Retrieval Visual Contrastive Decoding to Mitigate Object Hallucinations in Large Vision-Language Models

Jihoon Lee[*]
Yonsei University
Seoul, South Korea
jihoonlee98@yonsei.ac.kr

Min Song[†]
Yonsei University, ONOMA AI
Seoul, South Korea
min.song@onomaai.com

Abstract

Despite significant advancements in Large Vision-Language Models, Object Hallucination (OH) remains a persistent challenge. Building upon prior studies on contrastive decoding that address this issue without requiring additional model training, we introduce RVCD (Retrieval Visual Contrastive Decoding), an advanced method to suppress OH. RVCD leverages both negative and positive images at the logit level, explicitly referencing AI-generated images designed to represent a single concept. Our approach demonstrates substantial improvements over existing decoding-based methods. Code and data are released at https://github.com/JiHoonLee9898/RVCD.

1 Introduction

Large Vision Language Models (LVLMs) are models designed to generate sophisticated textual responses based on multimodal inputs of images and text. In recent years, successful experiments on integrating vision encoders with language models have demonstrated promising progress in this field (Zhu et al., 2023; Liu et al., 2023b; Zhang et al., 2024).

However, Large Vision Language Models (LVLMs) are still not free from the issue of Object Hallucination (OH), which refers to the phenomenon where LVLMs erroneously generate hallucinated objects and descriptions in their outputs (Rohrbach et al., 2018).

OH can be categorized into three types: generating descriptions of objects that do not exist in the image (existence), misdescribing the attributes of existing objects (attribute), and incorrectly describing the relationships between objects (relationship) (Gunjal et al., 2023; Zhai et al., 2023).

Previous studies have demonstrated that even more sophisticated and larger LVLMs are not free from the issue of Object Hallucination (OH) (Dai et al., 2022; Li et al., 2023; Guan et al., 2023).

To address Object Hallucination (OH), which undermines the reliability of LVLMs, various methodologies have been proposed. These include approaches that mitigate OH by modifying the outputs generated by LVLMs (Zhou et al., 2023), introducing self-correction pipelines (Yin et al., 2023), or employing decoding-based methods (Huang et al., 2023; Leng et al., 2023; Chen et al., 2024b; Zhuang et al., 2024).

Among these methodologies, visual contrastive decoding-based approaches are particularly attractive and practical because they mitigate OH without requiring additional training for the models (Leng et al., 2023; Chen et al., 2024b; Zhuang et al., 2024). These methods distort the input source images (Leng et al., 2023), or zoom into the local views containing important objects in the source images (Chen et al., 2024b; Zhuang et al., 2024) to generate logits for regulation. Theses logits modify or replace the logits generated from the original input images, contributing to producing better outputs.

However, despite the excellence of their methods, they do not fully exploit the potential of visual contrastive decoding — the potential that the images used to generate logits for regulation do not always need to be transformations of the original images.

We introduce a novel method called RVCD (Retrieval-Visual Contrastive Decoding), which maximizes the regulation strength by retrieving and leveraging multiple explicit images as regulatory targets. The explicit images we use are designed to encapsulate a *single concept*, enabling the contrastive decoding process to add or subtract the desired or undesired concept effectively, thereby allowing the adjusted logits to clearly align with the target image.

Our method retrieves multiple explicit reference

[*]First author.
[†]Corresponding author.

images, generates negative logits to be regulated, and recovers positive logits lost during the regulation process, utilizing them at every decoding step for token generation. These explicit reference images are created using image generation models to represent single concepts and are ultimately selected based on agreement conditions between the image generation models and LVLMs. Similar to prior studies, our method can be easily applied to open-source LVLMs, such as MiniGPT-4 (Chen et al., 2023), LLaVA (Liu et al., 2023b), and mPLUG-Owl2 (Ye et al., 2023).

Our contributions are summarized as follows: (1) We propose a novel plug-and-play decoding method called RVCD (Retrieval-Visual Contrastive Decoding). This method is train-free, strongly regulates OH, and simultaneously preserves high output text quality. (2) We provide a retrieval database utilized in RVCD. All images in the database are aligned with the consensus between diffusion-based image generation models and LVLMs, and they represent single concepts. This database enhances the explainability of our RVCD and can serve as a resource for future research leveraging explicit concepts in visual contrastive decoding. (3) Through comprehensive experiments, we demonstrate the strong OH reduction capability of RVCD, which significantly outperforms existing methods.

2 Related Work

Object Hallucination (OH) refers to the phenomenon where BERT-based Vision Language Models (VLMs) (Li et al., 2019; Radford et al., 2021), or more recent LVLMs (Liu et al., 2023b; Zhu et al., 2023; Tu et al., 2023; Cui et al., 2023; Wang et al., 2024; Zhou et al., 2024b), erroneously generate unfaithful contents. Gunjal et al. (2023) and Zhai et al. (2023) categorized OH into three types: existence, attribute, and relationship OH. These correspond to generating descriptions of non-existent objects, producing misleading descriptions, and generating incorrect descriptions of relationships between existing objects, respectively.

