# Fuzzy Contrastive Decoding to Alleviate Object Hallucination in Large Vision-Language Models

Jieun Kim Jinmyeong Kim Yoonji Kim Sung-Bae Cho
Yonsei University
{lilly9928, jmkim_, yoonjikim, sbcho}@yonsei.ac.kr

## Abstract

*Large vision-language models (LVLMs) often exhibit object hallucination, a phenomenon where models generate descriptions of non-existent objects within images. Prior methods have sought to mitigate this issue by adjusting model logits to reduce linguistic bias, but they often lack precise control over visual uncertainty, sometimes exacerbating hallucinations instead of mitigating them. To address this limitation, we propose a novel decoding strategy called fuzzy contrastive decoding (FuzzyCD) that uses Takagi-Sugeno fuzzy inference to refine hallucination control. FuzzyCD adaptively assigns weights to high-hallucination logits while mitigating unnecessary linguistic bias. Specifically, it transforms the log-probabilities of top-1 tokens from both standard and hallucination logits into a confidence linguistic fuzzy set. Through Takagi-Sugeno fuzzy inference, it dynamically adjusts hallucination logits to prevent the model from over-relying on spurious linguistic patterns. Experimental results on object hallucination datasets demonstrate that hallucination is mitigated by 11%p compared to conventional LVLMs. In-depth analyses highlight the effectiveness of FuzzyCD in enhancing the reliability of vision-language models. The source code is available at: https://github.com/lilly9928/FuzzyCD*

## 1. Introduction

Recent advances in large vision-language models (LVLMs) have led to substantial progress in multimodal understanding and interaction by integrating large-scale language models with visual input [8, 19, 21, 25, 32]. However, these models are prone to object hallucination, a phenomenon in which they either describe non-existent objects or fail to recognize objects that are actually present in the given image [11, 13, 17, 20, 34]. For example, for the question asked: "What is the vicuna standing on the sand looking at?" in Figure 1, LVLM fails to recognize the absence of a vicuna in the image and instead generates a response based on the implicit assumptions within the question. This phenomenon reflects an excessive reliance on textual cues over actual visual information, thereby constraining the efficacy of the model in real-world applications where precision is essential. The primary causes of object hallucination can be attributed to the statistical biases inherent in the training data and the reliance on language priors [1, 7, 14].

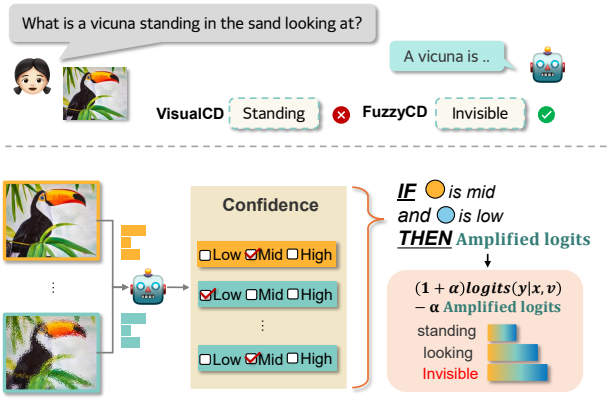


Figure 1. An illustration of the fuzzy contrastive decoding. Orange background image represents the original, while the blue background images represent the filtered.

To mitigate it, various methods have been proposed, including fine-tuning models with specially curated datasets designed to reduce hallucination and reinforcement learning with human feedback (RLHF) [17, 22, 30]. However, they require substantial human resources and computational costs and often exhibit limited generalization ability across various scenarios, thus constraining their scalability and practical applicability. Recently, decoding techniques have emerged to address object hallucination [9, 15, 20, 38], but they mainly focus on amplifying or suppressing hallucinations rather than directly detecting them. A key limitation is the lack of direct control over hallucination amplification, which prevents the selective enhancement of accurate tokens. Consequently, instead of mitigating hallucinations, they inadvertently introduce new hallucinations, further ex-

acerbating the problem [2, 11, 17].

To cope with the amplification of uncontrollable hallucination, this paper proposes a method of fuzzy constrative decoding (FuzzyCD), which consists of two main stages: fuzzy rule inference and contrastive decoding. In the fuzzy rule inference, log-probabilities are utilized as confidence scores, and fuzzification is applied to represent the presence of hallucination using membership degrees categorized as 'high, medium, and low.' In logical reasoning and inference, we apply the Takagi-Sugeno (T-S) fuzzy model [31] to calculate the weight of each rule. Subsequently, the amplified logits for the rules are computed, and the final combined logit is derived by calculating a weighted average. The final amplified logit computed is then removed from the original logit, systematically minimizing hallucination elements. The proposed method is validated through experiments using benchmark datasets specifically focused on object hallucination, demonstrating that it can significantly reduce the frequency of hallucination occurrences. The key contributions of this paper are summarized as follows.

- **Hallucination detection with probability of LVLMs**: An extensive analysis of hallucination detection in LVLMs by examining the mean, max, min of logits, and log-probability and probability for hallucination detection indicates that utilizing probability achieves the highest detection performance.
- **Fuzzy contrastive decoding**: The fuzzy contrastive decoding method for mitigating object hallucination guides to more accurately reflect actual visual information and reduce its reliance on textual biases by fuzzifying the logits into confidence fuzzy set of 'high,' 'medium,' and 'low' and dynamically adjusting logits using Takagi-Sugeno fuzzy model.
- **Scalable, training-free method**: Unlike existing methods that require fine-tuning with curated datasets or reinforcement learning, FuzzyCD is immediately deployable without additional training. This method can be applied to various LVLM architectures and datasets, making it a highly practical and scalable solution.

