Nullu: Mitigating Object Hallucinations in Large Vision-Language Models via HalluSpace Projection

Le Yang*, Ziwei Zheng*, Boxu Chen, Zhengyu Zhao, Chenhao Lin, Chao Shen†
Xi'an Jiaotong University
Xi'an, 710049, China
{yangle15@, ziwei.zheng@stu., chenboxu@stu., zhengyu.zhao@}xjtu.edu.cn
{linchenhao@, chaoshen@mail.}xjtu.edu.cn

Abstract

Recent studies have shown that large vision-language models (LVLMs) often suffer from the issue of object hallucinations (OH). To mitigate this issue, we introduce an efficient method that edits the model weights based on an unsafe subspace, which we call HalluSpace in this paper. With truthful and hallucinated text prompts accompanying the visual content as inputs, the HalluSpace can be identified by extracting the hallucinated embedding features and removing the truthful representations in LVLMs. By orthogonalizing the model weights, input features will be projected into the Null space of the HalluSpace to reduce OH, based on which we name our method Nullu. We reveal that HalluSpaces generally contain prior information in the large language models (LLMs) applied to build LVLMs, which have been shown as essential causes of OH in previous studies. Therefore, null space projection suppresses the LLMs' priors to filter out the hallucinated features, resulting in contextually accurate outputs. Experiments show that our method can effectively mitigate OH across different LVLM families without extra inference costs and also show strong performance in general LVLM benchmarks. Code is released at https://github.com/Ziwei-Zheng/Nullu.

1. Introduction

Recent advancements in large language models (LLMs) [1, 35, 40] have spurred rapid developments in large vision-language models (LVLMs), such as GPT-4V [1], Gemini [39], LLaVA [31] and mPLUG-Owl2 [48]. By integrating a vision encoder and fine-tuning on multimodal instruction-following datasets, LVLMs exhibit a strong ability to interpret and convert complex visual patterns into coherent linguistic representations. Whereas, the widespread use of these LVLMs has shed light on their limitations—object hallucinations (OH).

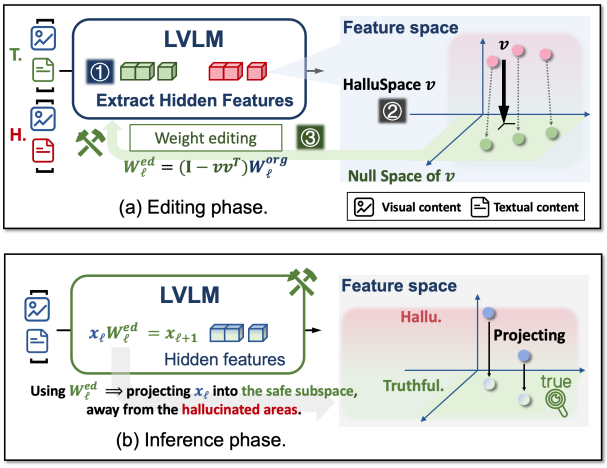


Figure 1. An illustration of Nullu. (a) In the editing phase, Nullu will ① Extract hidden features of truthful (T.) and hallucinated (H.) inputs. ② Explore a low-rank HalluSpace v in the feature space by contrasting the differences between T. and H. features. ③ Edit the model weights by projecting them to the null space of v. (b) In the inference phase, using the edited weights equals to project input features into the safe subspace, away from the hallucinated areas, leading to contextually accurate outputs.

Object Hallucination (OH) refers to the phenomenon where LVLMs generate text that inaccurately represents the actual objects in the accompanying image [4, 20, 23, 27, 29]. Considering the wide application of LVLMs in different fields, hallucinations in generated content can lead to misinformation or even harmful decisions. Therefore, mitigating OH is crucial for AI safety and trustworthiness. Recent works propose to address OH by either performing fine-tuning in an end-to-end manner [20, 22, 29] or using post-processing methods to modify model outputs [9, 23, 50, 53], which show effectiveness in mitigating OH for modern open-source LVLMs.

∗ Equal contribution. † Corresponding author.

Despite significant efforts, the underlying causes of OH within the models remain unexplored. This study investigates this issue by systematically examining the intrinsic distinctions between truthful and hallucinated responses, focusing on feature spaces. Our empirical results on open-source LVLMs reveal a low-rank subspace in the model parameter space, which we term "HalluSpace", representing the distinction between truthful and untruthful feature distributions, and can be a factor triggering OH.

Building on this finding, we mitigate OH by projecting the features away from the detected HalluSpaces. Namely, we project the original model weights to the null space of HalluSpaces, based on which we build a novel method called Nullu (shown in Figure 1). Our method first applies Principal Component Analysis (PCA) to extract the low-dimensional subspaces (HalluSpaces) from the difference matrix between truthful and hallucinated features. Then, by orthogonalizing model weights with respect to the HalluSpaces, Nullu effectively nullifies OH representations for inputs, filtering out these untruthful features and resulting in contextually more accurate outputs.

We also found that these HalluSpace are closely tied to the priors in LLMs, which is primary causative factors of OH in [23]. Therefore, projecting model weights to the null space of HalluSpace offers a distinct but straightforward approach to reducing these priors. Furthermore, we establish connections between Nullu and Direct Preference Optimization (DPO) [36], showing that both of these two methods attempt to achieve a similar goal to avoid OH.

Through comprehensive experiments, we demonstrate the effectiveness of Nullu in mitigating OH without compromising the models' general performance. Notably, Nullu needs no additional fine-tuning procedure and, introduces no extra inference costs compared to most post-hoc based methods [9, 23, 50], making Nullu much more efficient in alleviating OH. Our main contributions are as follows:

1. We propose a novel method named Nullu for OH mitigation. By extracting HalluSpaces and orthogonalizing the model weights, Nullu can effectively mitigate OH with no extra inference cost. Our method demonstrates the feasibility and effectiveness of mitigating OH by editing weight based on representation learning.
2. We show that HalluSpaces appear to form a prior closely related to the priors in LLMs causing OH. Therefore, our method coincides with previous studies debiasing LVLMs to reduce OH at outputs. Moreover, we show the connection between Nullu and DPO, which can provide nuanced insights, broadening the understanding of potential causes behind unsafe behaviors within LVLMs.
3. Experiments show that using Nullu achieves consistent improvements with zero-extra costs on the evaluated tasks across different open-source LVLMs, demonstrating the effectiveness and efficiency of Nullu.