The most dominant metric for evaluating OH is CHAIR (Rohrbach et al., 2018). This metric can be used in scenarios where a finite synonym dictionary and a set of ground truth objects mapped to an image are defined. It evaluates the proportion of synonyms appearing in LVLM outputs that are not defined in the ground truth object set (CHAIR_I), and the proportion of sentences where CHAIR_I is non-zero (CHAIR_S). Another well-known recent metric is POPE (Li et al., 2023), which measures the degree of OH using precision, recall, and accuracy. To do so, POPE frames a binary classification problem for the LVLM, evaluating its outputs based on the inclusion of positive or negative assertions (e.g., "yes" or "no"). Additionally, the traditional and standard text generation quality metric BLEU (Papineni et al., 2002) is still utilized in recent studies (Chen et al., 2024b; Zhuang et al., 2024). BLEU serves as an additional indicator to ensure that little sacrifice in text quality occurs while mitigating OH (Chen et al., 2024b).

Efforts to mitigate Object Hallucination (OH) have been ongoing since the introduction of the CHAIR metric by Rohrbach et al. (2018), yet OH remains an unsolved challenge despite advancements in LVLMs (Dai et al., 2022; Li et al., 2023; Zhou et al., 2024a). No LVLM to date has completely resolved OH in its outputs. To address OH, recent approaches have explored various strategies. Sun et al. (2023) and Jing and Du (2024) proposed reinforcement learning-based methods for training model parameters, while Xing et al. (2024) introduced a token reordering approach to mitigate the issue of long-term decay in Rotary Position Encoding (RoPE). Zhou et al. (2023) and Yin et al. (2023) proposed post-hoc or self-correction pipelines to reduce OH in final text outputs. Meanwhile, Huang et al. (2023), Leng et al. (2023), Chen et al. (2024b), Zhuang et al. (2024), and Yang et al. (2024b) introduced decoding based strategies. These approaches demonstrated that adjusting logits during the decoding steps, without training or modifying output text directly, can effectively reduce OH.

However, despite the excellence of the core idea of visual contrastive decoding (VCD) (Leng et al., 2023), including the above studies that leverage it, they fail to fully exploit its hidden potential. Specifically, they overlook the fact that the types of images used to generate regulated logits are not necessarily restricted to variations of the original input image.

Building on the intuition that explicit images for regulation can be derived from the external database, we propose Retrieval-Visual Contrastive Decoding (RVCD). Specifically, we constructed a database by generating explicit AI-created images that best represent the single concept of each word in the finite vocabulary used for OH evaluation (Rohrbach et al., 2018). In our approach,

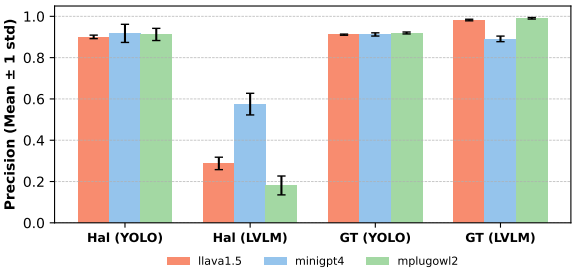

Figure 1: Detection precision for YOLO and LVLM detectors on MSCOCO Validation 2014 (Lin et al., 2014). Hal $(\cdot)$ shows the proportion of hallucinated objects from greedy-decoded captions detected by YOLO and LVLMs VQA that were true hallucinations. GT $(\cdot)$ illustrates the proportion of objects correctly identified as existing by YOLO and LVLMs. While both perform similarly in detecting existing objects, YOLO excels in hallucination detection, motivating us to transfer YOLO's strength to LVLMs for correcting hallucinated objects. The statistical details are provided in Appendix C.

multiple regulated and preserved logits are generated from images retrieved from this database.

3 Background and Motivation

3.1 Problem Definition

A typical LVLM, parameterized by θ, encodes an input text query x and an input image v, integrates the encoded embeddings to generate a multimodal embedding, and processes it autoregressively:

$$y_t \sim p_\theta(\cdot \mid v, x, y_{<t}) \propto \exp f_\theta(\cdot \mid v, x, y_{<t}), \quad (1)$$

where y_t represents the token of the time step (t), $y_{<t}$ is the sequence of output tokens generated up to the time step $(t-1)$, and f_θ is the logit distribution (unnormalized log-probabilities) generated by the LVLM ($\mathcal{M}_\theta^{\text{LVLM}}$).

Object Hallucination (OH) occurs when the information from the input image v conflicts with some tokens in y. To mitigate OH, y must describe only the information present in v while maintaining high text generation quality.