## 2. Related Work

In this section, we introduce the methods devised to address object hallucination in LVLMs and present techniques for hallucination detection in large-language models (LLMs) in the field of natural language processing.

### 2.1. Mitigating Hallucination for LVLMs

Various methods have been proposed to address hallucination in LVLMs, and the most dominant are fine-tuning with hallucination mitigation datasets and the use of reinforcement learning with human feedback [18, 27, 29, 33]. However, they entail substantial computational costs and require significant human labor [12, 23, 35]. To overcome these limitations, recent research has introduced training-free decoding methods. Decoding-based methods aim to mitigate hallucination by generating logits amplifying hallucination and then removing them from the original logits [12]. Techniques for this process include various image transformations or the use of uniform images (e.g., white or black).

Although they have been effective in reducing statistical biases in training data, they are constrained by an inability to precisely identify the logits responsible for hallucination, often resulting in outcomes that overly depend on amplified hallucination effects. In addition, when image transformations are used to amplify hallucination, unintended hallucination may occur in irrelevant areas, or logits that do not pertain to the actual image content may be produced [39]. To deal with these issues, in this paper, we explore the potential for detecting hallucination within the model's internal states and propose a fuzzy logic-based method to adjust logits based on these findings. This method enables a more targeted identification [3, 6] and mitigation of specific logits that contribute to hallucination.

### 2.2. Hallucination Detection in LLMs

To address the mitigation of hallucination in LLMs, extensive research has focused on hallucination detection [5, 16, 23, 26, 37, 39, 40]. In the NLP domain, methods for hallucination mitigation include classifiers trained on internal states of the model [24] and fine-tuning methods using datasets that assess the factuality of the generated outputs. In addition, using the mean, max, and min of logits as confidence score has been widely explored to detect hallucination [35]. Recent studies also suggest that the first token generated by an LLM plays a critical role in determining the likelihood of hallucination [28]. Building on these findings in hallucination detection for LLM, this paper investigates the feasibility of detecting hallucinations within LVLMs through a detailed analysis of their internal states. Furthermore, it introduces a novel method that uses this detection capability to enhance the reliability and factual consistency of LVLM output.

## 3. Method

### 3.1. Problem Definition

Given a linguistic query $x$ and an image $v$, a LVLM generates the next token $y_t$ by autoregressively sampling from a predefined probability distribution:

$$y_t \sim P(y_t \mid x, v, y_{<t}) \tag{1}$$

where $y_t$ denotes the current token and $y_{<t}$ represents the sequence of previously generated tokens. However, LVLMs are subject to object hallucination, where the generated tokens refer to objects that are non-existent or misrepresented in the given image.

To mitigate such hallucination, contrastive decoding is employed, which refines the token generation process by adjusting the output distribution as follows:

$$y_t \sim (1+\alpha) \cdot y_t - \alpha \cdot y'_t \tag{2}$$

where $y'_t$ is derived from an alternative probability distribution:

$$y'_t \sim P(y'_t \mid x', y_{<t}) \tag{3}$$

Here, $x'$ represents an input that is designed to induce hallucination. However, a critical issue arises when $x'$ does not effectively induce hallucinations, leading not to their mitigation but rather to the generation of new hallucinations. To address this limitation, we aim to detect hallucination and appropriately suppress the amplified logits of hallucinated tokens using fuzzy logic.

### 3.2. Model Confidence

Recent work has exploited the internal states of LLMs, specifically final hidden representations and exact answer tokens to improve hallucination detection and AUC performance across multiple datasets [23, 28]. These findings suggest that LLM logits can serve as effective confidence scores for hallucination detection. Table 1 and Figure 2 show an example of experimental results on hallucination detection. To identify which logit-based metrics are effective in detecting hallucinations, we use a simple classifier trained on the VizWiz training dataset to determine the presence of hallucinations. The classifier utilizes the mean, min, max, probability, and log-probability of logits as input features to predict hallucination occurrences. Table 1 shows that probability and log-probability achieve the highest AUC scores in all settings, making them the most effective criteria for hallucination detection. Logits-max performs relatively well, especially in the POPE dataset, while Logits-mean and Logits-min show worse performance. These results indicate that probability-based metrics provide stronger confidence signals for the detection of hallucinations. Figure 2 shows the density distributions of logit-based confidence scores for correct and incorrect answers. In the case of probability, the correct answers are concentrated around a specific value. This suggests that hallucination detection is possible using logit information. However, there are overlapping areas, indicating that classification based solely on logit values is challenging.

### 3.3. Fuzzy Contrastive Decoding

Based on the results of the previous section, we confirm that hallucinations can be detected by probability. Using this, we propose a fuzzy contrastive decoding (FuzzyCD) method that effectively detects the occurrence of hallucinations and amplifies hallucination logits. FuzzyCD consists

| | VizWiz | POPE | | |
|---|---|---|---|---|
| | val | random | popular | adversarial |
| Logits-mean | 0.58 | 0.24 | 0.26 | 0.31 |
| Logits-min | 0.56 | 0.25 | 0.27 | 0.31 |
| Logits-max | 0.70 | 0.69 | 0.72 | 0.72 |
| Probability | **0.75** | **0.79** | **0.78** | **0.76** |
| Log-probability | **0.75** | **0.79** | **0.78** | **0.76** |

Table 1. Hallucination detection criteria for LLaVA-1.5 on two datasets using the AUC metric. The best-performing criterion appears in bold.

