanding, and specialized domains remain underexplored. ViPSy may improve VLM reliability in applications such as education, accessibility, robotics, industrial inspection, and assistive scientific or medical workflows.

## B Related work

**Preference alignment.** Preference alignment aims to make model outputs better reflect human preferences. A representative approach is Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022; Sun et al., 2024), which trains a reward model to judge rollout preferences and then optimizes the policy with PPO (Schulman et al., 2017). This framework has been used for aligning models to be helpful (Bai et al., 2022a), while AI feedback can reduce reliance on human preference labeling (Bai et al., 2022b; Xiao et al., 2025; Yu et al., 2024b). Recent RL variants such as Group Relative Policy Optimization (GRPO) reduce policy-optimization cost with group-relative rewards (Shao et al., 2024), but still require rollout-level reward evaluation, which is difficult to define for VLM hallucination mitigation under vision-language misalignment (Sun et al., 2024). DPO (Rafailov et al., 2023) instead learns from pre-constructed preferred and dispreferred response pairs, avoiding explicit reward modeling or online RL and shifting the key challenge to preference construction. Building on the same preference optimization perspective, related methods have explored alternative objectives, including sequence-level likelihood calibration (Zhao et al., 2023a), general theoretical objectives for learning from preferences (Azar et al., 2024), alignment from binary desirable or undesirable feedback (Ethayarajh et al., 2024), and optimization without a reference model (Hong et al., 2024; Meng et al., 2024). Preference-based methods further improve visual faithfulness by modifying alignment objectives (Xie et al., 2024; Jiang et al., 2024a; Wang et al., 2024; Liu et al., 2025a). These works provide diverse preference optimization objectives, while our work complements them by constructing visually grounded preference data for VLM hallucination mitigation.

**Mitigating hallucination in VLMs.** Motivated by this shift from reward design to preference-data construction, hallucination mitigation in VLMs focuses on constructing visually grounded preference pairs, where preferred responses should be faithful to the image while dispreferred responses capture hallucinated or unsupported content. Preference-construction methods differ in how response pairs are obtained. Intervention-based methods construct pairs by modifying responses or visual inputs, including hallucination injection (Zhou et al., 2024), hallucination correction (Zhao et al., 2023b), segment-level correction (Yu et al., 2024a), semantic-component manipulation (He et al., 2024), phrase modification in ground-truth descriptions (Sarkar et al., 2024), and distorted images (Pi et al., 2024). Xu (2026) construct hard preference pairs by augmenting visually grounded examples with challenging queries and plausible but visually unsupported alternatives. In contrast, sampling-based methods derive pairs from policy-generated responses using external judges (Yu et al., 2024b), including hallucination classifiers that distinguish hallucination-free from hallucinated responses (Yu et al., 2026), SFT-guided on-policy alignment (Yang et al., 2025), and detector-based verification (Peng et al., 2025), which relies on detector models (Liu et al., 2024c; Cheng et al., 2024). More recently, More recently, Lim et al. (2026) construct in-distribution preference pairs by verifying and refining the policy's own responses through self-correction. Zhao & Li (2025) use externally generated reflective hints to guide the policy in regenerating preferred responses while preserving on-policy preference construction. Zhang et al. (2026b) generate policy responses from modified visual inputs to construct preference pairs targeting perceptual hallucinations. Our work follows this sampling-based direction while further incorporating visual information through visual cues extracted from self-captioned semantic synthesis and cue-conditioned self-distillation for preference synthesis.

**Policy-aligned self-generated data.** Sampling-based methods build on the observation that learning can benefit from training data that is close to the policy distribution (Tajwar et al., 2024; Agarwal et al., 2024). This perspective has motivated a broad range of self-generation approaches, including self-generated instructions (Wang et al., 2023b), self-generated rewards (Yuan et al., 2024), self-play preferences (Wu et al., 2024), synthetic preference data (Dong et al., 2024), and policy-aligned self-distillation (Yang et al., 2024; Zhao et al., 2026). However, these methods do not explicitly account for visual inputs during self-generation. When an image is provided, the resulting supervision can therefore be driven by language priors rather than the visual information in the image. ViPSy addresses this limitation by conditioning the policy's own rollouts on consistent visual cues, keeping the generated data close to the policy distribution while making its construction explicitly vision-grounded.

## C Training details

We build our experiments on top of LLaVA-1.5 (Liu et al., 2024a), using Vicuna-1.5 (Chiang et al., 2023) as the language model backbone and CLIP ViT-L/336px (Radford et al., 2021) as the vision encoder. The visual features are mapped to the language space through the default two-layer GELU MLP projector. The training set is constructed from Visual Genome images (Krishna et al., 2017). Our final preference dataset contains 7K preference pairs, which is smaller than the training data used by Peng et al. (2025), the previous state-of-the-art method for hallucination mitigation. This allows us to evaluate whether visually grounded preference construction can be effective even with a relatively compact preference dataset. For optimization, we use LoRA-based parameter-efficient tuning. The tuning targets include the LLM, the vision encoder, and the projector. We insert trainable LoRA modules into these components rather than updating all model parameters. The LoRA modules for the LLM and projector are trained with a learning rate of $2 \times 10^{-5}$, while the vision encoder uses $2 \times 10^{-6}$. We use LoRA rank $r = 128$, scaling factor $\alpha = 256$, and set the preference loss coefficient to $\beta = 0.1$. Models are trained with AdamW, cosine learning-rate decay, no weight decay, and a maximum context length of 2048. Training is performed for one epoch with a global batch size of 64. We employ ZeRO stage-2 optimization to reduce memory usage. Both $N_c$ and $N_s$ are set to 8, representing the number of captions and cue-conditioned self-generated candidates, respectively.

**Compute infrastructure.**

All experiments were conducted on compute cluster instances with two NVIDIA B200 GPUs, each with 183GB of VRAM, two Intel Xeon 6960P CPUs with 144 physical cores and 288 threads in total.

## D Models

**Policy models.**

- **LLaVA 1.5** (Liu et al., 2024a) is used as the main policy model in our experiments. We evaluate both the 7B and 13B variants to study the effect of policy model scale. LLaVA 1.5 combines a CLIP-based visual encoder with a Vicuna language model through a multimodal projector, enabling it to generate language responses conditioned on both images and textual instructions. In our framework, LLaVA 1.5 generates self-captions, produces cue-conditioned candidate responses, and is optimized with the preference pairs constructed by ViPSy. Its relatively simple connector-based design also makes it a useful baseline for isolating how preference optimization, rather than architectural complexity, affects visual grounding behavior.
- **InternVL3.5-8B** (Wang et al., 2025) is used as an additional policy model to evaluate whether ViPSy generalizes beyond the LLaVA family. InternVL3.5 is a multimodal instruction-following model designed for visual understanding and reasoning. Including this model allows us to test whether visually grounded preference construction remains effective for a different VLM architecture.
- **Kimi-VL-A3B** (Team et al., 2025) is used as a policy model for evaluating generalization across model families. It is a vision-language model developed for multimodal instruc-

tion following and image-grounded reasoning. This model is included to evaluate whether ViPSy improves visual grounding when applied to a policy model with a distinct multimodal architecture.

- **mPLUG-Owl3-7B** (Ye et al., 2024) is used as another policy model in our policy generalization experiments. The mPLUG-Owl family follows a modular multimodal design that connects visual perception with language generation. We include this model to examine whether ViPSy can improve visual grounding for a policy model with a different multimodal architecture.
- **Qwen2.5-VL-7B** (Team, 2025a) is used as a policy model in addition to its larger judge counterpart. It is an instruction-tuned vision-language model with strong visual understanding and multimodal reasoning capabilities. Using the 7B variant as a policy model helps evaluate whether ViPSy is effective for a compact Qwen-family VLM, rather than only for LLaVA-based policies.

**Judge models.**

- **Qwen2.5-VL-72B** (Team, 2025a) is used as the default judge model in our experiments. We use this model to extract consistent visual cues between original and synthetic images and to select preferred and dispreferred responses from cue-conditioned candidates. Due to its large scale and instruction-tuned multimodal capability, it provides a strong visual understanding for evaluating whether a response is grounded in the original image. Its fine-grained visual localization capability is useful for judging whether candidate responses preserve object- and region-level evidence rather than only global image semantics.
- **Qwen3-VL-30B** (Bai et al., 2025) is used as a judge model. It serves as an alternative multimodal evaluator for cue extraction and preference selection, allowing us to examine whether ViPSy depends on a single judge architecture. By including Qwen3-VL-30B, we assess the robustness of the constructed preferences across different VLM judges. In our experiments, Qwen3-VL-30B is used only for judging and not as the policy model being optimized.
- **GPT-5 mini** (Singh et al., 2025) is used as an additional judge model for cue extraction and preference selection. We include it as a complementary evaluator to the Qwen-family VLM judges, allowing us to examine whether preference construction remains robust when the judge comes from a different model family. In our experiments, GPT-5 mini is used only for judging and not as the policy model being optimized. As a lightweight text-and-vision model, it provides a cost-efficient contrast to larger VLM judges while retaining multimodal reasoning ability for preference selection.