3.2 Our Approach to OH Mitigation

We propose an approach to mitigate Object Hallucination (OH) by detecting hallucinated objects that should not have been generated in the greedy-decoded draft output and regulating this information to produce the target output. To this end, we first conducted an experiment to assess whether LVLMs can self-check OH occurrences in their draft outputs. However, as shown in Figure 1, object detection capability of LVLMs is insufficient to accurately detect hallucinated objects in their draft captions.

On the other hand, traditional object detection (OD) model YOLO (Redmon et al., 2015), which lack linguistic capabilities, shows much better hallucination detection precision than LVLMs VQA (Vision Question Answering) outputs. In this work, we utilized YOLOv8x (Ultralytics, 2023) due to its significant influence and widespread adoption in the deep learning community for both training and inference (Ultralytics, 2024).

We hypothesize that providing accurate OH detection information from the OD model at each decoding step of the LVLM will suppress hallucinated tokens while maintaining fluent language generation. To achieve this, we generate multiple logits from retrieved explicit images based on the OH detection information from the OD model and regulate them at each decoding step to mitigate OH.

Our Goal: Conveying accurate OH detection information of OD models to the LVLM's token generation stage to minimize OH in the target output while maintaining fluent language generation capability.

4 Methodology

The overview of our method is illustrated in Figure 2.

First, we generate a greedy-decoded text result for the input image using the LVLM, which we refer to as the draft caption. Simultaneously, we obtain a list of objects present in the same image using an object detection (OD) model such as YOLO.

Objects mentioned in the draft caption but not detected by the OD model are classified as negative objects. Objects detected by both the draft caption and the OD model are classified as positive objects.

Our goal is to suppress the generation of tokens related to negative objects while preserving the representation of positive objects at every token generation step, by corresponding logits generated from retrieved explicit images. We describe the details of each step below.

4.1 Generate Reference Images Corresponding to CHAIR Dictionary Words

For the quantitative evaluation of OH in output captions, prior studies have utilized the dictionary of CHAIR (Rohrbach et al., 2018). This dictionary is

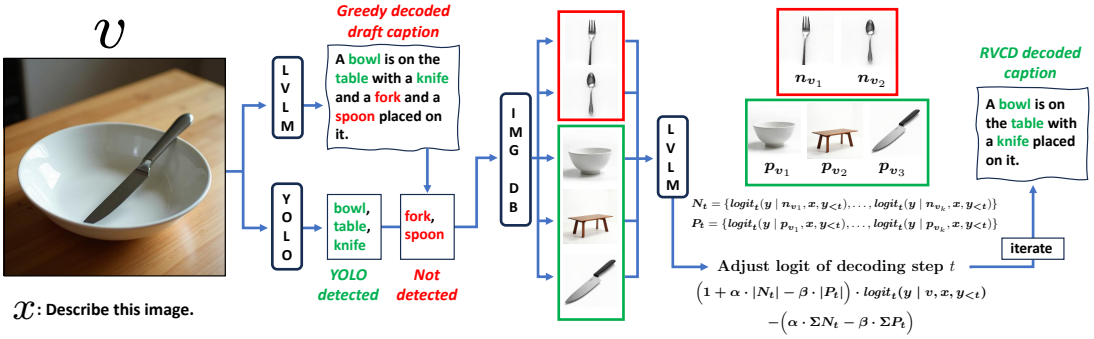

Figure 2: Overall pipeline of our RVCD. x denotes the input prompt, and v denotes the input image. n_{vi} and p_{vi} are images retrieved from the image database, representing single-concept images for objects identified as hallucinations (appearing only in the draft caption) and ground truth (appearing in both the OD model and draft caption), respectively. N_t and P_t represent the sets of logits generated from n_{vi} and p_{vi}, respectively. At each decoding step, the LVLM processes x, v, N_t, P_t, and ongoing output tokens $y_{<t}$, which are then integrated according to our proposed formula. This iterative decoding process produces the final caption of RVCD.

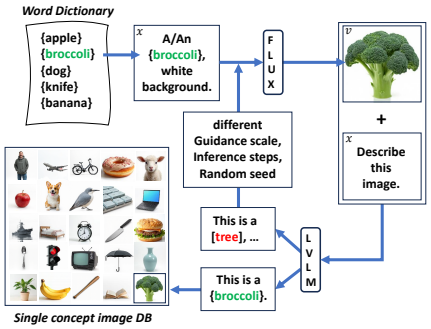

Figure 3: AI generated single concept image DB. we adopted FLUX.1-dev (Yang et al., 2024a) to generate 336 * 336 pixels images representing only a single concept corresponding to each word in the MSCOCO objects synonyms dictionary and stored them in an image database. Images were stored in the database only if the LVLM's output captions and the image generation model's input prompts both mentioned the corresponding concept. Otherwise, the hyperparameters of the image generation model were adjusted, and the images were regenerated. This process was repeated until images were generated for every word in the dictionary.

used to extract a list of objects from output descriptions via natural language processing (NLP) and compare it against a ground truth list. We extend this dictionary into the visual domain. Specifically, for every words (over 400) in the dictionary including the representative terms of 80 MSCOCO objects and their synonyms, we generated AI images that represent only the corresponding concept and mapped them to their respective entries as Figure 3.