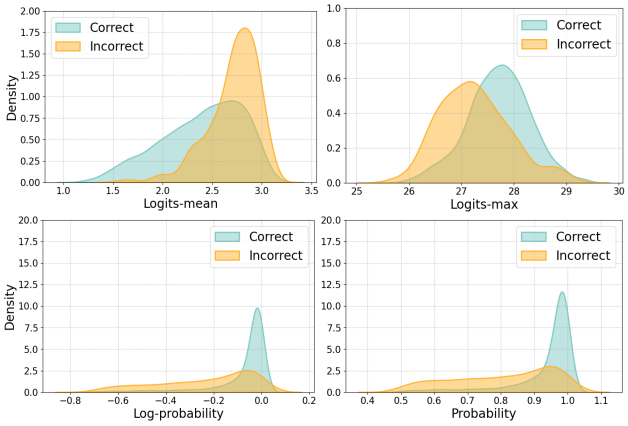


Figure 2. Density distributions of logit-based metrics for hallucination analysis. The density distributions of 'Correct' (non-hallucinated) and 'Incorrect' (hallucinated) predictions using different logit-based metrics.

of two stages: fuzzy rule inference and contrastive decoding. In the fuzzy rule inference stage, hallucinations are detected, and Takagi-Sugeno (T-S) fuzzy rules are used to generate logits that amplify hallucinations. Then, in the contrastive decoding stage, the amplified logits are used to perform a weighted average with each rule weight, ultimately generating hallucination-amplified logits, which are then removed from the original logits.

#### 3.3.1. Fuzzification

In Figure 3, phase 1 is the fuzzy rule inference process. According to the observations from the previous section, the log-probability is adopted as the confidence score of the model for fuzzification. Through this process, the membership degree of the confidence score is determined in the linguistic confidence categories of 'low,' 'medium,' and 'high,' thereby enabling the representation of uncertainty. To perform fuzzification, 10% of the dataset is initially sampled to calculate the mean ($x_{\text{mean}}$) and standard deviation ($\sigma$). Subsequently, these values are used to define membership functions for each confidence level, allowing the model to quantify the likelihood that the predicted confidence score

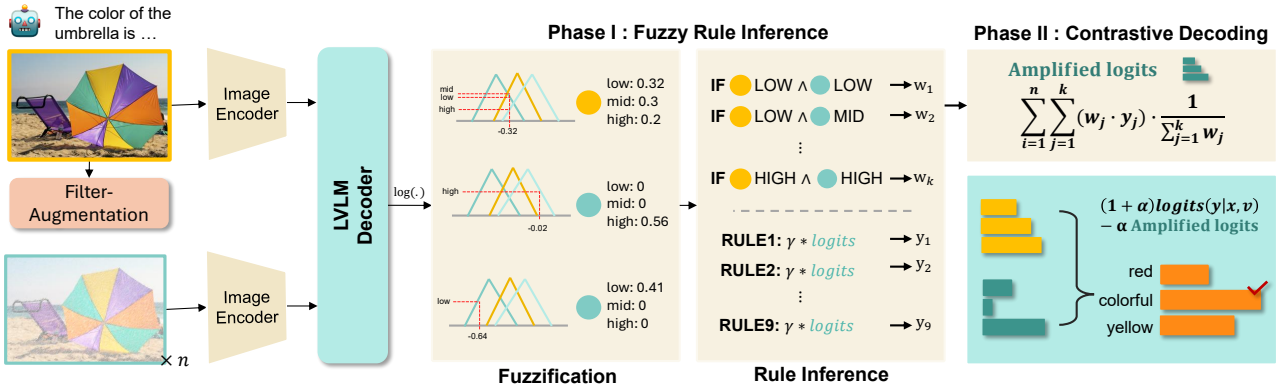


Figure 3. Overview of the proposed FuzzyCD. Given an image, hallucination-amplified data is generated through image filtering. Logits are then extracted by LVLM decoder, and the log-probability serves as a confidence score, categorized into low, medium, and high. Final logits are obtained through fuzzy logic calculation.

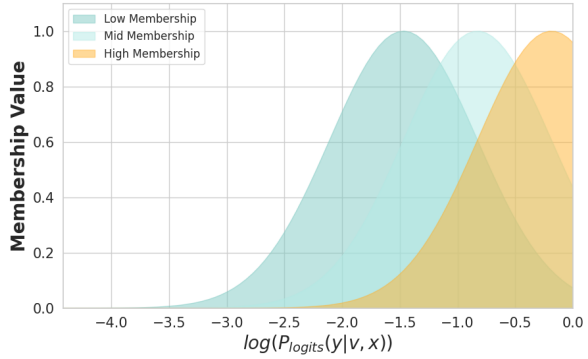


Figure 4. Visualization of membership functions.

(logits-mean) $x$ falls within the 'low,' 'medium,' or 'high' confidence range. Fuzzification is formulated based on a Gaussian distribution as follows:

$$\mu_k(x) = \exp\left(-\frac{(x - (x_{\text{mean}} + k \cdot \sigma))^2}{2\sigma^2}\right), \quad k \in \{-1, 0, 1\} \tag{4}$$

where $\sigma$ denotes the standard deviation, and each function $\mu_k(\cdot)$ represents the membership degree of the confidence score $x$ within a specific confidence category. The parameter $k$ designates the confidence level, where $k \in \{-1, 0, 1\}$ corresponds to "low," "medium," and "high" confidence levels, respectively. Figure 4 shows the generated membership functions.