**Text-to-Image models.**

- **SD-3.5** (Esser et al., 2024) is included as a representative latent-space diffusion model. It generates images in a compressed latent representation rather than directly in pixel space, which enables efficient image synthesis while maintaining high visual quality. In our experiments, SD-3.5-L is used to synthesize semantically aligned image variants for visual cue extraction. Its multimodal transformer backbone enables strong interaction between text and image representations, making it a reliable latent-generation baseline for cue-consistency analysis.
- **Flux-1.0-dev** (Labs, 2024) is included as a representative flow matching model. Unlike standard denoising diffusion formulations, flow matching learns a continuous transformation from noise to data. We use Flux-1.0-dev to test whether ViPSy is robust to a different text-to-image generation paradigm during semantic synthesis. As a high-capacity rectified-flow transformer trained with guidance distillation, it provides a distinct synthesis backbone with strong prompt-following behavior.
- **PixelDiT** (Yu et al., 2025) is included as a representative pixel-space diffusion model. In contrast to latent diffusion models, PixelDiT performs generation in pixel space. We include PixelDiT to examine whether visual cue extraction remains effective with pixel-space diffusion.

# E BASELINES

We compare ViPSy with hallucination mitigation methods for VLMs. All baselines use LLaVA-1.5 to ensure a fair comparison. We organize the baselines into three groups: sampling-based methods, intervention-based methods, and other methods. Sampling-based methods construct preference data from VLM-generated responses, which keeps the data close to the policy distribution, but they can still underuse visual information. Intervention-based methods construct preference data through external editing or correction, creating explicit contrasts between hallucinated and hallucination-free responses while moving the data away from the policy distribution. Other methods include one unlearning-based method and three training-free decoding methods.

**Sampling-based methods.**

- **LLaVA-RLHF** (Sun et al., 2024) learns a reward model from human feedback and uses reinforcement learning to train the VLM to prefer image-faithful responses. To better penalize hallucinations, the reward model is also given extra factual evidence, such as image captions or ground-truth answers.
- **RLAIF-V** (Yu et al., 2024b) uses sampling-based AI feedback to reduce hallucinations in VLMs. It generates multiple candidate responses via controlled sampling under the same input, allowing preference pairs to focus on trustworthiness differences. It then scores responses at the claim level and iteratively refines the VLM using the resulting preference feedback.
- **SENTINEL** (Peng et al., 2025) targets object hallucination via sentence-level early intervention. It constructs in-domain preference pairs by using open-vocabulary object detectors to validate objects in VLM-generated sentences, and trains the VLM to prefer context-coherent non-hallucinated sentences over hallucinated ones, reducing hallucination propagation.
- **OPA-DPO** (Yang et al., 2025) performs on-policy alignment with corrected responses. It uses GPT-4V to minimally revise hallucinated VLM outputs and applies LoRA-SFT to align both ground-truth and revised responses with the VLM's policy distribution.

**Intervention-based methods.**

- **HALVA** (Sarkar et al., 2024) addresses object hallucination in VLMs through Data-augmented Phrase-level Alignment (DPA). DPA constructs correct and hallucinated response pairs by selectively replacing grounded visual concepts in correct responses with hallucinated phrases. It penalizes hallucinated phrases relative to their correct counterparts, while using token-wise KL regularization to preserve general vision-language capabilities.
- **HA-DPO** (Zhao et al., 2023b) builds preference data from Visual Genome images by prompting VLMs to generate detailed descriptions and using GPT-4 with VG annotations to identify unsupported visual claims and revise them. The revised faithful responses are then paired with their original erroneous counterparts, producing positive and negative samples for preference-based training.
- **POVID** (Zhou et al., 2024) frames VLM hallucination as modality misalignment and constructs preference pairs by keeping grounded responses as preferred while generating dispreferences through text- and image-side interventions. Text interventions inject plausible hallucinations, whereas image distortions trigger inherent hallucination patterns, guiding preference optimization toward visually grounded generation.
- **HSA-DPO** (Xiao et al., 2025) trains a sentence-level hallucination detector from fine-grained AI feedback and uses it as a detect-then-rewrite intervention to build preference pairs.
- **TPR** (He et al., 2024) intervenes in preference data at the topic level. Through selective replacement with VLM-resampled topic alternatives, it controls reward gaps and applies curriculum hard negative mining for data-efficient alignment.

**Other methods.**

- **EFUF** (Xing et al., 2024) is an unlearning-based method for reducing object hallucinations in VLMs. It uses CLIP-based image-relevance scores to automatically curate grounded and hallucinated object mentions without paired annotations, then unlearns hallucinated subsentences while preserving correct object generation and sentence fluency.
- **VCD** (Leng et al., 2024) is a training-free decoding method based on visual contrastive decoding. It contrasts the output distributions conditioned on the original image and a noise-distorted visual input, thereby calibrating predictions that are overly driven by language priors and statistical biases rather than visual evidence.
- **DoLa** (Chuang et al., 2023) is a training-free decoding method that improves factuality by contrasting predictions from earlier and later transformer layers during decoding. This helps the model rely on factual knowledge it has already learned, without using external retrieval or additional fine-tuning.
- **OPERA** (Huang et al., 2024) is a training-free decoding method that adjusts beam search based on self-attention patterns. It penalizes candidates that rely too much on summary tokens and, when this reliance leads to hallucination, revises the generation by returning to an earlier token and choosing a different next token.
- **P$^2$-DPO** (Zhang et al., 2026b) is an on-policy preference optimization method that constructs preference pairs from modified visual inputs. It generates responses from enhanced, degraded, and noisy images to target perceptual bottlenecks and visual robustness, and further uses a calibration loss to strengthen visual grounding.

# F Data

## F.1 Training data

We construct the VLM preference training data using images from Visual Genome (VG) (Krishna et al., 2017). VG is a large-scale vision-language dataset that provides dense annotations for real-world images, including region descriptions, objects, attributes, relationships, and question-answer pairs. We use VG images as source images to generate visual variations and extract consistent visual cues for preference construction. Since our method relies on diverse visual scenes for this process, VG serves as a suitable data source for applying the proposed method.

## F.2 Evaluation data

We evaluate the VLM on two groups of benchmarks, namely hallucination benchmarks and general visual-language capabilities benchmarks. In addition, to examine whether the proposed training improves the underlying vision encoder, we conduct vision encoder evaluations using linear probing and segmentation datasets. For hallucination evaluation, we use Object HalBench, AMBER, and HallusionBench. For general multimodal evaluation, we use MMStar, MMVP, and CV-Bench. For vision encoder evaluation, we use ImageNet, ImageNet-O, Rendered-SST2, and MSCOCO.

### F.2.1 Hallucination Benchmarks

- **Object HalBench** (Rohrbach et al., 2018) is based on MSCOCO (Lin et al., 2014) and evaluates object hallucination in image captions. Evaluation is restricted to the 80 MSCOCO object categories with segmentation annotations. Object presence is determined using both MSCOCO segmentation labels and reference captions, and generated object mentions are normalized through singularization and synonym matching.

  We report two hallucination metrics, Resp. and Ment. Resp. measures the fraction of generated responses that contain at least one hallucinated object, while Ment. measures the fraction of object mentions that are hallucinated:

$$\text{Resp.} = \frac{N_{\text{resp.}}^{\text{hal.}}}{N_{\text{resp.}}}, \qquad \text{Ment.} = \frac{N_{\text{ment.}}^{\text{hal.}}}{N_{\text{ment.}}}. \tag{8}$$

  Here, $N_{\text{resp.}}$ is the total number of generated responses, and $N_{\text{resp.}}^{\text{hal.}}$ is the number of responses containing at least one hallucinated object. Similarly, $N_{\text{ment.}}$ is the total number

of generated object mentions, and $N^{\text{hal.}}_{\text{ment.}}$ is the number of hallucinated object mentions. Lower values indicate less object hallucination.

- **AMBER** (Wang et al., 2023a) contains 1,004 manually selected images annotated with visible objects, attributes, relations, and cognitively plausible hallucination target objects. We use its generative evaluation split, where models are prompted with *"Describe this image."* and hallucination is evaluated from the generated image description.