We adopted FLUX.1-dev (Yang et al., 2024a) as the image generation model and used the prompt "An/A {object}, white background" to create images that exclusively represent the single concept of the object. These images were then re-fed to the LVLM along with the prompt "Describe this image in detail." We adopted llava-1.5 (Liu et al., 2023b) as the LVLM. The image is stored only if its caption included the {object}, indicating alignment between the intended prompt of the image generation model and the interpretation by the LVLM.

The extended CHAIR dictionary, which includes an image database mapped to every word, serves as a reference database that retrieves a corresponding reference image visually representing the single concept associated with the word.

4.2 Comparison Between Draft Caption and OD Detected Object List

The base equation of general greedy decoding is as follows:

$$y_t = \arg\max(\text{softmax}[f_\theta(\cdot \mid v, x, y_{<t})]). \quad (2)$$

Here, f_θ is the logit distribution generated by the LVLM ($\mathcal{M}_\theta^{\text{LVLM}}$). Using greedy decoding with the LVLM, we obtain a draft caption for the input image and extract all mentioned objects to create a draft objects list.

Similarly, we use the OD model to generate a detected objects list for the same input image. Duplicate objects in the draft objects list and the detected objects list are represented as a single unique object. Objects that exist in the draft objects list but not in the detected objects list are defined as N (negative objects). Objects that exist in both the draft objects list and the detected objects list are defined as P

(positive objects). This preprocessing step aims to exclude negative objects tokens while preserving positive objects tokens in the target output.

4.3 Retrieval Visual Contrastive Decode

For each object in N and P, we retrieve a single corresponding image from the single concept image DB, and generate the output using the following formulas. To incorporate the concepts of P_t and N_t at each decoding step t: P_t is a set of logits computed using images v_{p_i} from P (positive objects list), where each image is retrieved from the single concept image DB.

$$P_t = \{f_\theta(\cdot \mid v_{p_1}, x, y_{<t}), f_\theta(\cdot \mid v_{p_2}, x, y_{<t}), \ldots, f_\theta(\cdot \mid v_{p_k}, x, y_{<t})\}. \quad (3)$$

N_t is a set of logits computed using images v_{n_i} from N (negative objects list), where each image is retrieved from the single concept image DB.

$$N_t = \{f_\theta(\cdot \mid v_{n_1}, x, y_{<t}), f_\theta(\cdot \mid v_{n_2}, x, y_{<t}), \ldots, f_\theta(\cdot \mid v_{n_m}, x, y_{<t})\}. \quad (4)$$

The adjusted logit at time t is defined as:

$$f_{\text{adjusted}_t}(\cdot \mid v, x, y_{<t}) = f_\theta(\cdot \mid v, x, y_{<t}) \cdot \Big(1 + \alpha \cdot \text{len}(N) - \beta \cdot \text{len}(P)\Big) - \Big(\alpha \cdot \text{sum}(N_t) - \beta \cdot \text{sum}(P_t)\Big), \quad (5)$$

by simplifying the following expression:

$$f_{\text{adjusted}_t}(\cdot \mid v, x, y_{<t}) = \mathcal{O}_{\text{logit}} + \alpha\left(\mathcal{O}_{\text{logit}} - N_{t_1}\right) + \cdots + \alpha\left(\mathcal{O}_{\text{logit}} - N_{t_m}\right) + \beta\left(P_{t_1} - \mathcal{O}_{\text{logit}}\right) + \cdots + \beta\left(P_{t_k} - \mathcal{O}_{\text{logit}}\right), \quad (6)$$

where $\mathcal{O}_{\text{logit}}$ denotes $f_\theta(\cdot \mid v, x, y_{<t})$ and N_{t_i}, P_{t_j} are individual logits from the sets N_t and P_t. The logit distribution $f_{\text{adjusted}_t}(\cdot \mid v, x, y_{<t})$ is computed by the same model parameters θ, Using the negative and positive logits, scaled by their respective parameters α and β. len$(\cdot)$ represents the length of the list containing negative or positive images. Note that N and N_t have the same length, and P and P_t also have the same length. The final output token at decoding step t is defined as:

$$RVCD_{y_t} = \arg\max(\text{softmax}[f_{\text{adjusted}_t}(\cdot \mid v, x, y_{<t}]). \quad (7)$$

Here, the final output token index at decoding step t is $RVCD_{y_t}$, which is obtained as the arg max from the softmax of f_{adjusted_t}.