### 3.3.2. Fuzzy Rule Inference

For rule inference, we propose a method to adjust logits based on the membership degrees of confidence fuzzy sets using the T-S fuzzy model, thereby generating amplified logits. The T-S fuzzy model classifies confidence levels into three categories: low, medium, and high. This model effectively handles uncertainties arising within these confidence ranges and enables the generation of hallucination-amplified logits based on fuzzy rules. We define nine fuzzy rules using three fuzzy sets that cover all possible combinations of the sets. Each rule computes its weight based on the membership degrees of the input values. For example, if $x_1$ and $x_2$ belong to the "low" category with membership degrees of 0.04 and 0.05, respectively, the weight of Rule 1 is calculated as 0.20. The weight of each rule, denoted as $w_{\text{rule}_i}$, is obtained by multiplying the membership functions $\mu_{\text{k}}(x_1)$ and $\mu_{\text{k}}(x_2)$, and is formally defined as follows:

$$w_{\text{rule}_i} = \mu_{\text{k}}(x_1) \cdot \mu_{\text{k}'}(x_2), \quad \text{for } i = 1, 2, \ldots, 9 \tag{5}$$

where k and k′ represent the fuzzy sets corresponding to the low, medium, or high confidence fuzzy set. The conclusion of each fuzzy rule is formulated as a logit amplification equation. The amplification coefficients corresponding to each filter image's confidence fuzzy set are defined as $a_{\text{high}}$, $a_{\text{medium}}$, and $a_{\text{low}}$.

Furthermore, as shown in Figure 5, we observe that the logits of the filtered image do not always induce hallucinations, but can also contribute to generating the correct token. Consequently, directly amplifying and eliminating the logits of the filtered image can reduce the probability of generating the correct token, potentially leading to additional hallucinations. To mitigate this issue, we propose an alternative approach in which the logits are reduced, rather than amplified, when the filtered image exhibits a high degree of membership in the 'high' fuzzy set, while the original image demonstrates a high degree of membership in the 'low' fuzzy set. This adjustment ensures that the final logits facilitate the generation of the correct token. Consequently, a correction coefficient is applied for Rule 3 (IF $x_1$ is low and $x_2$ is high) , $a_{\text{reduce}} = -\gamma$. Based on these definitions, the

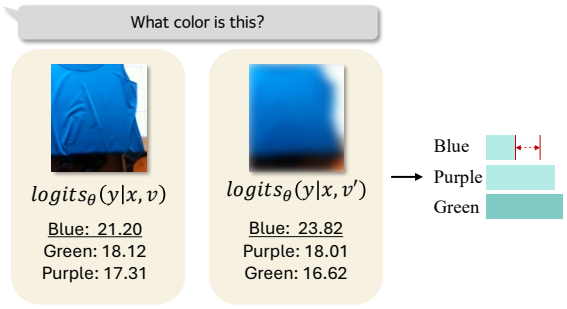


Figure 5. Applying a filter intended to amplify hallucination logits reduces confidence in the correct token ('Blue') and increases incorrect predictions ('Purple' and 'Green'), demonstrating unintended hallucination effects.

amplified logits for each rule are computed as follows:

$$y_{\text{rule}_i} = \begin{cases} a_{\text{high}} \cdot q, & i \in \{1,4\} \\ a_{\text{medium}} \cdot q, & i \in \{2,5\} \\ a_{\text{reduce}} \cdot q, & i = 3 \\ a_{\text{low}} \cdot q, & i \in \{6,7,8,9\} \end{cases} \tag{6}$$

where $q$ represents the logits of the filtered image. Each rule applies the corresponding amplification coefficient to the logits, resulting in an adjusted and refined output.

#### 3.3.3. Contrastive Decoding with Amplified Logits

The adjusted logits are modified according to specific rules and eventually combined using a weighted average. The amplified logits generated through each filter are defined as follows:

$$\text{Amplified logits} = \sum_{i=1}^{n} \sum_{j=1}^{k} \left( \frac{w_j \cdot y_j}{\sum_{j=1}^{k} w_j} \right), \tag{7}$$

where $w_j$ denotes the weight assigned to each logit $y_j$, which is normalized based on its respective weight before summation. This formulation ensures a consistent integration of logits computed across multiple filtering mechanisms.

The amplified logits are generated for each filter ($n$), and the final step involves removing the logit from the original logits that shows the greatest difference. This difference is determined by calculating the Jensen-Shannon (JS) divergence, which measures the similarity between two probability distributions. The JS divergence is defined as follows:

$$\text{JS}(P||Q) = \frac{1}{2}\text{KL}(P||M) + \frac{1}{2}\text{KL}(Q||M), \tag{8}$$

where $P$ represents the original logits distribution, $Q$ represents the amplified logits distribution, and $M$ is the average distribution given by:

$$M = \frac{1}{2}(P+Q). \tag{9}$$

The KL divergence is computed as:

$$\text{KL}(P||M) = \sum_i P(i) \log \frac{P(i)}{M(i)}. \tag{10}$$

By computing the JS divergence between the original and amplified logits, the logit with the highest divergence, indicating the most significant difference, is ultimately removed from the original logits.

Subsequently, contrastive decoding is performed according to the following equation:

$$y_t \sim ((1+\alpha) logits_\theta(y|x,v) - \alpha \cdot \text{amplified logits}), \tag{11}$$

where $logits(y|x,v)$ represents the base logits derived from the input $x$ and contextual information $v$, while the term amplified logits corresponds to the adjusted logits obtained from the previous formulation. The hyperparameter $\alpha$ regulates the trade-off between the original logits and the amplified logits. The final probability distribution is computed using the softmax function, from which the next token $y_t$ is sampled.

Furthermore, the set of candidate tokens for sampling is constrained as follows.

$$V_{\text{head}}(y_{<t}) = \Big\{ y_t \in V \mid p_\phi(y_t|x,v,y_{<t}) \geq \beta \max_{\text{t}} p_\phi(\text{t}|X_V, X_{\text{ins}}, y_{<t}) \Big\}. \tag{12}$$

This constraint ensures that only tokens with a probability at least $\beta$ times that of the most probable token are considered viable candidates. Consequently, low-probability tokens are filtered out, thereby improving decoding stability and reliability. This contrastive decoding method facilitates the generation of more coherent output while effectively mitigating input-induced decoding biases.