  Given a response $R$, noun mentions are extracted as candidate objects $R_{\text{obj}}$ and filtered using the AMBER object vocabulary $X_{\text{obj}}$:

$$R'_{\text{obj}} = R_{\text{obj}} \cap X_{\text{obj}}. \tag{9}$$

  The filtered object mentions $R'_{\text{obj}}$, which are then compared with the ground-truth object set $A_{\text{obj}}$, and the hallucination target set $H_{\text{obj}}$. We report three metrics: CHAIR, Hal., and Cog.

  CHAIR measures the fraction of generated object mentions ungrounded in the image:

$$\text{CHAIR}(R) = 1 - \frac{\left|R'_{\text{obj}} \cap A_{\text{obj}}\right|}{\left|R'_{\text{obj}}\right|}. \tag{10}$$

  Hal. is a response-level hallucination indicator that checks whether the response contains at least one ungrounded object:

$$\text{Hal}(R) = \mathbb{I}\left[\text{CHAIR}(R) \neq 0\right]. \tag{11}$$

  Cog. measures the fraction of generated object mentions that match AMBER's cognitively plausible hallucination targets, i.e., absent objects that are likely to be imagined from the visual context:

$$\text{Cog}(R) = \frac{\left|R'_{\text{obj}} \cap H_{\text{obj}}\right|}{\left|R'_{\text{obj}}\right|}. \tag{12}$$

  All three metrics are averaged over the generative-evaluation samples, and lower values indicate better performance.

- **HallusionBench** (Guan et al., 2024) evaluates image-context reasoning in VLMs, focusing on language hallucination and visual illusion. It consists of 346 images and 1,129 human-crafted yes/no questions across diverse visual formats, including charts, tables, maps, OCR images, optical illusions, math figures, posters, and video-like image sequences.

  We report the overall question-level accuracy, denoted as All Acc. (aAcc). For each valid image-question pair $(\mathbf{I}, \mathbf{q})$, the model prediction is mapped to a binary correctness value $b_M(\mathbf{I}, \mathbf{q}) \in \{0, 1\}$, where 1 indicates a correct prediction, and 0 indicates an incorrect prediction. The metric is computed as the average correctness over all valid pairs:

$$\text{aAcc} = \frac{\sum_{(\mathbf{I},\mathbf{q})\in\mathcal{V}} b_M(\mathbf{I}, \mathbf{q})}{|\mathcal{V}|}. \tag{13}$$

  Here, $\mathcal{V}$ is the set of valid image-question pairs. Higher values indicate better performance.

#### F.2.2 General benchmarks

- **MMStar** (Chen et al., 2024) is a vision-indispensable multimodal benchmark consisting of 1,500 human-curated samples across 6 core capabilities and 18 fine-grained axes. It evaluates the visual understanding and reasoning abilities of large vision-language models across coarse perception, fine-grained perception, instance reasoning, logical reasoning, science and technology, and mathematics. The benchmark is designed to reduce text-only shortcuts and data leakage by requiring models to rely on visual content to answer each question.
- **MMVP** (Tong et al., 2024b) evaluates the fine-grained visual grounding ability of multi-modal LLMs through visual question answering on CLIP-blind image pairs. These image

pairs are visually different to humans but are encoded similarly by CLIP, making them useful for testing whether a model can recognize subtle visual details. The benchmark covers basic visual patterns, including orientation and direction, presence of specific features, state and condition, counting, positional relationships, color and appearance, structural characteristics, text, and viewpoint. Following the official evaluation protocol, an image pair is counted as correct only when both associated questions are answered correctly.

- **CV-Bench** (Tong et al., 2024a) is a vision-centric benchmark for evaluating the fundamental visual understanding capabilities of multimodal LLMs. It reformulates standard computer vision benchmarks into VQA-style questions to assess both 2D and 3D perception. The 2D tasks evaluate spatial relationships and object counting, while the 3D tasks evaluate depth order and relative distance. The benchmark contains 2,638 manually inspected examples constructed from ADE20K (Zhou et al., 2017), MSCOCO (Lin et al., 2014), and Omni3D (Brazil et al., 2023). For CV-Bench, Acc. is computed as the average accuracy over the 2D and 3D task groups.

#### F.2.3 Vision encoder evaluation

- **ImageNet-1k** (Russakovsky et al., 2015) is a large-scale object recognition benchmark with 1,000 semantic categories. We use ImageNet-1k to evaluate the global semantic representation quality of the vision encoder through linear probing, where the vision encoder is frozen, and only a data-specific linear classifier is trained on top of the extracted visual features. We report classification accuracy (Acc.), where higher values indicate better performance.
- **ImageNet-O** (Hendrycks et al., 2021) is an out-of-distribution object recognition benchmark designed to evaluate model robustness beyond the ImageNet distribution. We use ImageNet-O to examine whether the learned visual representations remain effective on uncommon and challenging object-centric images. Since ImageNet-O is used in a few-shot evaluation setting, we select a small number of samples from its test set for few-shot training. Specifically, the vision encoder is frozen, and only a data-specific linear classifier is trained using the extracted visual features of the selected few-shot samples. We then evaluate the classifier on the remaining ImageNet-O test samples. This benchmark complements ImageNet-1k by evaluating whether visual features remain effective across diverse distributions. We report classification accuracy (Acc.), where higher values indicate better performance.
- **Rendered-SST2** (Socher et al., 2013; Radford et al., 2021) is constructed by rendering sentences from the SST-2 sentiment classification dataset as images. We use Rendered-SST2 under the same linear-probing setting, where the vision encoder is frozen, and only a data-specific linear classifier is trained on top of the extracted visual features. This benchmark evaluates whether the vision encoder preserves semantic information in text-rendered visual inputs. Unlike object-centric recognition datasets, Rendered-SST2 tests whether visual representations can capture image-based textual semantics. We report classification accuracy (Acc.), where higher values indicate better performance.
- **MSCOCO** (Lin et al., 2014) contains images with object-level annotations, including segmentation masks. We use MSCOCO for segmentation evaluation to assess whether the vision encoder can distinguish fine-grained object-level information. This evaluation complements linear probing by measuring local visual grounding rather than only global image-level semantics. We report Recall, where higher values indicate better performance.

## G Algorithm

Algorithm 1 presents the ViPSy preference construction pipeline. For each image-prompt pair, ViPSy first generates multiple self-captions using the policy model and synthesizes semantic image variants with a text-to-image model. A judge model compares each variant with the original image to extract shared object-level evidence, and aggregates these pairwise cues into a consistent visual cue. Then, the policy model samples multiple responses conditioned on the original prompt, image, and visual cue. The judge selects the most visually grounded candidate as the preferred response and the least grounded candidate as the dispreferred response, forming a preference pair. The synthetic

**Algorithm 1** ViPSy

**Input:** Dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{m}_i)\}_{i=1}^{M}$; policy $\pi_{\boldsymbol{\theta}}$; judge $\pi_{\boldsymbol{\phi}}$; T2I model $P_{\psi}$; number of self-captions/synthetic variants $N_c$; number of cue-conditioned rollouts $N_s$; prompts $\boldsymbol{x}_{\text{cap.}}$ for captioning, $\boldsymbol{x}_{\text{comp.}}$ for image comparison, $\boldsymbol{x}_{\text{agg.}}$ for cue aggregation, $\boldsymbol{x}_{\text{cue-cap.}}$ for cue-conditioned rollout, and $\boldsymbol{x}_{\text{vis.}}$ for visual-grounding preference selection

**Output:** Preference dataset $\mathcal{D}_{\text{pref}}$

$\mathcal{D}_{\text{pref}} \leftarrow \emptyset$ // Initialize preference dataset

**for** each $(\boldsymbol{x}_i, \boldsymbol{m}_i) \in \mathcal{D}$ **do**

**{Stage 1: Self-captioned visual cue extraction}**

$\{\boldsymbol{q}_j\}_{j=1}^{N_c} \sim \pi_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{x}_{\text{cap.}}, \boldsymbol{m}_i)$ // Generate $N_c$ self-captions

$\{\boldsymbol{m}'_{i,j}\}_{j=1}^{N_c} = \{P_{\psi}(\boldsymbol{q}_j)\}_{j=1}^{N_c}$ // Synthesize semantic image variants

$\boldsymbol{c}_{i,j} = \pi_{\boldsymbol{\phi}}(\boldsymbol{x}_{\text{comp.}}, \boldsymbol{m}_i, \boldsymbol{m}'_{i,j})$ // Extract consistent visual cues

$\boldsymbol{c}_i = \pi_{\boldsymbol{\phi}}(\boldsymbol{x}_{\text{agg.}}, \{\boldsymbol{c}_{i,j}\}_{j=1}^{N_c})$ // Aggregate consistent visual cues

**{Stage 2: Cue-conditioned preference synthesis}**

$\boldsymbol{y}_i^{(s)} \sim \pi_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{x}_{\text{cue-cap.}}, \boldsymbol{m}_i, \boldsymbol{c}_i), \quad s = 1, \ldots, N_s$ // Sample cue-conditioned responses

$(\boldsymbol{y}_{w,i}, \boldsymbol{y}_{l,i}) = \pi_{\boldsymbol{\phi}}\left(\boldsymbol{x}_{\text{vis.}}, \boldsymbol{m}_i, \{\boldsymbol{y}_i^{(s)}\}_{s=1}^{N_s}\right)$ // Select visually grounded and less grounded responses

$\mathcal{D}_{\text{pref}} \leftarrow \mathcal{D}_{\text{pref}} \cup \{(\boldsymbol{x}_i, \boldsymbol{m}_i, \boldsymbol{y}_{w,i}, \boldsymbol{y}_{l,i})\}$ // Store preference pair

**end for**

**return** $\mathcal{D}_{\text{pref}}$

images and visual cues are used only for constructing preference data and are not required during inference.