4.4 Addressing Challenges with Negative Logits: Why β and Positive Logits?

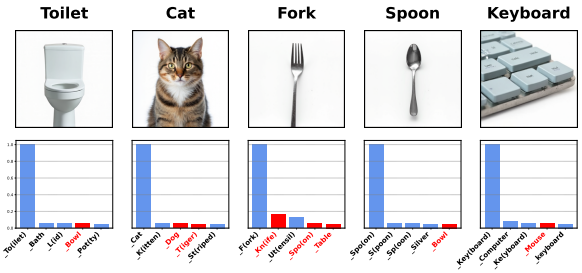

Figure 4: Top-5 token probabilities for each single concept images. When an LVLM is tasked with responding to an image using a single word, it frequently includes tokens representing other objects registered in the MSCOCO dictionary among its top-5 tokens, even for single-concept images. Details in Appendix F.

In RVCD, negative images represent a single concept. However, LVLMs often assign probabilities to tokens for commonly co-occurring objects (Li et al., 2023; Favero et al., 2024; Chen et al., 2024a) even when they are not explicitly present in the single concept image. For instance, with an image of a single fork, llava-1.5 ranks "fork" highest but still place "knife," "spoon," or "table" among the top predictions (Figure 4), due to their typical association in a kitchen setting. This behavior introduces a risk in RVCD: subtracting multiple logits generated from negative images could unintentionally suppress representations of objects that are actually part of the ground truth. Removing these unintended yet valid objects can degrade caption quality.

To address this, we preserve ground truth object representations by reintroducing information from positive logits using a parameter β. Positive logits are derived from reference images of objects identified by both the LVLM and the OD model. Our ablation study highlights the importance of β.

5 Experiments

Benchmarks. Following prior studies evaluating the performance of decoding methods (Chen et al., 2024b; Zhuang et al., 2024), we assessed our RVCD using CHAIR (Rohrbach et al., 2018), BLEU (Papineni et al., 2002), and POPE (Li et al., 2023) metrics on the MSCOCO dataset (Lin et al., 2014). Additionally, we conducted quantitative evaluations on the MME benchmark (Fu et al., 2023), and qualitative evaluation benchmark LLaVA-Bench (Liu et al., 2023a). These experiments comprehensively evaluate the accuracy and

Algorithm 1: RVCD Decoding

Require: LVLM $\mathcal{M}_\theta^{\text{LVLM}}$, text query x, image input v, object detection model O_D, image database *imgDB*, finite word dictionary D.

Output: Adjusted RVCD decoded tokens $RVCDy_0, \ldots, RVCDy_t$.

Draft Decoding:
repeat
  For each decoding step, greedy decode:
  $y_t = \arg\max(\text{softmax}[f_\theta(\cdot \mid v, x, y_{<t})])$
until Obtain decoded sequence $y_0, \ldots, y_t$.
Combine $y_0, \ldots, y_t$ into *draft*.
Object Lists Generation:
Extract words from *draft* that are elements of D to form a *draft object list*.
Apply O_D to input image v to detect all objects and obtain *OD object list*.
draft object list $\leftarrow$ List(Set(*draft object list*))
OD object list $\leftarrow$ List(Set(*OD object list*))
Positive and Negative Object Pairing:
$P \leftarrow []$, $N \leftarrow []$
for o_i in *draft object list* **do**
  if $o_i \in$ *OD object list* **then**
    Append (retrieved v_{p_i} at *imgDB* from o_i) to P.
  else
    Append (retrieved v_{n_i} at *imgDB* from o_i) to N.
  end if
end for
RVCD Adjusted Decoding:
repeat
  For each decoding step t:
  RVCD decode with input v, x, N_t, P_t, $y_{<t}$.
  Definitions:
  N_t: $\{f_\theta(\cdot \mid v_{n_i}, x, y_{<t}) \mid v_{n_i} \in N\}$
  P_t: $\{f_\theta(\cdot \mid v_{p_i}, x, y_{<t}) \mid v_{p_i} \in P\}$
  sum$(\cdot)$: The element-wise sum of all logits.
  Compute *RVCD*$_{y_t}$:

$$\begin{aligned}&\textit{RVCD}_{\textit{logit}_t} = \\ &\big(1 + \alpha \cdot \textit{len}(N) - \beta \cdot \textit{len}(P)\big) \\ &\cdot f_\theta(\cdot \mid v, x, y_{<t}) \\ &- \big(\alpha \cdot \textit{sum}(N_t) - \beta \cdot \textit{sum}(P_t)\big).\end{aligned}$$

$$\textit{RVCD}_{y_t} = \arg\max(\text{softmax}[\textit{RVCD}_{\textit{logit}_t}])$$

until Obtain decoded sequence $RVCDy_0$, $\ldots$, $RVCDy_t$.
Combine $RVCDy_0, \ldots, RVCDy_t$ into *RVCD output text*.

quality of the text generated by RVCD from the perspective of OH mitigation.

Baselines. Given that RVCD is a decoding method, we compared it with general decoding strategies such as greedy decoding and beam search, as well as established state-of-the-art (SOTA) decoding methods: DOLA (Chuang et al., 2023), OPERA (Huang et al., 2023), VCD (Leng et al., 2023), and HALC (Chen et al., 2024b). All evaluations were conducted under identical experimental settings.