2. Related Work

2.1. Large Visual-Language Models (LVLMs)

Based on the development of LLMs, such as LLaMA [41] and Vicuna [10], large vision-language models (LVLMs) have made significant advancements in recent years. Early works, such as BLIP [24, 25] and BERT-based VLMs [12, 34], have successfully adapted LLMs to visual tasks, demonstrating notable generative capabilities and in-context learning abilities. These advancements typically involve connecting a vision encoder to an LLM through various fused modules, such as linear projection [31] or a Q-former [54]. With the adoption of visual instruction tuning techniques, LVLMs [31, 52] have further enhanced their abilities, enabling language generation models to perform complex image understanding and reasoning tasks, resulting in a series of modern well-performed LVLMs, such as LLaVA [30, 31], Mini-GPT4 [54], mPLUG-Owl [47, 48], Shikra [6], InternVL [8] and Qwen-VL [2, 3]. However, despite these successes, recent LLMs and LVLMs still face security issues [21, 44, 51], such as hallucination [4],

2.2. Mitigation of Object Hallucination

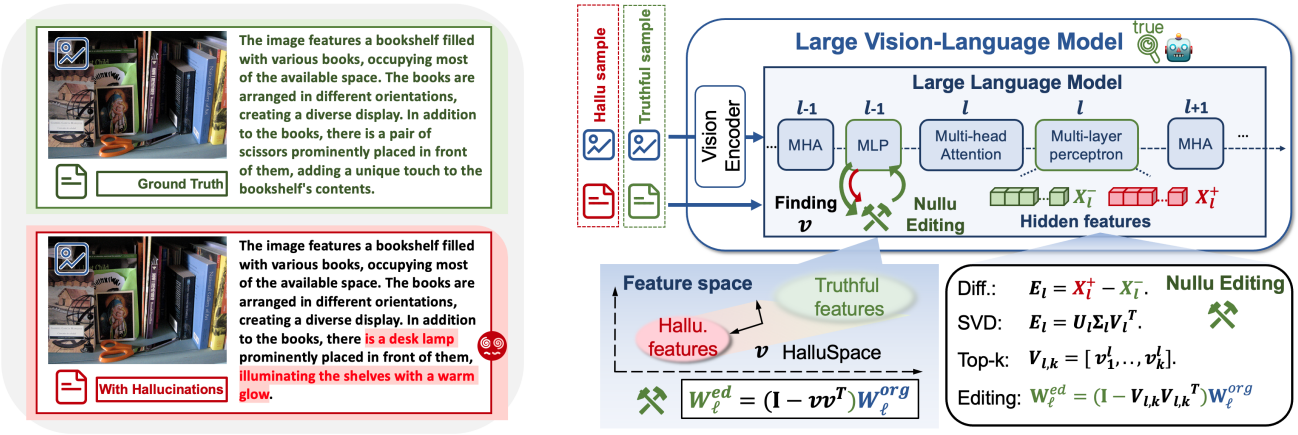
Object hallucination occurs when the model generates semantically coherent texts misaligned with actual objects in the accompanying image [19, 37, 45, 46]. Various approaches have been proposed to address this issue. Considering a possible cause of hallucination lies in data biases and the knowledge gap between visual and linguistic information, recent studies propose to fine-tune the LVLMs for robustness [17, 29], cross-modality matching [20, 22] or preference alignments [7, 38]. While these methods are effective, they are notorious for the demands of substantial computational resources for training and fine-tuning.

Researchers also propose post-processing methods to avoid high computational costs, which apply new strategies or external tools to modify the response. LURE [53] trains a reviser to edit the possible hallucinated words in the responses. Other methods incorporate an external visual model improved answers [5, 9, 49]. Moreover, recent studies have revealed that the LVLMs are often affected by the strong LLM's priors and visual uncertainty, leading to less attention to the visual contexts and resulting in OH [13, 18, 23, 33, 50, 55].

Compared to these works, Nullu mitigates OH by a new editing strategy that contrasts the truthful and hallucinated features, resulting in an effective and efficient solution for OH issues.

2.3. Feature steering

In the LLM and LVLM fields, some works use a similar approach of measuring the "contrast" between the internal features of positive and negative samples and then modify-



(a) Paired truthful and hallucinated samples.

(b) Editing LVLM weights with Nullu.

Figure 2. An overview of Nullu, which identifies the HalluSpaces to edit model weights for LVLMs. (a) The paired truthful and hallucinated samples. (b) Nullu first calculates the difference matrix of hidden features for the paired samples and then conducts the SVD to find the main directions of the difference as the HalluSpace. Then Nullu projects the original MLP's weights to the null space of the HalluSpace. This procedure will be repeated for a series of layers, $\{\ell\}$, in the LLM of an LVLM.

ing features at inference time using this "contrast" to adjust the model's behavior. E.g., ITI [26], ActADD [42], and VTI [32] shift the features via an obtained direction for truthfulness, detoxification, and OH mitigation, respectively. Our method projects the features for editing. Although recent work [43] also proposes the null space projection strategy for editing, its target is the detoxification of LLMs, which differs from ours.

3. Method

The framework of Nullu is illustrated in Figure 2. We first describe how to construct the data pairs of truthful and hallucinated text prompts, then introduce how Nullu extracts the HalluSpace to edit the model weights. We further provide an in-depth analysis by decoding the HalluSpaces, and discuss the connection between Nullu and DPO.

3.1. Data-pair construction

We begin with constructing paired vision-language inputs as shown in Figure 2 (a). The two inputs have the same image but different text prompts, where one contains the truthful ground truth, x_i^-, that accurately describes the objects in the image, and the other contains hallucinated descriptions, x_i^+. The whole dataset can be represented by $\mathcal{D} = (x_i^+, x_i^-)_{i=1}^N$.

As stated in [53], the hallucinatory descriptions can be generated by modifying the accurate descriptions using GPT-3.5. These adjustments are guided by factors related to object hallucination, including co-occurrence, object uncertainty, and object position. Then, we can obtain the data pair whose sample contains an image with a GT description and the same image with a hallucinated description by substituting the objects in the GT description with the most likely hallucinated ones.

3.2. Nullu

Exploring HalluSpaces. At each model layer ℓ in the LLM of the LVLM, where $\ell \in L_0 \dots L$, we compute embedding features, denoted as $\boldsymbol{x}_{i,\ell}^+$ and $\boldsymbol{x}_{i,\ell}^-$ for the hallucinated and truthful features, respectively. We then average each sample over the token dimension and stack these embedding features into matrices $\boldsymbol{X}_\ell^+, \boldsymbol{X}_\ell^- \in \mathbb{R}^{N\times D}$, where D is the embedding dimension. Then the difference matrix $\boldsymbol{E}_\ell$ at ℓ-th layer can be calculated by:

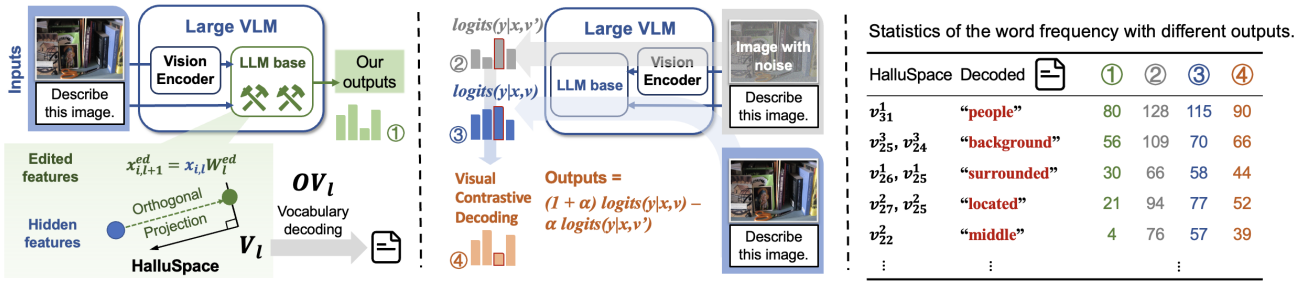
$$\boldsymbol{E}_\ell = \boldsymbol{X}_\ell^+ - \boldsymbol{X}_\ell^- \ , \quad \boldsymbol{E}_\ell \in \mathbb{R}^{N\times D}. \tag{1}$$