## 4. Experiments

### 4.1. Experimental Setting

**Implementation Details**. We evaluate the effectiveness of our FuzzyCD on state-of-the-art (SoTA) LVLMs, specifically implementing it on LLaVA-1.5 [21] and InstructBLIP [19], which utilize Vicuna-7B. In our experimental setup, the parameters $\alpha$ and $\beta$ are set to 1 and 0.1, respectively, with sharpen filtering applied as the default image filter. Unless otherwise noted, all experiments adhere to these configurations. For a consistent comparative analysis, we adopt the original model decoding strategy. All baselines are evaluated in identical experimental settings to ensure fair comparison.

**Baselines**. We compare FuzzyCD with four SoTA models for mitigating hallucination in LVLMs:

| Dataset | Setting | Method | LLaVA-1.5 | | | | InstructBLIP | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Acc. | Prec. | Rec. | F1 | Acc. | Prec. | Rec. | F1 |
| MSCOCO | Random | *default* | 82.90 | 92.00 | 72.06 | 80.82 | 80.63 | 81.19 | 79.73 | 80.46 |
| | | *+vcd* | 87.73 | 91.42 | 72.80 | 87.16 | 84.53 | 88.55 | 79.32 | 83.68 |
| | | *+icd* | 83.66 | 80.66 | 80.73 | **88.77** | 86.43 | 92.01 | 80.73 | 85.61 |
| | | *+vdd* | 83.52 | 89.25 | 76.22 | 82.22 | 80.96 | 82.05 | 72.26 | 80.63 |
| | | *+OPERA* | 86.26 | 97.14 | 74.73 | 84.47 | 86.56 | 90.72 | **81.46** | 85.84 |
| | | ***+fuzzycd*** | **88.97** | 93.20 | **84.07** | 88.40 | **86.66** | 91.16 | 81.20 | **85.89** |
| | Popular | *default* | 81.00 | 87.88 | 72.06 | 79.19 | 80.50 | 80.80 | 80.00 | 80.40 |
| | | *+vcd* | 85.38 | 86.92 | 83.28 | 85.06 | 81.47 | 82.89 | 79.32 | 81.07 |
| | | *+icd* | 80.46 | 76.39 | **88.83** | 82.14 | 82.93 | 84.45 | 80.73 | 82.55 |
| | | *+vdd* | 85.87 | 94.32 | 76.33 | 84.38 | 78.90 | 78.35 | 79.86 | 79.10 |
| | | *+OPERA* | 85.26 | 94.64 | 74.73 | 83.53 | 80.93 | 75.46 | **91.66** | 82.78 |
| | | ***+fuzzycd*** | **86.10** | 91.81 | 79.26 | **85.08** | **85.20** | 88.26 | 81.20 | **84.58** |
| | Adversarial | *default* | 78.60 | 82.89 | 72.06 | 77.10 | 77.40 | 76.11 | 79.87 | 77.94 |
| | | *+vcd* | 80.88 | 79.45 | 83.29 | 81.33 | 79.56 | 79.67 | 79.39 | 79.52 |
| | | *+icd* | 76.07 | 70.77 | **89.47** | 79.03 | 80.87 | 80.95 | 80.73 | 80.84 |
| | | *+vdd* | 83.52 | 89.25 | 76.22 | 82.22 | 81.63 | 75.70 | 81.00 | 78.26 |
| | | *+OPERA* | 83.23 | 89.84 | 74.93 | 81.71 | 76.33 | 70.15 | **91.66** | 79.47 |
| | | ***+fuzzycd*** | **83.93** | 88.61 | 77.86 | **82.89** | **81.90** | 82.35 | 81.20 | **81.77** |

Table 2. Performance comparison of LLaVA-1.5 and InstructBLIP across different settings and augmentation methods on the MSCOCO dataset. The best and second-best results are highlighted in **bold** and underline, respectively.

- **VCD** [17] amplifies and removes hallucinations by introducing distorted visual inputs, improving the robustness of the model against spurious correlations.
- **ICD** [33] utilizes instructional prompts to explicitly induce and eliminate hallucinations.
- **VDD** [36] applies a projection-based debiasing technique that removes biased directions in text embeddings, reducing both social and spurious biases without additional training.
- **OPERA** [14] mitigates hallucinations by penalizing overconfident beam search outputs and reallocating attention to image tokens via Retrospection-Allocation.

**Datasets and Evaluation Metrics**. We evaluate hallucination using three established benchmarks, with detailed descriptions provided in the supplement.

- **POPE** [34] evaluates object hallucinations through binary object presence classification in different sampling strategies.
- **MME** [10] assesses multimodal reasoning with specific subsets used to measure object-level hallucinations and generalization.
- **ROPE** [4] measures multi-object hallucinations based on recognition tasks conditioned on visual prompts.

## 4.2. Experimental Results

### 4.2.1. Evaluation on POPE

Table 2 presents the evaluation results on the COCO POPE dataset, where bold text highlights the highest accuracy scores, and underlined text indicates the second highest. The application of the proposed FuzzyCD method leads to substantial performance improvements in all evaluated LVLMs, demonstrating its broad applicability. Specifically, accuracy gains of approximately 5.43%p and 5.07%p are achieved by the LLaVA-1.5 and InstructBLIP models, respectively.

A more detailed analysis reveals that the FuzzyCD positively contributes not only to accuracy but also to precision and recall. In the Random setting, the LLaVA-1.5 model exhibits a gain in accuracy of 5.86%p, along with improvements of 0.66%p in precision, 12.14%p in recall, and 7.4%p in F1 score. These enhancements suggest that FuzzyCD supports a more balanced performance across key metrics and effectively mitigates object hallucination. InstructBLIP exhibits a 9.97%p improvement in precision, enhancing the ability of FuzzyCD to minimize false positives, thus supporting generalization ability in various LVLMs.