LVLM Backbones. We adopted three widely used 7B backbones for our experiments: MiniGPT-4 V2 with vicuna-7b (Chen et al., 2023), LLaVA-1.5 (Liu et al., 2023b), and mPLUG-Owl2 (Ye et al., 2023). These backbones were selected to enable the most direct comparison with results from prior decoding-based studies (Chen et al., 2024b; Zhuang et al., 2024).

5.1 CHAIR, BLEU and POPE on MSCOCO

To reproduce the evaluation methodologies of prior studies and assess the performance of RVCD under the same conditions, we used their identical experimental setup (Huang et al., 2023; Liu et al., 2023b). Specifically, we employed the validation set of MSCOCO 2014 (Lin et al., 2014), randomly sampling 500 images five times with replacement and reporting the mean and standard deviation for CHAIR and POPE metrics.

CHAIR. CHAIR (Caption Hallucination Assessment with Image Relevance) quantifies OH in image captioning tasks (Rohrbach et al., 2018). It assumes the existence of a ground truth object list for each image and uses a dictionary to map objects in the output captions to representative MSCOCO synonyms. The primary metrics are:
CHAIR_I: The ratio of objects in the captions that do not appear in the ground truth object list to the total number of objects mentioned in the captions.
CHAIR_S: The proportion of sentences with hallucination (i.e., sentences where CHAIR_I is non-zero). Lower values of CHAIR_I and CHAIR_S indicate lower levels of OH. In line with prior studies, we performed image captioning using the same prompt: "Please describe this image in detail." The results are shown in Table 1. In addition to CHAIR_I and CHAIR_S, we provide BLEU scores (Papineni et al., 2002).

Methods	LLaVA-1.5			MiniGPT-4			mPLUG-Owl2		
	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑
Greedy	22.08±1.05	7.08±0.37	16.06±0.17	20.32±1.45	7.03±0.59	16.17±0.26	23.87±0.92	8.77±0.41	15.43±0.20
Beam Search	20.60±1.39	6.95±0.28	16.33±0.09	20.64±0.74	7.32±0.63	16.55±0.26	21.60±1.14	8.02±0.39	15.61±0.28
DoLA	21.36±0.65	6.82±0.20	16.11±0.12	20.36±1.87	7.08±0.68	16.10±0.25	24.40±1.65	8.76±0.52	15.46±0.24
OPERA	18.72±1.20	6.56±0.39	**16.65**±0.21	19.44±1.71	7.22±0.71	17.77±0.25	20.24±0.79	7.80±0.38	15.49±0.10
VCD	23.24±1.17	7.73±0.28	14.97±0.24	21.72±1.26	8.08±0.40	15.92±0.24	26.72±1.57	10.08±0.60	14.27±0.29
HALC	18.60±0.70	6.03±0.32	16.32±0.13	15.36±2.26	5.55±0.71	**17.83**±0.38	21.08±1.37	7.54±0.49	**15.63**±0.26
RVCD	**11.32**±0.92	**3.87**±0.35	15.48±0.13	**9.00**±1.17	**3.61**±0.50	15.98±0.33	**10.04**±1.77	**3.73**±0.54	14.78±0.22

Table 1: The averages and sample standard deviations of CHAIR and BLEU metrics with different decoding baselines were calculated over five different sampling seeds, each involving a random sampling of 500 instances from the MSCOCO dataset. Lower scores in $CHAIR_S$ and $CHAIR_I$ indicate less OH, while higher BLEU scores reflect better caption quality.

Methods	LLaVA-1.5			MiniGPT-4			mPLUG-Owl2		
	Accuracy ↑	Precision ↑	F_1 ↑	Accuracy ↑	Precision ↑	F_1 ↑	Accuracy ↑	Precision ↑	F_1 ↑
Greedy	72.19±6.10	65.28±5.49	77.86±3.88	62.98±9.36	58.72±7.28	72.24±5.38	74.36±5.89	67.23±5.59	79.23±3.86
Beam Search	78.27±4.47	71.94±4.96	81.28±3.17	67.51±7.49	62.67±6.89	73.82±4.68	80.17±5.08	74.30±6.01	82.64±3.72
DoLA	72.48±6.10	65.54±5.54	78.04±3.90	72.26±3.96	75.41±7.04	70.86±3.09	74.56±5.85	67.43±5.59	79.34±3.85
OPERA	74.43±4.11	67.43±3.93	78.94±2.71	67.49±6.74	62.41±6.05	73.89±4.16	79.01±5.62	72.71±6.36	81.98±4.03
VCD	69.86±3.48	63.44±3.02	75.87±2.14	61.79±3.39	58.97±3.07	67.34±2.02	73.66±3.80	67.33±3.76	77.93±2.53
HALC	72.48±6.10	65.54±5.54	78.04±3.90	72.26±3.96	75.41±7.04	70.86±3.09	74.54±5.85	67.42±5.59	79.33±3.84
RVCD	**88.54**±2.59	**89.92**±4.70	**88.43**±2.33	**85.96**±2.37	**88.14**±4.34	**85.63**±2.05	**87.45**±1.64	**87.91**±2.89	**87.41**±1.44

Table 2: POPE evaluation results on MSCOCO dataset of LVLMs with different decoding baselines designed to mitigate OH. Higher accuracy, precision, and F_1 indicate better performance.