We conduct the principal component analysis for $\boldsymbol{E}_\ell$ via singular value decomposition (SVD). As shown in Figure 2 (b), this matrix contains the underlying differences between the truthful and hallucinated embeddings in the feature space. Therefore, by exploring the structure of $\boldsymbol{E}_\ell$, we can find the main directions as a low-rank approximation of $\boldsymbol{E}_\ell$, based on which we can effectively eliminate the hallucinated features. Formally, the SVD can be conducted by

$$\boldsymbol{E}_\ell = \boldsymbol{U}_\ell \boldsymbol{\Sigma}_\ell \boldsymbol{V}_\ell^\top, \ \boldsymbol{U}_\ell \in \mathbb{R}^{N\times N}, \ \boldsymbol{V}_\ell \in \mathbb{R}^{D\times D}, \tag{2}$$

where $\boldsymbol{\Sigma}_\ell$ is the diagonal matrix containing the singular values with the descending sort.

Next, we pick the right singular vectors with top-k singular values, $\boldsymbol{v}_1^\ell, \boldsymbol{v}_2^\ell, \dots, \boldsymbol{v}_k^\ell$, which are the first k column vectors of $\boldsymbol{V}_\ell$. These directions represent the main difference between the truthful and hallucinated features and, therefore, can be considered the directions in model weight spaces that trigger the hallucinated descriptions. Therefore $\boldsymbol{V}_{\ell,k} = [\boldsymbol{v}_1, ..., \boldsymbol{v}_k] \in \mathbb{R}^{D\times k}$ is the target HalluSpace.



(a) Vocabulary decoding for HalluSpace. (b) Debiasing via VCD. (c) Word frequency.

Figure 3. The relation between Nullu and other debiasing methods [23, 50]. The inference procedure of (a) ours and (b) VCD [23]. (c) The statistics of the word frequency in outputs of Nullu (①), with LLM priors (②), LLaVA (③) and VCD (④).

Usually, with a $k \ll D$, the HalluSpace can be well defined, which means the hallucinated descriptions are possible within a low-rank dimensionality. We then show how to edit the model weights to mitigate OH in LVLMs.

Editing the model weights. As the $\boldsymbol{V}_{\ell,k}$ represents the main different directions between the truthful and untruthful data distributions, we can effectively remove the hallucination information from the model features by projecting them to the null space of HalluSpace, which can be achieved by orthogonalizing the model weights with respect to $\boldsymbol{V}_{\ell,k}$. The null space of $\boldsymbol{V}_{\ell,k}$ can be presented by $(\boldsymbol{I} - \boldsymbol{V}_{\ell,k}\boldsymbol{V}_{\ell,k}^\top)$[1].

Therefore, we project the original weights of MLP at ℓ-th layer $\boldsymbol{W}_\ell^{\text{org}}$ to the null-space of $\boldsymbol{V}_{\ell,k}$ with

$$\boldsymbol{W}_\ell^{\text{ed}} = (\boldsymbol{I} - \boldsymbol{V}_{\ell,k}\boldsymbol{V}_{\ell,k}^\top)\, \boldsymbol{W}_\ell^{\text{org}} . \tag{3}$$

With the $\boldsymbol{W}_\ell^{\text{ed}}$, Nullu can effectively mitigate OH while generating responses. As the new weights can be reloaded into the model, there is no extra cost during inference, making our method computationally efficient.

Algorithm 1 Nullu

Input: Paired data $\mathcal{D}$, LVLM $\mathcal{M}$, target layers $\{\ell\}$, rank number k.
Output: Edited LVLM $\mathcal{M}^{ed}$.
for ℓ in $\{\ell\}$ **do**
  Calculating $\boldsymbol{X}_\ell^+$ and $\boldsymbol{X}_\ell^-$ ▷ Hidden features
  $\boldsymbol{E}_\ell = \boldsymbol{X}_\ell^+ - \boldsymbol{X}_\ell^-$
  $\boldsymbol{E}_\ell = \boldsymbol{U}_\ell \boldsymbol{\Sigma}_\ell \boldsymbol{V}_\ell^\top$
  $\boldsymbol{V}_{\ell,k} = [\boldsymbol{v}_1, ..., \boldsymbol{v}_k]$ ▷ Selecting Top-k
  $\boldsymbol{W}_\ell^{\text{ed}} = (\boldsymbol{I} - \boldsymbol{V}_{\ell,k}\boldsymbol{V}_{\ell,k}^\top)\, \boldsymbol{W}_\ell^{\text{org}}$ ▷ Editing ℓ-th MLP
end for

Nullu. We summarize Nullu in Algorithm 1. Our method contains two key parameters: the indices of layers needed to be editing $\{\ell\}$ and the selected top-k singular vectors. Although we do not have a theoretical argument for the best values, we explore their effects experimentally and determine optimal values via a standard hyper-parameter sweep.

3.3. Decoding information in HalluSpace

We provide a case study to explore how the proposed Nullu mitigates OH. We conduct the analysis experiments on CHAIR [37] with LURE [53], which provides the ground truth and possible hallucinated responses with each visual input. The introduction of the dataset is provided in Section 4. We use LLaVA-1.5-7B [30] for our evaluation.

Mapping the subspace back to vocabulary. Following previous works [16, 43], to explore the information behind the learned vectors $\boldsymbol{v}_i \in \mathbb{R}^D$, a common approach is to decode the embeddings with $\boldsymbol{O}\boldsymbol{v} \in \mathbb{R}^{|\mathcal{V}|}$, where $\boldsymbol{O} = [\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{|\mathcal{V}|}]^\top \in \mathbb{R}^{|\mathcal{V}| \times D}$ and $\mathcal{V}$ denotes the vocabulary. We then sort $\boldsymbol{O}\boldsymbol{v}$ in ascending order, find the top-m indices, and use the corresponding words to interpret $\boldsymbol{v}$. The following analysis is performed at each layer ℓ, while we will drop ℓ to avoid the notational burden.

Using Nullu, we can extract the $\boldsymbol{V}$ at different layers and then decode it via $\boldsymbol{O}\boldsymbol{v}$ to explore the internal information behind $\boldsymbol{V}$. For better interpretation, we only select the most representative results[2], containing high-frequent words like "background" and "middle", shown in Supplementary materials. As far as we know, projecting the model weights orthogonally to $\boldsymbol{V}$ can reduce the semantic information of the decoded words during responding. However, why does such a procedure mitigate OH?

Debiasing LVLM from LLM prior. Recent studies [23] have revealed that hallucinated responses of LVLMs can be attributed to the strong LLMs' priors. Therefore, following the settings in [23], we collect the outputs of LLaVA-1.5 with normal inputs (③) and visually distorted inputs (②), as shown in Figure 3 (b). The statistical results of word frequency of ② and ③ show that LLMs indeed bias on some words. And techniques proposed to calibrate the outputs by

[1]The proof can be found in supplementary materials.

[2]The complete decoding results of $\boldsymbol{V}$ and the reason why we select the most representative results are provided in Supplementary materials.

reducing these detected LLM priors [23] (resulting in the results ④), or, enhancing the visual information [18, 33, 55] during inference to remove these priors in LLMs.