# H Additional results

We provide the full results discussed in Section 3. Specifically, Hal. AVG. denotes the average score over Object HalBench and AMBER, which measure hallucination in generated responses. Gen. AVG. denotes the average score over MMStar, MMVP, and CV-Bench, which are used to evaluate general visual-language capabilities in Section 3.

## H.1 Policy model ablation

Table 7 reports the results of applying preference alignment to different policy models using policy-specific preference data constructed by ViPSy. The results show that ViPSy not only mitigates hallucination but also improves general vision-language capabilities. This indicates that ViPSy is not limited to a single policy model and can generalize across different VLMs.

## H.2 Model component ablation

In Table 8, we analyze how each model component in ViPSy affects performance. We vary the image-to-text captioning model, the text-to-image synthesis model, and the rollout model used in ViPSy. The comparison highlights that self-captioning and self-rollouts are important for keeping the constructed preference data close to the policy distribution, while the choice of text-to-image model has a smaller effect.

## H.3 Number of rollouts ablation

Table 9 reports the results for different numbers of cue-conditioned rollouts. Overall, increasing the number of rollouts generally improves performance across both hallucination and general benchmarks, as a larger candidate pool provides more diverse responses for preference selection. In particular, using more rollouts enables the judge to compare candidates with different grounding quality under the same image, prompt, and visual cue, resulting in more informative preference pairs for training. While 16 rollouts further improve several metrics, the gains over 8 rollouts are relatively

Table 7: Results on hallucination and general benchmarks using various policy models. We apply ViPSy to diverse policy models.

| Method | Hallucination benchmarks | | | | | | General benchmarks | | |
|---|---|---|---|---|---|---|---|---|---|
| | Object HalBench | | AMBER | | | HallusionBench | MMStar | MMVP | CV-Bench |
| | Resp.↓ | Ment.↓ | CHAIR↓ | Hal.↓ | Cog.↓ | aAcc.↑ | Acc.↑ | Acc.↑ | Acc.↑ |
| **Qwen2.5-VL-7B** | | | | | | | | | |
| Vanilla | 9.0 | 5.3 | 5.0 | 26.4 | 1.7 | 57.7 | 59.8 | 56.6 | 78.3 |
| ViPSy (**Ours**) | 7.6 | 4.8 | 4.1 | 23.3 | 1.0 | 60.4 | 61.7 | 58.6 | 78.0 |
| **mPLUG-Owl3-7B** | | | | | | | | | |
| Vanilla | 11.6 | 6.6 | 5.4 | 8.3 | 1.4 | 50.8 | 48.3 | 43.3 | 55.0 |
| ViPSy (**Ours**) | 9.6 | 5.9 | 4.8 | 7.8 | 1.1 | 54.6 | 48.2 | 45.3 | 56.8 |
| **InternVL3.5-8B** | | | | | | | | | |
| Vanilla | 8.3 | 4.6 | 4.3 | 24.6 | 1.5 | 64.2 | 66.3 | 59.3 | 81.6 |
| ViPSy (**Ours**) | 7.3 | 4.2 | 3.9 | 17.3 | 1.2 | 65.4 | 67.9 | 60.0 | 81.4 |
| **Kimi-VL-A3B** | | | | | | | | | |
| Vanilla | 5.6 | 3.5 | 5.9 | 38.8 | 3.8 | 56.0 | 49.5 | 48.6 | 75.8 |
| ViPSy (**Ours**) | 5.0 | 3.1 | 4.1 | 29.4 | 1.9 | 57.7 | 52.2 | 50.0 | 76.4 |

Table 8: Results on hallucination and general benchmarks using LLaVA-1.5-7B. We vary the image-to-text captioning model (I→T), the text-to-image synthesis model (T→I), and the rollout model while keeping the remaining components fixed. "Self" denotes using the policy model itself.

| ViPSy Configuration | | | Hallucination benchmarks | | | | | | General benchmarks | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| I→T | T→I | Rollout | Object HalBench | | AMBER | | | HallusionBench | MMStar | MMVP | CV-Bench |
| | | | Resp.↓ | Ment.↓ | CHAIR↓ | Hal.↓ | Cog.↓ | aAcc.↑ | Acc.↑ | Acc.↑ | Acc.↑ |
| *Vanilla (LLaVA-1.5-7B)* | | | 51.3 | 25.9 | 7.9 | 36.8 | 4.3 | 46.8 | 33.4 | 23.3 | 61.9 |
| Self | SD-3.5-L | Self | 4.0 | 2.6 | 1.8 | 9.6 | 0.6 | 48.1 | 35.4 | 30.6 | 64.4 |
| GPT-5 mini | SD-3.5-L | Self | 6.6 | 4.4 | 1.9 | 12.0 | 0.7 | 47.2 | 33.8 | 29.3 | 64.6 |
| Qwen3-VL-30B | SD-3.5-L | Self | 4.6 | 3.5 | 2.2 | 12.4 | 0.8 | 47.9 | 34.4 | 28.0 | 64.0 |
| Qwen2.5-VL-72B | SD-3.5-L | Self | 4.0 | 2.2 | 2.1 | 12.1 | 0.9 | 47.7 | 34.8 | 29.3 | 63.1 |
| Self | PixelDiT | Self | 3.3 | 2.4 | 2.4 | 13.5 | 0.9 | 48.7 | 34.7 | 28.6 | 64.8 |
| Self | Flux-1.0-dev | Self | 4.6 | 3.1 | 2.3 | 11.1 | 0.7 | 48.7 | 34.6 | 29.3 | 63.6 |
| Self | SD-3.5-M | Self | 4.3 | 2.6 | 1.7 | 10.1 | 0.6 | 48.2 | 34.6 | 29.3 | 64.7 |
| Self | SD-3.5-L | GPT-5 mini | 30.6 | 15.1 | 5.3 | 24.3 | 2.9 | 47.7 | 33.0 | 28.0 | 62.2 |
| Self | SD-3.5-L | Qwen3-VL-30B | 29.6 | 16.6 | 5.1 | 25.6 | 3.1 | 47.9 | 33.1 | 30.6 | 62.5 |
| Self | SD-3.5-L | Qwen2.5-VL-72B | 21.6 | 10.4 | 3.8 | 18.5 | 1.8 | 47.7 | 33.7 | 28.6 | 63.1 |

marginal, and 8 rollouts even achieve better performance on CV-Bench. Considering the additional computational cost incurred by generating more rollouts, we choose 8 rollouts as a practical default.

### H.4 Preference selection judge ablation

Table 10 provides the complete benchmark results under different preference selection judge models. Since the judge selects preferred and dispreferred responses from multiple candidates, this ablation examines whether ViPSy depends on a particular judge model. The results remain stable across judges, indicating that the preference selection process is robust to the choice of judge model.