### 4.2.2. Evaluation on ROPE

Table 3 shows the evaluation results in the ROPE dataset under in-wild, homogeneous, heterogeneous settings, with

| Model | Multi-Object | | | Single-Object | | |
|---|---|---|---|---|---|---|
| | Wild | Hom | Het | Wild | Hom | Het |
| LLaVA1.5 | 13.96 | 31.88 | 3.98 | 13.96 | 31.88 | 3.98 |
| + vcd | 13.57 | 28.68 | 6.37 | 24.76 | 48.33 | 9.84 |
| + icd | 14.35 | 32.63 | 5.69 | 22.92 | 45.47 | 10.24 |
| + vdd | 17.8 | 40.77 | 7.40 | 27.49 | 52.12 | 11.06 |
| + OPERA | 13.20 | 37.14 | 3.82 | 13.20 | 37.14 | 3.82 |
| **+fuzzycd** | **21.12** | **46.44** | **7.45** | **29.01** | **57.88** | **11.30** |

Table 3. Performance comparison in ROPE.

bold text indicating the highest accuracy. In the in-wild setting, five objects per image are randomly selected and ordered to simulate realistic mixed-distribution scenarios. The homogeneous setting queries five instances of the same object class (e.g., AAAAA), stressing the model's ability to avoid repetitive hallucinations, while the heterogeneous setting probes five distinct classes (ABCDE) to examine robustness under various queries.

FuzzyCD outperforms all baselines in every ROPE setting. We demonstrate performance improvements of 7.16%p, 14.56%p, and 3.47%p over the standard model in the wild, homogeneous, and heterogeneous settings, respectively. Furthermore, our method consistently outperforms all recent state-of-the-art models. The particularly large gain in the homogeneous setting highlights FuzzyCD's ability to suppress repeated class hallucinations, while consistent improvements under the in-wild and heterogeneous conditions underscore the robustness of our model in various object distributions.

#### 4.2.3. Evaluation on MME

Table 4 and Figure 6 present a comparative analysis of performance metrics for the LLaVA-1.5 model and state-of-the-art methods on the MME dataset, assessing tasks of object and attribute levels. Object-level tasks include existence and count, while attribute-level tasks comprise position and color. The total score represents the aggregated performance across these metrics. In Figure 6, a star indicates the highest score among the compared methods. FuzzyCD generally improves performance on all the evaluation metrics compared to other decoding methods. Specifically, it achieves 195 in object existence, marking a 19.33%p improvement over regular decoding. For object counting, FuzzyCD attains 163.33, reflecting a gain of 56.66, while in position, it reaches 123.33, representing an increase of 9.33%p. Although there is a slight decrease in color accuracy compared to the VDD model (from 165 to 160), the overall improvements, especially in object existence and counting, demonstrate that FuzzyCD significantly enhances object-level reasoning and reduces hallucinations with large objects. The substantial improvements in existence and counting accuracy suggest enhanced object recognition and multi-object differentiation, thereby improving overall vision-language reasoning.

| Method | Object-Level | | Attribute-Level | | Total Scores |
|---|---|---|---|---|---|
| | Existence | Count | Position | Color | |
| *default* | 175.67 | 106.67 | 114 | 160 | 556.34 |
| *+vcd* | 184.66 | 138.33 | **128.67** | 153 | 604.66 |
| *+icd* | 185 | 131.66 | 113.33 | 138.33 | 568.32 |
| *+vdd* | 190 | 138.33 | 126.67 | **165** | 620 |
| *+fuzzycd* | **195** | **163.33** | 123.33 | 160 | **641.66** |

Table 4. Comparison on Object-Level and Attribute-Level tasks.

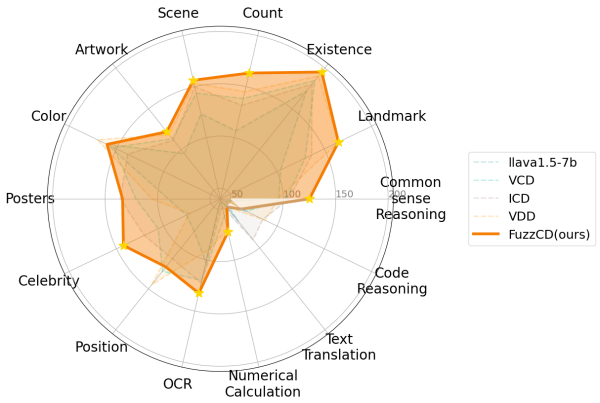


Figure 6. Result of MME in all categories.

### 4.3. Discussions

#### 4.3.1. Ablation Study

Figure 7 shows a model-scale ablation on the POPE benchmark, comparing standard decoding with our FuzzyCD across five vision-language models: LLaVA-1.5(7B), Instruct-BLIP(7B), LLaVA-Next(13B), QwenVL(32B), and QwenVL(72B). FuzzyCD consistently improves accuracy, with the largest gains in smaller models, 5.12%p for LLaVA-1.5 and 5.41%p for Instruct-BLIP(7B) and diminishing but still positive gains for larger models (4.46%p, 0.72%p, and 1.05%p, respectively). These results confirm that the FuzzyCD strategy demonstrates robustness on different model scales. Table 5 reports the accuracy on the POPE dataset using different image filters. Sobel gives the highest accuracy (89.33%), while median gives the lowest (86.53%). Bilateral, sharpen, and Gaussian filters show similar results (86.86%∼88.76%). These results suggest that model performance remains stable across filtering methods, with only minor accuracy variations.