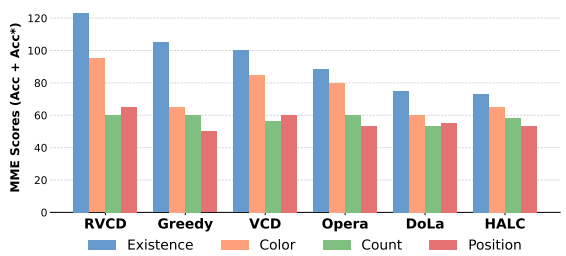

Figure 5: Comparison of different decoding baselines on MME Metric with llava-1.5 as a backbone LVLM. Refer to Table 8 for detailed information, including MiniGPT-4 and mPLUG-Owl2.

BLEU. BLEU (Bilingual Evaluation Understudy) (Papineni et al., 2002) is a traditional metric for evaluating caption quality based on n-gram matching rates. As shown in Table 1, our RVCD significantly reduces OH compared to other decoding methods while maintaining comparable BLEU scores. This demonstrates the effectiveness of retrieval-based negative and positive logits adjustment of our RVCD.

POPE. POPE (Polling-based Object Probing Evaluation) (Li et al., 2023) evaluates OH through binary classification. The LVLM is prompted to answer "yes" or "no" to whether specific objects exist in an image. POPE provides three evaluation scenarios: random, popular, and adversarial. Detailed descriptions of these options are provided by Li et al. (2023). As shown in Table 2, our RVCD significantly outperformed all other methods in accuracy, precision, and F_1 scores. This demonstrates that RVCD effectively suppresses OH of LVLMs.

MME. MME (The Multimodal Large Language Model Evaluation) (Fu et al., 2023) is a quantitative evaluation benchmark, similar to CHAIR, BLEU, and POPE, but provides diverse subsets for evaluation. Following prior studies (Yin et al., 2023; Leng et al., 2023; Chen et al., 2024b; Zhuang et al., 2024), we evaluated RVCD on the "existence", "count", "position", and "color" subsets to comprehensively assess OH. Notably, the existence subset highlights RVCD's ability to effectively transfer the object detection capabilities of OD models to LVLMs.

LLaVA-Bench. LLaVA-Bench (Liu et al., 2023a) consists of 24 images, each paired with highly accurate and detailed human-generated descriptions. This benchmark includes three types of questions: simple question answering (conversation), detailed descriptions, and complex reasoning. To qualita-

N, P Settings	LLaVA-1.5			MiniGPT-4			mPLUG-Owl2		
	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑
gt (ann), hal (ann)	30.2±0.86	10.75±0.36	12.06±0.19	14.20±2.44	9.08±1.15	12.88±0.39	27.88±1.54	11.76±0.69	11.88±0.24
gt+hal (ann), $\emptyset$	29.04±0.72	10.05±0.26	11.78±0.21	12.96±1.58	7.87±0.67	12.51±0.30	25.52±1.82	10.33±1.02	11.60±0.29
hal (yl 3), gt (yl 3)	12.84±1.11	4.48±0.65	14.99±0.21	9.28±0.99	3.68±0.39	15.81±0.24	12.12±2.38	4.58±0.67	14.45±0.22
hal (yl 8), gt (yl 8)	12.4±0.70	3.98±0.34	15.18±0.10	8.84±0.50	3.47±0.17	15.99±0.37	10.68±2.27	3.96±0.80	14.74±0.24
hal (ann), gt (ann)	**8.44**±0.84	**2.77**±0.12	**15.61**±0.13	**6.48**±0.36	**2.48**±0.13	**16.10**±0.28	**7.96**±1.00	**2.84**±0.46	**15.02**±0.31

Table 3: Performance under CHAIR and BLEU with increasing detection rates, simulated using annotations or Object Detection models. Detection rate in order: 0%, treating all objects as N, detection with YOLOv3 (yl 3), detection with YOLOv8x (yl 8), 100%. α, β were set to 1 and 0, respectively for this experiment.