### H.5 Synthesis strategy ablation

Table 11 reports results for cue conditioning and visual variation strategies used for cue extraction. The no-cue setting still benefits from policy self-sampling, but it is less effective than cue-conditioned preference construction, indicating that grounded visual cues are important for guiding rollouts toward visually faithful responses. For visual variation, simple transformations such as translation, resizing, masking, and cropping improve performance but provide limited semantic diversity. Video-based variation uses an image-to-video model (Team, 2025b) to generate candidate frames from the input image. We then compute the cosine similarity between the visual features of

Table 9: Results on hallucination and general benchmarks using LLaVA-1.5-7B. We compare preference construction with different numbers of cue-conditioned rollouts. For each rollout setting, we report the mean and standard deviation over five runs.

| Number of Rollouts | Hallucination benchmarks | | | | | | General benchmarks | | |
|---|---|---|---|---|---|---|---|---|---|
| | Object HalBench | | AMBER | | | HallusionBench | MMStar | MMVP | CV-Bench |
| | Resp.↓ | Ment.↓ | CHAIR↓ | Hal.↓ | Cog.↓ | aAcc.↑ | Acc.↑ | Acc.↑ | Acc.↑ |
| Vanilla | 51.3 | 25.9 | 7.9 | 36.8 | 4.3 | 46.8 | 33.4 | 23.3 | 61.9 |
| 8 | $3.67 \pm 0.53$ | $2.54 \pm 0.26$ | $1.92 \pm 0.29$ | $9.68 \pm 1.52$ | $0.64 \pm 0.13$ | $48.25 \pm 0.67$ | $35.39 \pm 0.67$ | $30.53 \pm 1.85$ | $64.27 \pm 0.37$ |
| 2 | $4.73 \pm 0.55$ | $2.84 \pm 0.40$ | $2.82 \pm 0.34$ | $13.16 \pm 1.66$ | $0.78 \pm 0.13$ | $47.65 \pm 0.61$ | $34.00 \pm 0.21$ | $29.27 \pm 1.21$ | $63.06 \pm 0.62$ |
| 4 | $4.33 \pm 0.91$ | $2.86 \pm 0.53$ | $2.38 \pm 0.19$ | $11.90 \pm 1.23$ | $0.72 \pm 0.13$ | $47.55 \pm 0.52$ | $34.59 \pm 0.25$ | $29.47 \pm 1.66$ | $63.79 \pm 0.32$ |
| 16 | $3.33 \pm 0.33$ | $1.84 \pm 0.10$ | $2.08 \pm 0.04$ | $9.40 \pm 0.40$ | $0.64 \pm 0.05$ | $48.83 \pm 0.20$ | $35.41 \pm 0.10$ | $30.73 \pm 0.60$ | $64.02 \pm 0.19$ |

Table 10: Results on hallucination and general benchmarks using LLaVA-1.5-7B. We compare preference construction by varying the preference selection judge in ViPSy.

| Judge | Hallucination benchmarks | | | | | | General benchmarks | | |
|---|---|---|---|---|---|---|---|---|---|
| | Object HalBench | | AMBER | | | HallusionBench | MMStar | MMVP | CV-Bench |
| | Resp.↓ | Ment.↓ | CHAIR↓ | Hal.↓ | Cog.↓ | aAcc.↑ | Acc.↑ | Acc.↑ | Acc.↑ |
| GPT-5 mini | 3.3 | 2.2 | 1.9 | 9.6 | 0.5 | 48.1 | 34.5 | 27.3 | 64.7 |
| Qwen3-VL-30B | 3.3 | 2.2 | 2.0 | 9.8 | 0.7 | 47.7 | 35.0 | 28.0 | 64.4 |
| Qwen2.5-VL-72B | 4.0 | 2.6 | 1.8 | 9.6 | 0.6 | 48.1 | 35.4 | 30.6 | 64.4 |

the original image and each generated frame, and sample frames whose similarity is closest to 0.8. This selection encourages moderate visual variation while maintaining semantic consistency with the original image. However, despite this filtering, uncontrolled changes across generated frames can still make the extracted cues less reliable. In contrast, diffusion-based semantic synthesis produces richer semantic variations and achieves the strongest overall results, showing that semantically diverse visual variations help extract more consistent and reliable object-level cues.

### H.6 Variance of the ViPSy

In the main experiments, we follow prior work and use temperature 0 during inference, which removes randomness in generation and ensures a fair comparison with existing hallucination-mitigation methods. Since VLM outputs can vary when using a nonzero temperature, we further examine the inference-time variance of the model preference-aligned with ViPSy-constructed data. Specifically, we set the temperature to 0.7 and repeat inference five times. As shown in Table 12, which summarizes the mean and standard deviation across the five runs, the model preference-aligned with ViPSy-constructed data achieves lower mean hallucination scores than the vanilla model on Object HalBench and AMBER, while achieving higher mean scores on HallusionBench and general vision-language benchmarks. Overall, these results show that preference alignment with ViPSy-constructed data maintains its hallucination-mitigation effect while enhancing general vision-language capability under repeated inference with temperature 0.7.

## I Analysis of Cue-guided preference construction

**Effect of synthesis and voting.** Table 6 compares cues extracted once from the original image, cues aggregated from repeated extractions of the same image, cues aggregated from multiple self-captions, and cues constructed from semantically synthesized image variants. Direct extraction reduces Hal. Avg. from 7.14 to 6.17, while repeated extraction with voting improves cue correctness from 81% to 85% and further reduces Hal. Avg. to 5.94. Caption-only voting achieves 83% cue correctness and a Hal. Avg. of 6.18, whereas ViPSy achieves 95% cue correctness, a Hal. Avg. of 3.72, and a Gen. Avg. of 43.47. These results show that repeated sampling and voting explain only part of the improvement, while semantic image synthesis provides additional visual evidence for constructing more accurate cues.

Table 11: Results on hallucination and general benchmarks using LLaVA-1.5-7B. We compare preference construction without visual cues, simple image transformations (translation, resize, mask, and crop), video-based variation, and diffusion-based semantic synthesis.

| Strategy | Hallucination benchmarks | | | | | | General benchmarks | | |
|---|---|---|---|---|---|---|---|---|---|
| | Object HalBench | | AMBER | | | HallusionBench | MMStar | MMVP | CV-Bench |
| | Resp.↓ | Ment.↓ | CHAIR↓ | Hal.↓ | Cog.↓ | aAcc.↑ | Acc.↑ | Acc.↑ | Acc.↑ |
| Vanilla | 51.3 | 25.9 | 7.9 | 36.8 | 4.3 | 46.8 | 33.4 | 23.3 | 61.9 |
| w/o cue | 7.6 | 3.9 | 3.3 | 19.7 | 1.2 | 47.1 | 34.3 | 28.6 | 64.6 |
| Translation | 5.3 | 4.6 | 2.3 | 12.5 | 0.8 | 48.6 | 34.3 | 30.6 | 64.5 |
| Resize | 4.3 | 3.2 | 2.4 | 13.4 | 0.8 | 47.2 | 35.0 | 26.6 | 65.8 |
| Mask | 5.0 | 3.7 | 2.4 | 12.5 | 0.5 | 47.2 | 34.9 | 28.0 | 64.7 |
| Crop | 5.0 | 3.9 | 2.3 | 11.8 | 0.8 | 47.6 | 34.6 | 28.0 | 64.7 |
| Video | 4.3 | 3.4 | 2.3 | 11.5 | 0.8 | 48.0 | 33.2 | 28.0 | 63.9 |
| Diffusion | 4.0 | 2.6 | 1.8 | 9.6 | 0.6 | 48.1 | 35.4 | 30.6 | 64.4 |

Table 12: Variance of ViPSy on LLaVA-1.5-7B. Results are computed over five inference runs with temperature 0.7 and reported as mean and standard deviation.

| Method | Hallucination benchmarks | | | | | | General benchmarks | | |
|---|---|---|---|---|---|---|---|---|---|
| | Object HalBench | | AMBER | | | HallusionBench | MMStar | MMVP | CV-Bench |
| | Resp.↓ | Ment.↓ | CHAIR↓ | Hal.↓ | Cog.↓ | aAcc.↑ | Acc.↑ | Acc.↑ | Acc.↑ |
| Vanilla | $53.0 \pm 1.9$ | $27.7 \pm 1.1$ | $9.3 \pm 0.8$ | $41.7 \pm 3.0$ | $4.3 \pm 0.2$ | $46.4 \pm 0.3$ | $33.0 \pm 0.8$ | $27.0 \pm 2.4$ | $57.8 \pm 2.4$ |
| ViPSy (**Ours**) | $4.1 \pm 0.4$ | $2.8 \pm 0.5$ | $2.0 \pm 0.3$ | $10.2 \pm 1.3$ | $0.6 \pm 0.1$ | $47.8 \pm 0.8$ | $34.4 \pm 0.9$ | $29.8 \pm 1.1$ | $64.6 \pm 1.4$ |

**Consensus ablation for cue aggregation.** We use Agree=8 by default and vary the agreement threshold from one to eight to examine cue reliability. As shown in Table 13, increasing the required agreement from one to eight improves cue correctness from 78% to 95%, reduces Hal. Avg. from 8.30 to 3.72, and increases Gen. Avg. from 41.69 to 43.47. Requiring only one agreeing cue performs worse than the no-cue baseline on both average hallucination and general benchmark scores, indicating that inaccurate cues can misguide cue-conditioned generation. The consistent improvement with stronger agreement shows that reliable consensus is important for cue construction.