#### 4.3.2. Qualitative Results

Figure 8 illustrates the FuzzyCD process. Given images $x_1$ and $x_2$ and a query $q$, FuzzyCD computes membership degrees (high, medium, low) based on the log-probability of next-token logits. In the example, *domino* and *sugar* are the

| Dataset | Filter | Accuracy (%) |
|---|---|---|
| POPE | Bilateral | 86.86 |
| | Median | 86.53 |
| | Sobel | **89.33** |
| | Sharpen | 88.76 |
| | Gaussian | 87.83 |

Table 5. Accuracies for different filters on the POPE dataset

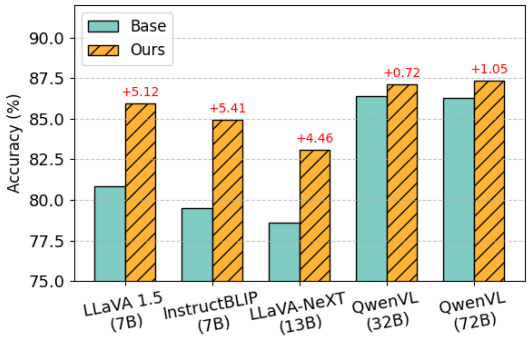


Figure 7. Model-scale ablation on the POPE dataset.

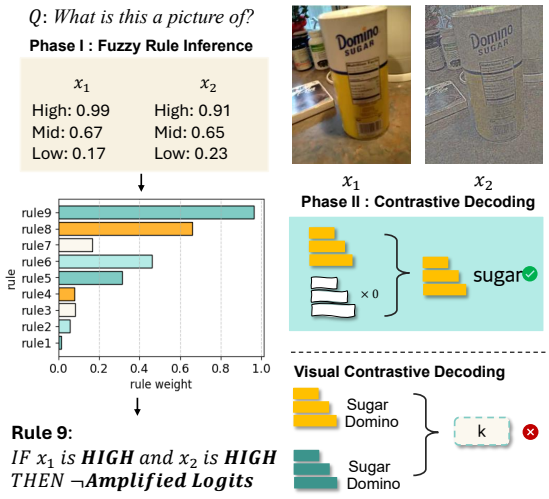


Figure 8. Visual analysis of FuzzyCD process.

top candidates, both falling into the high membership. Rule 9, which applies when both inputs have high membership, receives the highest weight, indicating that there is no hallucination. Thus, the logit of $x_2$ is preserved. In contrast, standard contrastive decoding may mistakenly suppress the correct tokens that appear in both top positions. FuzzyCD avoids this and produces more stable and accurate output.

Figure 9 shows an example on the WHOOPS dataset that involves an unusual image of two young children with backpacks standing on a snowy mountain. The question is:

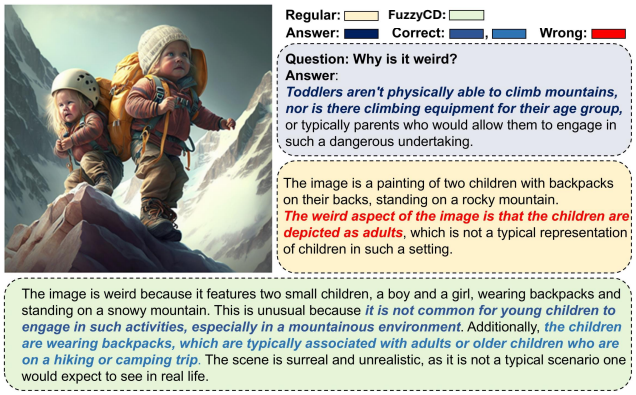


Figure 9. An example of hallucination correction using the proposed FuzzyCD on the WHOOPS dataset.

"Why is it weird?" The correct answer highlights the inappropriateness of such an activity for children and their portrayal of an adult. LLaVA-1.5 partially captures this, noting the presence of children in an unusual setting but missing a deeper context. In contrast, LLaVA-1.5 with FuzzyCD provides a more comprehensive explanation, addressing the setting, activity, and portrayal of the children. This demonstrates FuzzyCD's ability to reduce hallucinations and improve contextual understanding in complex reasoning tasks.

## 5. Concluding Remarks

This paper proposes a novel method to address object hallucination in LVLMs by detecting hallucinations through log-probability analysis, rather than relying solely on output-level corrections. By examining logits-based metrics, we show that hallucinations can be identified early and effectively mitigated. Based on these insights, we introduce FuzzyCD, which dynamically adjusts logits using fuzzy logic and confidence scores. This improves prediction accuracy by increasing the sensitivity to visual cues while reducing the dependence on textual biases. The method is training-free, does not require additional labeled data, and is scalable across various LVLM architectures and tasks, offering strong practical applicability. Experimental results on multiple benchmarks and state-of-the-art LVLMs demonstrate that FuzzyCD effectively reduces hallucinations and improves recognition performance, highlighting its potential to enhance both reliability and accuracy without additional training.

**Limitations**. As this study focuses mainly on hallucination mitigation and its empirical validation, more analysis is needed on attribute-level tasks and general VQA performance to assess the broader applicability. Future work should explore its efficacy across larger datasets and diverse model families, and compare it against alternative approaches.

**Acknowledgments**. This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2020-II201361, Artificial Intelligence Graduate School Program (Yonsei University); No. RS-2021-II212068, Artificial Intelligence Innovation Hub).