Interpreting HalluSpaces. From the results of ② in Figure 3 (b) and the results in Table 1 in Supplementary materials, we see that the decoded words from HalluSpace overlap with the high frequent words when with the distorted visual inputs (②), indicating that HalluSpace $\boldsymbol{V}$ might contain similar latent information as LLM's prior. Moreover, by visual contrastive decoding, we see that these LLM's "preferred" words are reduced (see ④), which has a similar trend to Nullu (①). Therefore, we indicate that Nullu projects the features to the null space of the HalluSpace $\boldsymbol{V}$, achieving a similar function of reducing LLM prior in LVLMs. By editing the LVLMs with Nullu, the lower CHAIR_S (the metric to evaluate hallucination, the lower, the better) indicates that OH is effectively mitigated by Nullu. Therefore, we can consider Nullu as a new way that achieves debiasing from the model parameter aspect.

3.4. How Nullu works?

The analysis is performed for each layer ℓ, and to avoid the notational burden, we will drop ℓ and focus on each layer separately. We also suppose the number of tokens as 1; the generated features can then be calculated as $\boldsymbol{f}_i = \boldsymbol{x}_i \boldsymbol{W}^{\text{org}}$, where $\boldsymbol{W}^{\text{org}} \in \mathbb{R}^{D \times D_o}$, and $\boldsymbol{x}_i \in \mathbb{R}^{1 \times D}$, $\boldsymbol{f}_i \in \mathbb{R}^{1 \times D_o}$.

Based on the heuristic in [43], an embedding vector in any transformer layer can be decomposed into interpretable components. We suppose that the generated features $\boldsymbol{f}_i$ can be separated into three different elements:

$$\boldsymbol{f}_i \Rightarrow \underbrace{\hat{\boldsymbol{f}}_i \hat{\boldsymbol{B}}}_{\text{truthful contexts}} + \underbrace{\widetilde{\boldsymbol{f}}_i \widetilde{\boldsymbol{B}}}_{\text{hallucinated biases}} + \underbrace{\boldsymbol{u}_i}_{\text{noise}}, \tag{4}$$

where we suppose $\hat{\boldsymbol{B}} \in \mathbb{R}^{\hat{k} \times D_o}$ contains the truthful directions and $\widetilde{\boldsymbol{B}} \in \mathbb{R}^{\widetilde{k} \times D_o}$ contains $\widetilde{k}$ hallucinated directions, and $\boldsymbol{u}_i$ represents typical randomness unaccounted for by the statistical model. $\hat{\boldsymbol{f}}_i \in \mathbb{R}^{1 \times \hat{k}}$, $\widetilde{\boldsymbol{f}}_i \in \mathbb{R}^{1 \times \widetilde{k}}$ are "latent factors". Therefore $\widetilde{\boldsymbol{f}}_i \widetilde{\boldsymbol{B}}$ can represent the hallucinated features for the i-th input.

However, with the $\boldsymbol{W}^{\text{ed}}$ and combining the hypothesize in Eq. (4), the edited features can be calculated as

$$\begin{aligned} \boldsymbol{f}_i^{\text{ed}} &= \boldsymbol{x}_i (\boldsymbol{I} - \boldsymbol{V}_k \boldsymbol{V}_k^\top) \boldsymbol{W}^{\text{org}} \\ &= \hat{\boldsymbol{f}}_i \hat{\boldsymbol{B}} + (\widetilde{\boldsymbol{f}}_i \widetilde{\boldsymbol{B}} - (\boldsymbol{x}_i \boldsymbol{V}_k)(\boldsymbol{V}_k^\top \boldsymbol{W}^{\text{org}})) + \boldsymbol{u}_i. \end{aligned} \tag{5}$$

As the $\boldsymbol{V}_k$ gives the best k-rank approximation of the difference between the truthful and untruthful features via SVD, our approach can remove the hallucinated features and enhance the truthful answers for accurate responses.

Relation to DPO Following previous studies in [43], we exhibit the conceptual connection between DPO [36] and Nullu by studying a simple logistic model $\pi_{\boldsymbol{W}}$ for the output given $\boldsymbol{x}$ as inputs. The conditional probability can be

$$\pi_{\boldsymbol{W}}(y|\boldsymbol{x}_i) = Z_{\boldsymbol{W}}^{-1} \exp\left(\boldsymbol{o}_y^\top \boldsymbol{W} \boldsymbol{x}_i\right) \tag{6}$$

where $\boldsymbol{o}_y$ is the output decoding vector for any $y \in \mathcal{V}$, $Z_{\boldsymbol{W}}$ is the normalization factor. Similar expression holds for $\boldsymbol{x}_i^+$ and $\boldsymbol{x}_i^-$. With some calculations, the gradient with respect to $\boldsymbol{W}$ of the DPO loss can be represented as

$$\begin{aligned} \nabla_{\boldsymbol{W}} \mathcal{L}_{\text{DPO}} &= -\frac{\beta}{N} \sum_{i=1}^{N} \left(\boldsymbol{o}_{y_i^+} (\boldsymbol{x}_i^+)^\top - \boldsymbol{o}_{y_i^-} (\boldsymbol{x}_i^-)^\top \right) \\ &= -\frac{\beta}{N} \sum_{i=1}^{N} \big(\underbrace{\boldsymbol{o}_{y_i^+} (\boldsymbol{x}_i^+ - \boldsymbol{x}_i^-)^\top}_{\text{feature difference}} + \underbrace{(\boldsymbol{o}_{y_i^+} - \boldsymbol{o}_{y_i^-})(\boldsymbol{x}_i^-)^\top}_{\text{output difference}} \big). \end{aligned} \tag{7}$$

The gradient in (7) contains a feature difference term. Therefore, the gradient update can be interpreted as an attempt to eliminate feature differences to avoid hallucinated responses. For Nullu, it tries to approximate such difference via SVD and also attempts to eliminate it by null space projection, which shows the connection between Nullu and DPO. See Supplementary Materials for more details.

4. Experiments

In this section, we conduct experiments to evaluate the effectiveness of our method in mitigating OH. Experimental analyses and a case study are also provided.

4.1. Datasets and baselines

We use the widely used CHAIR [37] and POPE [27], and randomly sampled 500 images from the validation split of MSCOCO [28] for evaluations following previous settings in [9, 18, 49]. We repeat the experiments three times for each metric with different random seeds. Moreover, MME [15] and LLaVA-Bench [30] are applied as extensive benchmarks tailored to assess the overall performance after editing. The detailed descriptions and implementations can be found in Supplementary Materials.

CHAIR. CHAIR is a tailored tool created to evaluate the occurrence of OH in an image description by determining the proportion of the mentioned objects that are absent in the ground-truth label set. For metrics, CHAIR_S measures proportion of the hallucinated sentences over all sentences, and CHAIR_I measures proportion of the hallucinated objects over all generated objects. Lower scores indicate less OH. We also report BLEU as an assessment of the text generation quality. For implementation, we use LURE [53] as the paired data for Nullu. We prompt all methods with "*Please describe this image in detail.*".