## J Dependence on External Models

**Sensitivity to cue extractors.** To examine sensitivity to the choice of cue extractor, we replace Qwen2.5-VL-72B with Qwen3-VL-8B or Qwen2.5-VL-7B for pairwise cue extraction. Table 14 reports Hal. Avg. of 4.04 and 4.12 for the two alternative extractors, compared with 3.72 for Qwen2.5-VL-72B and 7.14 without cues. Gen. Avg. is 43.26 and 43.12, respectively, compared with 43.47 for Qwen2.5-VL-72B and 42.50 without cues. These results indicate that the alternative extractors maintain comparable average hallucination and general benchmark scores. This suggests that ViPSy does not strongly depend on a particular cue extractor, as the improvements are largely maintained across different models.

**Comparison under the same judge models.** We additionally compare HA-DPO (Zhao et al., 2023b), SENTINEL (Peng et al., 2025), OPA-DPO (Yang et al., 2025), and ViPSy using the same judge models. Table 15 reports results with GPT-5 mini, Qwen3-VL-30B, and Qwen2.5-VL-72B. ViPSy achieves the lowest Hal. Avg. and highest Gen. Avg. among the compared methods across all three judges. Under Qwen2.5-VL-72B, ViPSy achieves a Hal. Avg. of 3.72 and a Gen. Avg. of 43.47, compared with 5.56 and 39.59 for OPA-DPO. These results indicate that ViPSy's preference construction remains effective across different judge models.

Table 13: Ablation on consensus for cue aggregation with average hallucination and general benchmark scores. Cue correctness is measured against Visual Genome annotations.

| Configuration | Cue correctness (%) ↑ | Hal. Avg. ↓ | Gen. Avg. ↑ |
|---|---|---|---|
| No cue | – | 7.14 | 42.50 |
| Agree = 1 | 78 | 8.30 | 41.69 |
| Agree = 2 | 86 | 5.92 | 42.80 |
| Agree = 4 | 90 | 4.32 | 42.94 |
| Agree = 8 | **95** | **3.72** | **43.47** |

Table 14: Effect of different pairwise cue extractors on average hallucination and general benchmark scores.

| Pairwise cue extractor | Hal. Avg. ↓ | Gen. Avg. ↑ |
|---|---|---|
| No cue | 7.14 | 42.50 |
| Qwen2.5-VL-7B | 4.12 | 43.12 |
| Qwen3-VL-8B | 4.04 | 43.26 |
| Qwen2.5-VL-72B | **3.72** | **43.47** |

Table 15: Comparison across different judge models with average hallucination and general benchmark scores.

| | GPT-5 mini | | Qwen3-VL-30B | | Qwen2.5-VL-72B | |
|---|---|---|---|---|---|---|
| Method | Hal. Avg. ↓ | Gen. Avg. ↑ | Hal. Avg. ↓ | Gen. Avg. ↑ | Hal. Avg. ↓ | Gen. Avg. ↑ |
| HA-DPO | 17.71 | 41.87 | 8.61 | 41.67 | 7.90 | 40.40 |
| SENTINEL | 9.69 | 41.62 | 6.74 | 40.93 | 5.98 | 40.29 |
| OPA-DPO | 8.94 | 41.14 | 5.87 | 39.26 | 5.56 | 39.59 |
| ViPSy | **3.50** | **42.17** | **3.60** | **42.47** | **3.72** | **43.47** |

## K Quantitative Analysis of Preference Pairs and Responses

**Quantifying the distributional shift.** Table 16 measures how far the preference responses constructed by ViPSy and SENTINEL deviate from the response distribution of the vanilla policy. For the same image and prompt, we compute token-level $\mathrm{KL}(\pi_{\text{method}} \| \pi_{\text{vanilla}})$, where $\pi_{\text{vanilla}} = \pi_\theta(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{m})$, and average it over the full preference dataset constructed by each method. A lower KL divergence indicates that the constructed responses remain closer to the vanilla policy distribution. We additionally report the change in average token log-likelihood and perplexity relative to the vanilla policy, where smaller absolute changes indicate less deviation from the vanilla policy distribution. ViPSy yields a KL divergence of 0.018, compared with 0.074 for SENTINEL. Its log-likelihood change is also smaller in magnitude at $-0.022$ compared with $-0.154$, and its perplexity increase is smaller at $+0.046$ compared with $+0.302$. These results provide empirical evidence that ViPSy constructs preference pairs with substantially smaller distributional deviation from the base policy while incorporating visual evidence.

**Preferred versus dispreferred responses.** We compare the preferred and dispreferred responses constructed by ViPSy. As shown in Table 17, preferred and dispreferred responses show similar lengths and object counts, with 88.94 and 90.65 words and 7.57 and 7.70 objects, respectively. This indicates that the judge does not simply select responses based on length or the number of mentioned objects. For response quality evaluation, we provide GPT-5 mini with the original image, the corresponding response, and questions on visual grounding, relational accuracy, attribute accuracy, spatial accuracy, and detail informativeness, and obtain a score between 0 and 1 for each aspect. Preferred responses achieve substantially higher scores across all five aspects. Visual grounding increases from 0.441 to 0.863 and spatial accuracy from 0.521 to 0.821. These results suggest that the generated candidates differ meaningfully in visual correctness, rather than only in response length or object count.

Table 16: Distributional shift of preference pairs. KL divergence is measured against the vanilla policy, while log-likelihood and perplexity are reported as changes from the vanilla policy. Lower KL divergence and smaller absolute changes indicate less deviation.

| Method | $\mathrm{KL}(\pi_{\mathrm{method}} \| \pi_\theta) \downarrow$ | $\Delta$ Log-likelihood | $\Delta$ Perplexity |
|---|---|---|---|
| SENTINEL | 0.074 | $-0.154$ | $+0.302$ |
| ViPSy | **0.018** | $\mathbf{-0.022}$ | $\mathbf{+0.046}$ |

Table 17: Quantitative analysis of preferred and dispreferred responses.

| | Response quality | | | | | Response statistics | |
|---|---|---|---|---|---|---|---|
| Response type | Visual grounding ↑ | Relational accuracy ↑ | Attribute accuracy ↑ | Spatial accuracy ↑ | Detail informativeness ↑ | Word count | Object count |
| Dispreferred | 0.441 | 0.507 | 0.554 | 0.521 | 0.514 | $90.65 \pm 20.32$ | $7.70 \pm 2.95$ |
| Preferred | **0.863** | **0.854** | **0.869** | **0.821** | **0.753** | $88.94 \pm 20.44$ | $7.57 \pm 2.61$ |

Table 18: Quantitative analysis of generated responses on AMBER after preference alignment.

| Method | Avg. objects | Object recall (%) | Avg. words | Avg. characters |
|---|---|---|---|---|
| Vanilla | 8.26 | 51.7 | 76.26 | 432.38 |
| SENTINEL | 7.23 | 43.7 | 70.07 | 383.83 |
| OPA-DPO | 6.94 | 46.6 | 71.65 | 406.14 |
| ViPSy | 7.45 | 50.1 | 74.26 | 419.17 |

**Response statistics after preference alignment.** We analyze responses generated after preference alignment on the AMBER benchmark to examine whether hallucination reduction is accompanied by reduced descriptive detail. As shown in Table 18, ViPSy produces 74.26 words and 7.45 object mentions per response, compared with 76.26 words and 8.26 object mentions for the vanilla policy, while achieving an object recall of 50.1% compared with 51.7% for vanilla. Compared with SENTINEL and OPA-DPO, ViPSy generates longer responses, mentions more objects, and retains higher object recall while achieving lower hallucination scores. These results indicate that ViPSy's hallucination reduction is not simply explained by generating shorter or less detailed responses.

## L Beyond Object-level Visual Grounding

**Cue granularity.** We extend visual cues beyond objects to spatial relationships and actions. Action cues describe relations such as "a person riding a bicycle," while spatial cues capture relations such as "a cup on a table." As shown in Table 19, spatial cues achieve the highest Gen. Avg. of 43.83 and MMVP accuracy of 32.6, while object cues achieve the lowest Hal. Avg. of 3.72 and the highest HallusionBench accuracy of 48.1. These results show that spatial information can further improve visual reasoning, while simple object cues remain highly competitive. Moreover, Table 17 shows that preferred responses improve in relational, attribute, and spatial accuracy even when candidates are generated with the same object cue. This suggests that object cues provide grounding for rollout generation, while preference selection captures broader forms of visual correctness.