## References

[1] 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*, pages 1955–1960, Austin, Texas, 2016. Association for Computational Linguistics. 1

[2] Plamen P Angelov and Dimitar P Filev. An approach to online identification of takagi-sugeno fuzzy models. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(1):484–498, 2004. 2

[3] Seok-Jun Buu and Sung-Bae Cho. A transformer network calibrated with fuzzy logic for phishing url detection. *Fuzzy Sets and Systems*, 517:109474, 2025. 2

[4] Xuweiyi Chen, Ziqiao Ma, Xuejun Zhang, Sihan Xu, Shengyi Qian, Jianing Yang, David Fouhey, and Joyce Chai. Multi-object hallucination in vision language models. *Advances in Neural Information Processing Systems*, 37:44393–44418, 2024. 6

[5] Yuyan Chen, Qiang Fu, Yichen Yuan, Zhihao Wen, Ge Fan, Dayiheng Liu, Dongmei Zhang, Zhixu Li, and Yanghua Xiao. Hallucination detection: Robustly discerning reliable answers in large language models. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 245–255, 2023. 2

[6] Sung-Bae Cho and J.H. Kim. Multiple network fusion using fuzzy logic. *IEEE Transactions on Neural Networks*, 6(2): 497–501, 1995. 2

[7] Jiwan Chung and Youngjae Yu. Vlis: Unimodal language models guide multimodal language generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 700–721, 2023. 1

[8] Yuhao Dong, Zuyan Liu, Hai-Long Sun, Jingkang Yang, Winston Hu, Yongming Rao, and Ziwei Liu. Insight-v: Exploring long-chain visual reasoning with multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9062–9072, 2025. 1

[9] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14303–14312, 2024. 1

[10] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, and Rongrong Ji. Mme: A comprehensive evaluation benchmark for multimodal large language models. *ArXiv*, abs/2306.13394, 2023. 6

[11] 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*, pages 14375–14385, 2024. 1, 2

[12] Nuno M. Guerreiro, Elena Voita, and André Martins. Looking for a needle in a haystack: A comprehensive study of hallucinations in neural machine translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1059–1075, Dubrovnik, Croatia, 2023. Association for Computational Linguistics. 2

[13] 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*, pages 18135–18143, 2024. 1

[14] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13418–13427, 2024. 1, 6

[15] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27036–27046, 2024. 1

[16] Liqiang Jing, Ruosen Li, Yunmo Chen, and Xinya Du. FaithScore: Fine-grained evaluations of hallucinations in large vision-language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 5042–5063, Miami, Florida, USA, 2024. Association for Computational Linguistics. 2

[17] 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*, pages 13872–13882, 2024. 1, 2, 6

[18] Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. Halueval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, 2023. 2

[19] 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 *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 1, 5

[20] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Mitigating hallucination in large multi-modal models via robust instruction tuning. In *The Twelfth International Conference on Learning Representations*, 2023. 1

[21] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee.

Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024. 1, 5

[22] Xinyu Lyu, Beitao Chen, Lianli Gao, Hengtao Shen, and Jingkuan Song. Alleviating hallucinations in large vision-language models through hallucination-induced optimization. *Advances in Neural Information Processing Systems*, 37:122811–122832, 2024. 1

[23] Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. Llms know more than they show: On the intrinsic representation of llm hallucinations. In *ICLR*, 2025. 2, 3

[24] Yifu Qiu, Yftah Ziser, Anna Korhonen, Edoardo Ponti, and Shay Cohen. Detecting and mitigating hallucinations in multilingual summarisation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8914–8932, Singapore, 2023. Association for Computational Linguistics. 2

[25] 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*, pages 8748–8763. PMLR, 2021. 1

[26] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. In *Conference on Empirical Methods in Natural Language Processing*, 2018. 2

[27] Rico Sennrich, Jannis Vamvas, and Alireza Mohammadshahi. Mitigating hallucinations and off-target machine translation with source-contrastive and language-contrastive decoding. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 21–33, St. Julian's, Malta, 2024. Association for Computational Linguistics. 2

[28] Aviv Slobodkin, Omer Goldman, Avi Caciularu, Ido Dagan, and Shauli Ravfogel. The curious case of hallucinatory (un) answerability: Finding truths in the hidden states of over-confident large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3607–3625, 2023. 2, 3

[29] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. Aligning large multimodal models with factually augmented RLHF. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13088–13110, Bangkok, Thailand, 2024. Association for Computational Linguistics. 2

[30] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. Aligning large multimodal models with factually augmented RLHF. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13088–13110, Bangkok, Thailand, 2024. Association for Computational Linguistics. 1

[31] Tomohiro Takagi and Michio Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE transactions on systems, man, and cybernetics*, (1):116–132, 1985. 2

[32] 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

[33] Xintong Wang, Jingheng Pan, Liang Ding, and Chris Biemann. Mitigating hallucinations in large vision-language models with instruction contrastive decoding. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 15840–15853, 2024. 2, 6

[34] Kun Zhou Jinpeng Wang Wayne Xin Zhao Yifan Li, Yifan Du and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. 1, 6

[35] Hanning Zhang, Shizhe Diao, Yong Lin, Yi Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. R-tuning: Instructing large language models to say 'i don't know'. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7106–7132, 2024. 2

[36] Yi-Fan Zhang, Weichen Yu, Qingsong Wen, Xue Wang, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. Debiasing multimodal large language models. *arXiv preprint arXiv:2403.05262*, 2024. 6

[37] 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*, pages 600–610, Harbin, China, 2023. Chinese Information Processing Society of China. 2

[38] 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. In *The Twelfth International Conference on Learning Representations*. 1

[39] Derui Zhu, Dingfan Chen, Qing Li, Zongxiong Chen, Lei Ma, Jens Grossklags, and Mario Fritz. Pollmgraph: Unraveling hallucinations in large language models via state transition dynamics. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 4737–4751, 2024. 2

[40] Xi Zhu, Zhendong Mao, Chunxiao Liu, Peng Zhang, Bin Wang, and Yongdong Zhang. Overcoming language priors with self-supervised learning for visual question answering. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 1083–1089, 2021. 2