POPE. The POPE dataset presents a streamlined approach to assess object hallucination. With POPE, LVLMs

Method	LLaVA-1.5			MiniGPT-4			mPLUG-Owl2		
	CHAIR$_S$ ↓	**CHAIR$_I$** ↓	BLEU↑	**CHAIR$_S$** ↓	**CHAIR$_I$** ↓	BLEU↑	**CHAIR$_S$** ↓	**CHAIR$_I$** ↓	BLEU↑
Greedy	$20.40_{\pm 2.80}$	$7.08_{\pm 0.33}$	$15.72_{\pm 0.10}$	$32.40_{\pm 2.20}$	$12.20_{\pm 0.42}$	$14.57_{\pm 0.11}$	$22.90_{\pm 0.90}$	$8.62_{\pm 0.11}$	$15.01_{\pm 0.24}$
Beam Search [14]	$19.50_{\pm 2.30}$	$6.84_{\pm 0.79}$	$15.99_{\pm 0.14}$	$30.10_{\pm 0.30}$	$11.87_{\pm 0.37}$	$15.35_{\pm 0.24}$	$20.30_{\pm 0.70}$	$7.62_{\pm 0.19}$	$15.43_{\pm 0.05}$
DoLa [11]	$20.20_{\pm 2.80}$	$6.75_{\pm 0.54}$	$15.68_{\pm 0.10}$	$31.90_{\pm 3.30}$	$12.15_{\pm 0.89}$	$14.54_{\pm 0.12}$	$22.40_{\pm 1.80}$	$8.36_{\pm 0.04}$	$15.13_{\pm 0.21}$
OPERA [18]	$17.50_{\pm 0.50}$	$6.07_{\pm 0.32}$	$16.02_{\pm 0.02}$	$29.70_{\pm 0.30}$	$11.96_{\pm 0.29}$	$14.82_{\pm 0.05}$	$20.07_{\pm 2.07}$	$7.18_{\pm 0.39}$	$15.41_{\pm 0.12}$
VCD [23]	$20.30_{\pm 1.10}$	$7.28_{\pm 0.10}$	$14.53_{\pm 0.01}$	$29.00_{\pm 2.80}$	$12.64_{\pm 1.19}$	$14.42_{\pm 0.01}$	$22.80_{\pm 0.80}$	$8.68_{\pm 0.17}$	$15.14_{\pm 0.13}$
Woodpecker [49]	$23.85_{\pm 4.62}$	$7.50_{\pm 0.01}$	$17.05_{\pm 0.00}$	$28.87_{\pm 2.20}$	$10.20_{\pm 0.85}$	$15.30_{\pm 0.01}$	$26.33_{\pm 1.98}$	$8.43_{\pm 0.80}$	$16.43_{\pm 0.00}$
LURE [53]	$19.48_{\pm 2.35}$	$6.50_{\pm 0.38}$	$15.97_{\pm 0.01}$	$27.88_{\pm 2.25}$	$10.20_{\pm 0.85}$	$15.03_{\pm 0.11}$	$21.27_{\pm 0.06}$	$7.67_{\pm 0.16}$	$15.65_{\pm 0.15}$
HALC [9]	$16.90_{\pm 2.10}$	$5.72_{\pm 0.55}$	$16.02_{\pm 0.04}$	$25.20_{\pm 2.00}$	$9.42_{\pm 0.41}$	$14.91_{\pm 0.13}$	$18.80_{\pm 1.20}$	$7.00_{\pm 0.01}$	$15.33_{\pm 0.24}$
Nullu	**15.20**$_{\pm 0.60}$	**5.30**$_{\pm 0.03}$	$15.69_{\pm 0.04}$	**21.40**$_{\pm 1.00}$	**8.99**$_{\pm 0.36}$	$14.81_{\pm 0.06}$	**15.60**$_{\pm 1.20}$	**5.77**$_{\pm 0.01}$	$15.45_{\pm 0.01}$

Table 1. CHAIR evaluation results on MSCOCO dataset of LVLMs (LLaVA-1.5, MiniGPT-4 and mPLUG-Owl2) with different methods for mitigating OH. Lower CHAIR$_S$ and CHAIR$_I$ indicate less OH. Higher BLEU generally represent higher captioning quality. We use 64 as the max token number in this experiment. Bold indicates the best result of all methods.

are queried to answer whether a specific object exists in the image. It encompasses three sampling settings: *random, popular, and adversarial*, each distinct in constructing negative samples. Besides the basic evaluation method, we further use offline POPE (OPOPE) [9], which keeps the object sampling and yes/no query strategy from POPE but replaces the live interactions with offline checks. Specifically, instead of querying the model with "*Is there a {} in the image?*", where "{}" is the queried object, we first ask the examined LVLM to give its detailed descriptions of the image and then check if the sampled positive/negative objects exist in the captions when computing the OPOPE scores.

MME and LLaVA-Bench. The Multi-modal Large Language Model Evaluation (MME) benchmark comprises ten perception-related and four cognition-related tasks. We use all tasks to test the edited models comprehensively. Moreover, LLaVA-Bench is a collection of 24 images, where each image is paired with a detailed, manually crafted description and carefully selected questions. Both of these datasets are used to assess the capability of LVLMs to tackle more challenging tasks.

LVLM Baselines. We evaluate the effectiveness of our Nullu on three popular LVLMs, including LLaVA-1.5 [31] with Vicuna [10], MiniGPT-4 [54] with Llama2 [41] and mPLUG-Owl2 [48]. See supplementary materials for more implementation details.

4.2. Results on CHAIR

We provide the experimental results in Table 1, from which we see that Nullu consistently outperforms the evaluated methods. For the metric, CHAIR$_S$ (C$_S$) can be more critical, since a caption containing many correct objects with one Hallucinated Object is still an error with C$_S$. A significant improvement in C$_S$ indicates that Nullu addresses more last hallucinated object. Moreover, although Nullu achieves moderate gain compared to HALC with C$_I$, it's over 10× faster than HALC.

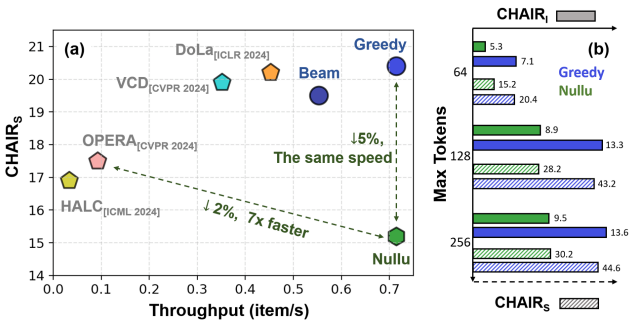


Figure 4. (a) The throughput (tested on NVIDIA RTX 4090) v.s. CHAIR$_S$. (b) Performance with different max token numbers.

Moreover, we compared our method to a series of baseline models on LLaVA-1.5 with different numbers of max tokens, and the results are shown in Figure 4 (b). Results show that more output tokens lead to more hallucinations in the responses. While the proposed Nullu still outperforms the baseline model by maintaining a low ratio of hallucination with longer responses and more generated objects.