**Additional benchmark evaluation.** To evaluate whether ViPSy generalizes beyond object-presence hallucination, we further evaluate it on MMHal (Sun et al., 2024), MMBench (Liu et al., 2024d), and AMBER-D, the discriminative subset of AMBER (Wang et al., 2023a). Table 20 shows that ViPSy improves over the vanilla policy across all three benchmarks. The MMHal average score increases from 2.10 to 2.54, while its hallucination rate decreases from 64.58 to 51.04. On AMBER-D, accuracy increases from 71.5 to 80.1 and F1 from 74.1 to 83.7, while MMBench accuracy improves from 74.24 to 75.08. These results suggest that the benefits of ViPSy extend beyond object-presence hallucination. Although the extracted cues are object-level, they guide the generation of complete responses, allowing the resulting preference pairs to capture broader aspects of image-grounded correctness, including attributes, relations, and contextual details.

Table 19: Comparison of object, action, and spatial relationship cues.

| Cue configuration | Hal. Avg. ↓ | HallusionBench ↑ | Gen. Avg. ↑ | MMVP ↑ |
|---|---|---|---|---|
| Vanilla | 25.24 | 46.8 | 39.53 | 23.3 |
| Action | 5.06 | 46.8 | 42.62 | 27.3 |
| Spatial relationship | 4.03 | 47.2 | **43.83** | **32.6** |
| Object | **3.72** | **48.1** | 43.47 | 30.6 |
| Object + Spatial relationship | 3.98 | 47.6 | 43.60 | 31.3 |

Table 20: Additional evaluation on MMHal, AMBER-D, and MMBench.

| | MMHal | | AMBER-D | | MMBench |
|---|---|---|---|---|---|
| Method | Avg. ↑ | Hal. ↓ | Acc. ↑ | F1 ↑ | Acc. ↑ |
| Vanilla | 2.10 | 64.58 | 71.5 | 74.1 | 74.24 |
| ViPSy | **2.54** | **51.04** | **80.1** | **83.7** | **75.08** |

## M Offline Computational Cost and Efficiency

**Computational cost of preference construction.** We measure offline preference construction on two NVIDIA B200 GPUs. As shown in Table 21, the default ViPSy configuration constructs 7,000 preference pairs in 49.4 GPU-hours, compared with 56.2 GPU-hours for TPR (He et al., 2024), a state-of-the-art intervention-based method, which constructs 21,376 pairs. With these datasets, ViPSy achieves a Hal. Avg. of 3.72 compared with 8.50 for TPR, and a Gen. Avg. of 43.47 compared with 42.03. This indicates that ViPSy is more cost-effective than TPR at the dataset level.

**Number of synthetic variants.** Reducing the number of synthetic variants $N_c$ provides a direct way to lower construction cost. With four, two, and one variant, the total costs are 26.5, 14.4, and 9.0 GPU-hours, respectively. The corresponding Hal. Avg. values are 4.01, 4.46, and 4.99, compared with 3.72 for eight variants. Gen. Avg. decreases from 43.47 to 42.27 when reducing $N_c$ from eight to one. Even the configuration using only one variant achieves better average hallucination and general benchmark scores than TPR in this comparison, while requiring substantially less total construction time.

**Lighter synthesis backbones.** Table 22 reports the computation cost of each component. Diffusion-based image synthesis accounts for most of the cost, requiring 40.37 GPU-hours with SD-3.5-L, compared with 19.41 and 7.88 GPU-hours with SD-3.5-M and SD-1.4. The remaining components require 9.03 GPU-hours in total. Accordingly, replacing SD-3.5-L with SD-3.5-M or SD-1.4 reduces the total construction cost from 49.4 to 28.4 or 16.9 GPU-hours, respectively. Their Hal. Avg. values are 3.86 and 4.17, while their Gen. Avg. values are 42.87 and 42.33. Thus, lighter synthesis backbones provide a practical option for reducing offline construction cost while maintaining competitive average hallucination and general benchmark scores.

## N Qualitative Results

We provide additional qualitative comparisons to illustrate how ViPSy constructs preference data and how the aligned model behaves after training. We also provide examples of our proposed self-captioned semantic synthesis pipeline, which visualizes how synthetic image variations are generated from policy-generated captions.

### N.1 Preference Data Examples

Figure 6 shows additional examples of preference pairs constructed by ViPSy. Each example contains the image, the prompt, the extracted visual cue, and the selected preferred and dispreferred responses. The preferred responses describe objects and relations supported by the image and visual cue, whereas the dispreferred responses include information that is not present in the image.

Table 21: Offline preference construction cost and average hallucination and general benchmark scores. Construction costs were measured on two NVIDIA B200 GPUs.

| Configuration | Construction cost (GPU-hours) | Preference pairs | Hal. Avg. ↓ | Gen. Avg. ↑ |
|---|---|---|---|---|
| TPR | 56.2 | 21,376 | 8.50 | 42.03 |
| ViPSy ($N_c = 8$) | 49.4 | 7,000 | **3.72** | **43.47** |
| ViPSy ($N_c = 4$) | 26.5 | 7,000 | 4.01 | 43.08 |
| ViPSy ($N_c = 2$) | 14.4 | 7,000 | 4.46 | 42.90 |
| ViPSy ($N_c = 1$) | 9.0 | 7,000 | 4.99 | 42.27 |
| ViPSy (SD-3.5-M) | 28.4 | 7,000 | 3.86 | 42.87 |
| ViPSy (SD-1.4) | 16.9 | 7,000 | 4.17 | 42.33 |

Table 22: Detailed offline preference construction cost of each component for the 7,000 preference pairs. The three synthesis backbones are alternative configurations, while the remaining components are shared across configurations.

| Stage | Time / sample (ms) | Total GPU-hours |
|---|---|---|
| Self-caption generation | $129 \pm 18$ | 2.05 |
| Diffusion-based synthesis (SD-3.5-L) | $2{,}575 \pm 564$ | 40.37 |
| Diffusion-based synthesis (SD-3.5-M) | $1{,}328 \pm 317$ | 19.41 |
| Diffusion-based synthesis (SD-1.4) | $617 \pm 81$ | 7.88 |
| Visual cue extraction | $237 \pm 45$ | 3.64 |
| Cue-conditioned rollout | $107 \pm 36$ | 1.98 |
| Judge-based preference selection | $694 \pm 89$ | 1.36 |

These examples show that cue-conditioned self-rollouts produce candidate responses with different degrees of visual grounding, enabling the judge to select informative preference pairs for preference alignment.

### N.2 Image description examples

Figure 7 compares image descriptions from the vanilla model and the model preference-aligned using preference pairs constructed by ViPSy. The vanilla model often includes hallucinated details that are not supported by the image. In contrast, the model trained with ViPSy produces descriptions that are better grounded in the image. This supports the effectiveness of the proposed preference construction in reducing hallucination.

### N.3 Self-captioned semantic synthesis examples

Figures 8 and 9 illustrate examples of self-captioned semantic synthesis. For each source image, the policy first generates a self-caption, which is then used as a prompt for text-to-image synthesis. Although the synthetic images vary in appearance, recurring objects and scene elements across the original and synthetic images provide reliable evidence for constructing the visual cue. These examples show how semantic diversity from synthesis helps ViPSy extract image-grounded object-level cues.

## O Prompts

We list prompts used in ViPSy for preference construction. For self-captioning, we use the prompt "*Describe the image in detail.*" The pairwise comparison prompt $\boldsymbol{x}_{\text{comp.}}$ in Figure 10, the aggregation prompt $\boldsymbol{x}_{\text{agg.}}$ in Figure 11, the cue-conditioned captioning prompt $\boldsymbol{x}_{\text{cue-cap.}}$ in Figure 12, and the visual grounding prompt $\boldsymbol{x}_{\text{vis.}}$ in Figure 13 are used for cue extraction and preference construction.