A unique advantage of Nullu is that it can reparameterize the edited weights to the original model and, therefore, introduces no extra inference costs. However, most post-hoc OH mitigation methods need to modify the final inference procedure by either debiasing the LLM biases with double-inference [23] or even using multiple adaptive inference procedures [9] and therefore introduce plenty of extra costs during inference. We test the inference speed of DoLa [11], VCD [23], and HALC [9] along with the original LLaVA-1.5 with greedy and beam search decoding strategies. The results are shown in Figure 4 (a). Compared to LLaVA-1.5 with greedy decoding strategy, Nullu can decrease the CHAIR$_S$ by over 5% with the same throughput. Although HALC performs well in mitigating OH, it also introduces many extra inference procedures, which slow the inference speed. Compared to OPERA, Nullu can achieve 7× faster speed while outperforming OPERA in CHAIR$_S$.

Method	LLaVA-1.5			MiniGPT-4			mPLUG-Owl2		
	Accuracy↑	**Precision↑**	**F score↑**	**Accuracy↑**	**Precision↑**	**F score↑**	**Accuracy↑**	**Precision↑**	**F score↑**
Greedy	$79.14_{\pm 0.89}$	$91.98_{\pm 0.82}$	$90.45_{\pm 0.86}$	$71.22_{\pm 1.27}$	$93.72_{\pm 1.02}$	$90.04_{\pm 1.23}$	$76.46_{\pm 0.92}$	$88.85_{\pm 1.15}$	$87.29_{\pm 1.15}$
Beam Search [14]	$79.41_{\pm 0.69}$	$92.52_{\pm 0.55}$	$90.96_{\pm 0.59}$	$71.65_{\pm 1.15}$	$94.70_{\pm 0.60}$	$90.97_{\pm 0.85}$	$76.76_{\pm 1.02}$	$90.28_{\pm 0.80}$	$88.56_{\pm 0.87}$
DoLa [11]	$78.98_{\pm 0.56}$	$91.66_{\pm 0.81}$	$90.15_{\pm 0.79}$	$71.28_{\pm 1.15}$	$93.92_{\pm 0.83}$	$90.22_{\pm 1.04}$	$76.07_{\pm 1.09}$	$88.54_{\pm 1.25}$	$86.95_{\pm 1.27}$
OPERA [18]	$79.29_{\pm 0.32}$	$92.25_{\pm 0.07}$	$90.71_{\pm 0.11}$	$70.48_{\pm 1.63}$	$94.41_{\pm 1.11}$	$90.66_{\pm 1.42}$	$75.49_{\pm 1.29}$	$91.23_{\pm 1.06}$	$89.11_{\pm 1.17}$
VCD [23]	$78.01_{\pm 0.75}$	$91.33_{\pm 0.88}$	$89.69_{\pm 0.89}$	$70.83_{\pm 1.83}$	$92.31_{\pm 0.88}$	$88.76_{\pm 1.29}$	$75.49_{\pm 1.27}$	$88.75_{\pm 1.56}$	$87.02_{\pm 1.57}$
HALC [9]	$77.87_{\pm 0.22}$	$93.17_{\pm 0.39}$	$91.25_{\pm 0.38}$	$71.17_{\pm 0.89}$	$94.88_{\pm 0.15}$	$90.95_{\pm 0.42}$	$74.93_{\pm 1.09}$	$90.20_{\pm 0.90}$	$88.12_{\pm 0.99}$
Nullu	$\mathbf{79.52}_{\pm 0.04}$	$\mathbf{93.46}_{\pm 0.03}$	$\mathbf{91.79}_{\pm 0.04}$	$\mathbf{71.92}_{\pm 0.39}$	$\mathbf{95.96}_{\pm 0.65}$	$\mathbf{92.07}_{\pm 0.65}$	$\mathbf{77.09}_{\pm 1.37}$	$\mathbf{92.83}_{\pm 0.29}$	$\mathbf{90.80}_{\pm 0.52}$

Table 2. The OPOPE evaluation results on MSCOCO dataset of LVLMs with different methods for mitigating OH. Higher accuracy, precision, and F score indicate better performance. Bold indicates the best result of all methods.

4.3. Results on POPE

Experimental results on POPE under the random, popular, and adversarial settings are summarized in the Supplementary Materials. Here, we mainly provide the results using OPOPE metrics, shown in Table 2. All the numbers are averaged results of the three sampling methods. Generally, OPOPE is more challenging than POPE since VLMs first need to describe a whole image and then find a specific object from the response, where the object can often be missing. The results show that Nullu also performs better than all evaluated methods regarding accuracy, precision, and F score metrics. The experimental results demonstrate the effectiveness of Nullu in OH mitigation and show its broad applicability on different open-source LVLMs.

4.4. Results on MME

We present the results of LLaVA-1.5-7B as a representative to evaluate the general ability of the edited model. From Figure 5, we see that implementing Nullu leads to a consistent enhancement in perception- and recognition-based tasks. Moreover, it is interesting that using Nullu can enhance the LVLM in text-related tasks, such as OCR and Code Sense Reasoning. This may be attributed to the fact that Nullu achieves similar optimization attempts with DPO [36], potentially improving LVLMs' general capacities. See more discussions and numerical results in Supplementary materials.

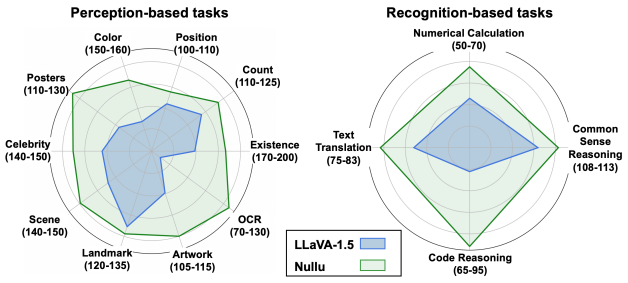


Figure 5. MME full set evaluation results on LLaVA-1.5.

4.5. Ablation Studies and Further Analysis

Effects of editing layers $\{\ell\}$ and rank k. Nullu contains two key parameters: the indices of layers needed to be edited $\{\ell\}$ and the selected top-k singular vectors. We test different values of these hyper-parameters with LLaVA-1.5-7B on CHAIR (shown in Table 3). The editing layers can affect overall performance; therefore, we use $\{\ell\} = \{16, .., 32\}$ in all experiments. Moreover, we further found that the performance of Nullu with a certain k for building HalluSpace achieves the best performance, indicating that the HalluSpace can be with a low-rank structure.

$\{\ell\}$	C_S	C_I	BLEU	k	C_S	C_I	BLEU
16-32	**15.2**	**5.30**	**15.7**	2	20.2	7.22	15.3
24-32	16.2	5.58	15.1	4	**15.2**	**5.30**	**15.7**
28-32	18.2	6.27	15.1	8	17.8	5.92	15.2
31-32	16.4	5.59	15.1	16	17.2	6.27	14.8

Table 3. Ablation study of the hyper-parameters in Nullu.

Does HalluSpace represent the hallucination biases? Ideally, if HalluSpace effectively represents these biases, the difference vectors from test samples should have big projected components when mapped onto HalluSpace. To evaluate this, we select 100 test samples from CHAIR where Nullu successfully mitigates OH issues. We compute difference vectors $\boldsymbol{e}_i$ for each sample between the raw and edited LLaVA features. Moreover, we generate 100 random vectors $\boldsymbol{r}_i$ as a comparison baseline. All these vectors are normalized to avoid the effects of norms. Moreover, we use σ_i to represent the projected components. Figure 7 (a) shows the distribution of vectors on a normalized sphere.

Given $\boldsymbol{V}_4$ (rank-4), each projected component σ_i resides within $\mathbb{R}^4$. We then calculated σ_i for all selected and random samples, averaging $||\sigma||$ across samples. Figure 7 (b) presents these results, showing that the average $||\sigma||$ of difference vectors across layers is significantly larger ($10\times$) than that of random vectors. Since the selected test samples were successfully edited to avoid OH, this evidence indicates that HalluSpace captures directions in the feature space associated with hallucinations in LVLMs.

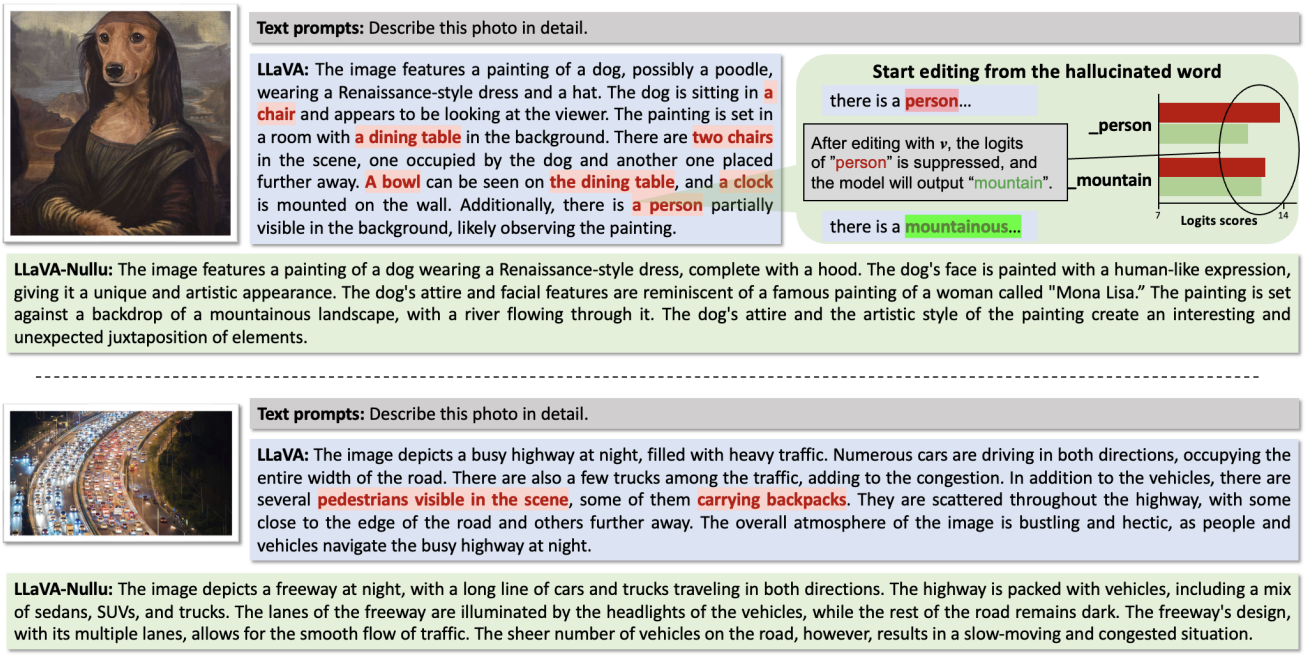


Figure 6. Illustration of hallucination correction by our proposed Nullu with two samples from LLaVA-Bench using LLaVA-1.5-7B. Hallucinated objects from the original model are highlighted in red.

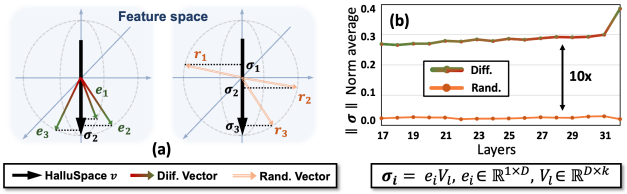


Figure 7. (a) The illustration of experiments. (b) The mean of $||\sigma||$ of 100 test and random vectors at different layers.

Model	Method	Accuracy↑	Detailedness↑
LLaVA-1.5	Original	5.89	5.02
	Nullu	**6.53**	**5.59**
MiniGPT-4	Original	4.11	4.07
	Nullu	**5.63**	**4.87**
mPLUG-Owl2	Original	5.96	4.52
	Nullu	**6.26**	**4.72**

Table 4. Results of GPT-4V-aided evaluation on LLaVA-Bench following the setting in [23]. Both metrics are on a scale of 10.

GPT-4V Aided Evaluation on LLaVA-Bench. Following [23, 49], we leverage LLaVA-Bench [30] to qualitatively evaluate the performance using GPT-4V Aided Evaluation. The prompt used for evaluation and an evaluation case are provided in Supplementary Materials. Results in Table 4 show consistent improvements using Nullu for each model, demonstrating our method's effectiveness in both OH mitigation and general model ability improvement.

Case Study on LLaVA-Bench Figure 6 demonstrates two case studies on LLaVA-1.5-7B given identical prompts and images. The results show that the original LLaVA wrongly describes the object "*chair*", then it hallucinates the frequently co-occurring objects such as "*dining table*" and "*bowl*". A similar issue also happens in the second example, where the model wrongly generates "*pedestrians*" due to the possible descriptions of the "*traffic*". In contrast, implementing Nullu can mitigate these hallucination issues effectively. We further test Nullu in an online setting, where we edited the features with the learned $\boldsymbol{V}$ before a hallucinated word, and we found that null space projection effectively fixed the hallucinated word "person" to a correct word "mountain". More results can be found in our Supplementary Materials.

5. Conclusion and Future Work

In this paper, we proposed a novel method, Nullu, to address OH in modern LVLMs. Nullu extracts HalluSpaces, which are low-rank subspaces of the differences between truthful and hallucinated features, and further edits the LVLMs weights to mitigate OH. Empirical results show that using the edited model weights reduces the priors in the LLMs, which has been proven as an essential factor for OH. Experiments demonstrate that Nullu can significantly mitigate OH with no extra inference cost, making it much more efficient than many existing methods and maintaining strong performance in general LVLM benchmarks.

Considering the connection between Nullu and DPO, Nullu can potentially be an alternative but effective way for safety alignment, providing a more efficient and straightforward way to improve LVLMs trustworthiness.

Acknowledgments

This work is supported in part by the National Key Research and Development Program of China (2023YFE0209800) and the National Natural Science Foundation of China (62206215, U24B20185, 62161160337 and 62376210).