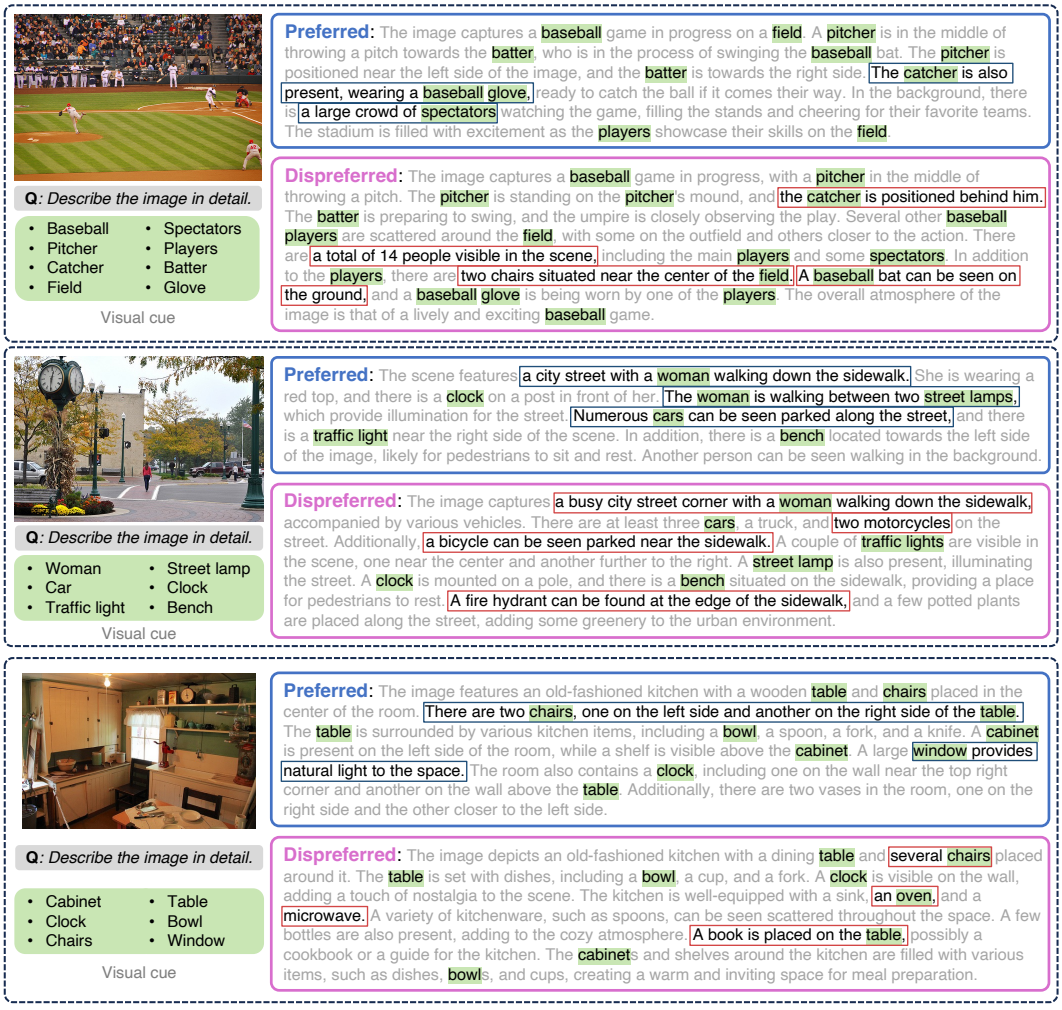


Figure 6: Qualitative examples of cue-conditioned preference pairs. Given the same image, prompt, and visual cue, preferred responses remain image-grounded, while dispreferred ones hallucinate.

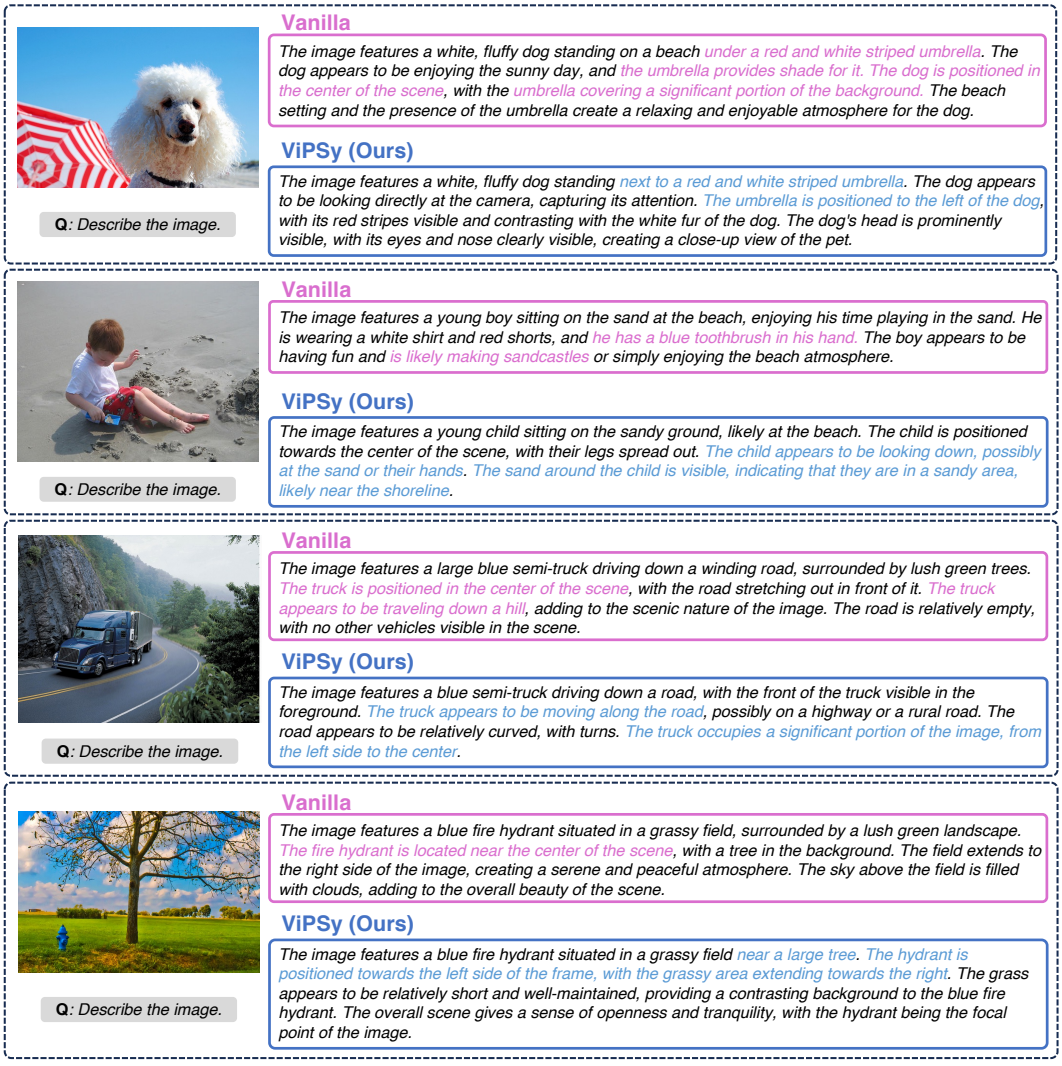


Figure 7: Qualitative comparison between the vanilla model and the model preference-aligned using preference pairs constructed by ViPSy. After preference alignment, the aligned model produces image-grounded descriptions, whereas the vanilla model describes content that is not present in the image. Highlights indicate image-inconsistent descriptions for the vanilla model and grounded descriptions for the preference-aligned model.




Figure 8: Examples of self-captioned semantic synthesis. The source image is first captioned by the policy model, and the caption is used to generate semantically aligned synthetic images using the text-to-image models SD-3.5-L, SD-3.5-M, Flux-1.0-dev, and PixelDiT.




Figure 9: Examples of self-captioned semantic synthesis. The source image is first captioned by the policy model, and the caption is used to generate semantically aligned synthetic images using the text-to-image models SD-3.5-L, SD-3.5-M, Flux-1.0-dev, and PixelDiT.

Compare two images. The first image is the reference image. The second image is the comparison image. Keep only clearly shared facts. Do not include differences, unique details or extra keys. Keep phrases short, factual and reusable. If uncertain, omit it.

{*an image pair*}

Figure 10: Pairwise comparison prompt.

You are given pairwise comparison results between one reference image and multiple comparison images. Each pairwise result contains shared facts. Merge similar phrases, remove near-duplicates, and produce one consolidated result.

Rules:
- chosen should contain the merged shared content from the pairwise chosen results.
- Keep only content that is supported as shared visual content across the pairwise chosen results.
- Merge synonymous or near-duplicate phrases into one natural phrase.
- Prefer short, reusable, neutral wording.
- Do not invent new facts that are not supported by the pairwise results.
- Do not mention pair indices, or phrases like 'in comparison image'.
- If a fact is uncertain, omit it.

{*pairwise visual cues*}

Figure 11: Aggregation prompt.

Describe the image in detail.
Please use the following object information as extra context: {*visual cues*}

Figure 12: Cue-conditioned captioning prompt.

You are evaluating candidate responses for DPO data construction.
You are given:

1. The original image
2. Multiple candidate responses generated from repeated inference runs on the same original image
Important:
- Use the original image as the main visual evidence.
Your task:
Select:
- the single best candidate response
- the single worst candidate response
Selection criteria:
- The best candidate should be the most visually grounded to the original image and should avoid unsupported details.
- The worst candidate should be the least grounded, adding unsupported objects, people, actions, attributes, or scene details.
- Prefer a worst candidate that is plausible but likely unsupported rather than one that is merely short or generic.

{*candidates*}

Figure 13: Visual grounding prompt.