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1

[2] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 2

[3] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 2

[4] Zechen Bai, Pichao Wang, Tianjun Xiao, Tong He, Zongbo Han, Zheng Zhang, and Mike Zheng Shou. Hallucination of multimodal large language models: A survey. *arXiv preprint arXiv:2404.18930*, 2024. 1, 2

[5] Ali Furkan Biten, Lluís Gómez, and Dimosthenis Karatzas. Let there be a clock on the beach: Reducing object hallucination in image captioning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1381–1390, 2022. 2

[6] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023. 2

[7] Yangyi Chen, Karan Sikka, Michael Cogswell, Heng Ji, and Ajay Divakaran. Dress: Instructing large vision-language models to align and interact with humans via natural language feedback. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 14239–14250, 2024. 2

[8] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 24185–24198, 2024. 2

[9] Zhaorun Chen, Zhuokai Zhao, Hongyin Luo, Huaxiu Yao, Bo Li, and Jiawei Zhou. Halc: Object hallucination reduction via adaptive focal-contrast decoding. In *Int. Conf. Machine Learn.*, 2024. 1, 2, 5, 6, 7

[10] 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, 2023. 2, 6

[11] Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. In *Int. Conf. Learn. Represent.*, 2023. 6, 7

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2

[13] Alessandro Favero, Luca Zancato, Matthew Trager, Siddharth Choudhary, Pramuditha Perera, Alessandro Achille, Ashwin Swaminathan, and Stefano Soatto. Multi-modal hallucination control by visual information grounding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 14303–14312, 2024. 2

[14] Markus Freitag and Yaser Al-Onaizan. Beam search strategies for neural machine translation. *arXiv preprint arXiv:1702.01806*, 2017. 6, 7

[15] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 5

[16] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Adv. Neural Inform. Process. Syst.*, pages 5484–5495, 2021. 4

[17] Anisha Gunjal, Jihan Yin, and Erhan Bas. Detecting and preventing hallucinations in large vision language models. In *AAAI*, pages 18135–18143, 2024. 2

[18] Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. Opera: Alleviating hallucination in multimodal large language models via over-trust penalty and retrospection-allocation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13418–13427, 2024. 2, 5, 6, 7

[19] Wen Huang, Hongbin Liu, Minxin Guo, and Neil Zhenqiang Gong. Visual hallucinations of multi-modal large language models. *arXiv preprint arXiv:2402.14683*, 2024. 2

[20] Chaoya Jiang, Haiyang Xu, Mengfan Dong, Jiaxing Chen, Wei Ye, Ming Yan, Qinghao Ye, Ji Zhang, Fei Huang, and Shikun Zhang. Hallucination augmented contrastive learning for multimodal large language model. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 27036–27046, 2024. 1, 2

[21] Liqiang Jing and Xinya Du. Fgaif: Aligning large vision-language models with fine-grained ai feedback. *arXiv preprint arXiv:2404.05046*, 2024. 2

[22] Jae Myung Kim, A Koepke, Cordelia Schmid, and Zeynep Akata. Exposing and mitigating spurious correlations for cross-modal retrieval. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2584–2594, 2023. 1, 2

[23] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13872–13882, 2024. 1, 2, 4, 5, 6, 7, 8

[24] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Int. Conf. Machine Learn.*, pages 12888–12900, 2022. 2

[25] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Int. Conf. Machine Learn.*, pages 19730–19742, 2023. 2

[26] Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[27] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305, 2023. 1, 5

[28] 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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 5

[29] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Mitigating hallucination in large multi-modal models via robust instruction tuning. *arXiv preprint arXiv:2306.14565*, 2023. 1, 2

[30] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 2, 4, 5, 8

[31] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Adv. Neural Inform. Process. Syst.*, 36, 2024. 1, 2, 6

[32] Sheng Liu, Haotian Ye, and James Zou. Reducing hallucinations in large vision-language models via latent space steering. In *Int. Conf. Learn. Represent.*, 2024. 3

[33] Shi Liu, Kecheng Zheng, and Wei Chen. Paying more attention to image: A training-free method for alleviating hallucination in lvlms. In *Eur. Conf. Comput. Vis.*, 2024. 2, 5

[34] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 2

[35] AI Meta. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI Blog (accessed 2024–04–20).*, 2024. 1

[36] 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. *Adv. Neural Inform. Process. Syst.*, 36, 2024. 2, 5, 7

[37] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. *arXiv preprint arXiv:1809.02156*, 2018. 2, 4, 5

[38] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, Yiming Yang, et al. Aligning large multimodal models with factually augmented rlhf. *arXiv preprint arXiv:2309.14525*, 2023. 2

[39] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1

[40] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1

[41] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 2, 6

[42] Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. *arXiv e-prints*, pages arXiv–2308, 2023. 3

[43] Rheeya Uppaal, Apratim Dey, Yiting He, Yiqiao Zhong, and Junjie Hu. Model editing as a robust and denoised variant of dpo: A case study on toxicity. In *Int. Conf. Learn. Represent.*, 2024. 3, 4, 5

[44] Huanqian Wang, Yang Yue, Rui Lu, Jingxin Shi, Andrew Zhao, Shenzhi Wang, Shiji Song, and Gao Huang. Model surgery: Modulating llm's behavior via simple parameter editing. *arXiv preprint arXiv:2407.08770*, 2024. 2

[45] Junfei Wu, Qiang Liu, Ding Wang, Jinghao Zhang, Shu Wu, Liang Wang, and Tieniu Tan. Logical closed loop: Uncovering object hallucinations in large vision-language models. *arXiv preprint arXiv:2402.11622*, 2024. 2

[46] Mingrui Wu, Jiayi Ji, Oucheng Huang, Jiale Li, Yuhang Wu, Xiaoshuai Sun, and Rongrong Ji. Evaluating and analyzing relationship hallucinations in large vision-language models. In *Int. Conf. Machine Learn.*, pages 53553–53570, 2024. 2

[47] 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, 2024. 2

[48] Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13040–13051, 2024. 1, 2, 6

[49] Shukang Yin, Chaoyou Fu, Sirui Zhao, Tong Xu, Hao Wang, Dianbo Sui, Yunhang Shen, Ke Li, Xing Sun, and Enhong Chen. Woodpecker: Hallucination correction for multimodal large language models. *arXiv preprint arXiv:2310.16045*, 2023. 2, 5, 6, 8

[50] Yi-Fan Zhang, Weichen Yu, Qingsong Wen, Xue Wang, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. Debiasing large visual language models. In *Eur. Conf. Comput. Vis.*, 2024. 1, 2, 4

[51] Andrew Zhao, Quentin Xu, Matthieu Lin, Shenzhi Wang, Yong-jin Liu, Zilong Zheng, and Gao Huang. Diver-ct: Diversity-enhanced red teaming with relaxing constraints. *arXiv preprint arXiv:2405.19026*, 2024. 2

[52] Wangbo Zhao, Yizeng Han, Jiasheng Tang, Zhikai Li, Yibing Song, Kai Wang, Zhangyang Wang, and Yang You. A

stitch in time saves nine: Small vlm is a precise guidance for accelerating large vlms. *arXiv preprint arXiv:2412.03324*, 2024. 2

[53] Yiyang Zhou, Chenhang Cui, Jaehong Yoon, Linjun Zhang, Zhun Deng, Chelsea Finn, Mohit Bansal, and Huaxiu Yao. Analyzing and mitigating object hallucination in large vision-language models. *arXiv preprint arXiv:2310.00754*, 2023. 1, 2, 3, 4, 5, 6

[54] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2, 6

[55] Lanyun Zhu, Deyi Ji, Tianrun Chen, Peng Xu, Jieping Ye, and Jun Liu. Ibd: Alleviating hallucinations in large vision-language models via image-biased decoding. In *Int. Conf. Machine Learn.*, 2024. 2, 5