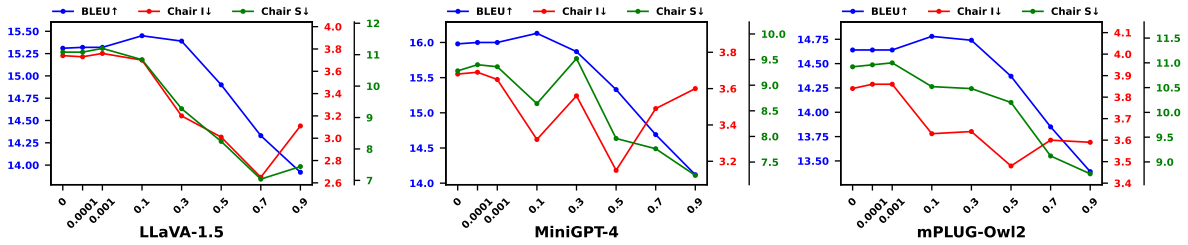

Figure 6: BLEU and CHAIR scores based on the variation of the β value, with α set to 1. The mean of five samples, each consisting of 500 instances, sampled with replacement from the MSCOCO 2014 validation dataset.

tively evaluate RVCD, we leveraged LLaVA-Bench as a case study, following the methodologies of prior studies. The detailed results are provided in Appendix H.

6 Analysis and Ablation Studies

6.1 Effect of Accurate Detection

In Table 3, the N and P settings, gt (ann), hal (ann) indicate a 0% detection rate when based on annotations. gt+hal (ann), $\emptyset$ assumes that all objects in the draft caption are treated as negative logits. hal (yl 3, 8), gt (yl 3, 8) refer to the application of object detection models YOLOv3 (Redmon and Farhadi, 2018) and YOLOv8x (Ultralytics, 2023) with confidence threshold 0.25 (default setting), and hal (ann), gt (ann) represent a scenario where the detection rate is 100% based on annotations. For a fair comparison between Setting gt+hal (ann), $\emptyset$ and other settings, α, β were set to 1 and 0, respectively for this experiment. CHAIR and BLEU scores for captioning improve with increased detection accuracy. This shows that our RVCD becomes increasingly effective as detection accuracy improves, suggesting that advancements in object detection models can have a direct positive impact on RVCD performance.

6.2 Ablation Study of α and β

The statistical details of the ablation of α and β are presented in Appendix G.1. Increasing α strengthens the regulation of hallucinated (Hal) objects, benefiting the CHAIR score. However, as observed in Section 4.4, it also unintentionally removes parts of the ground truth (GT) objects. The CHAIR gains from Hal object removal outweigh the CHAIR losses caused by GT object degradation, ultimately resulting in a net benefit as α increases. Nevertheless, the gradual decline in BLEU indicates that the issue of GT object degradation persists (Table 11).

To address this, we introduced β and positive logits at a level that restores the degraded GT objects, as represented in Equation 6. α maximizes the benefits of Hal object regulation at $\alpha = 1$, while β maximizes GT recovery at $\beta = 0.1$. As shown in Figure 6, at $\beta = 0.1$, the recovery of the degraded GT objects leads to gains across $CHAIR_S$, $CHAIR_I$, and BLEU. Therefore, we propose (α, β) = (1, 0.1) as the optimal setting of RVCD.

6.3 Decoding Latency Analysis on Image Captioning

Different state-of-the-art (SOTA) decoding methods vary in the frequency and timing of external model calls, depending on their underlying methodologies. Furthermore, the computational cost can differ across specific decoding steps. As such, empirically measuring token generation latency is a reasonable and appropriate approach.

To this end, we evaluate decoding efficiency by sampling 500 images with replacement from the MSCOCO 2014 validation set, repeating the pro-

cess five times and reporting the mean and standard deviation across runs. For details on the decoding configurations, refer to Appendix G.

RVCD operates by removing hallucinated objects N from the greedy decoded caption and reintroducing ground-truth objects P. As demonstrated in our ablation study (Table 3, 4th row), even a lightweight adjustment—removing only the hallucinated objects N without reintroducing ground-truth objects P (i.e., $\beta = 0$)—achieves superior OH reduction compared to other state-of-the-art decoding methods. Introducing P ($\beta = 0.1$), as in the full RVCD pipeline, can further enhance performance (Figure 6).

RVCD exhibits significantly lower decoding latency than previous state-of-the-art methods such as HALC (Chen et al., 2024b) and OPERA (Huang et al., 2023). Despite incorporating the generation of a greedy decoded draft, comparison with objects detected via YOLO, and contrastive decoding processes involving multiple explicit images, RVCD achieves superior performance in both output quality and decoding efficiency.

Methods	Avg. Latency (s/token)	Relative
Greedy	0.034 ± 0.002	1.000×
DoLa	0.048 ± 0.001	1.416×
VCD	0.073 ± 0.002	2.174×
OPERA	0.341 ± 0.004	10.128×
HALC	0.800 ± 0.017	23.795×
RVCD ($\beta = 0$)	0.143 ± 0.003	4.242×
RVCD ($\beta \neq 0$)	0.204 ± 0.004	6.053×

Table 4: Decoding Latency per Token.

7 Conclusion

We propose RVCD, an advanced train-free decoding-based plug-and-play method that significantly alleviates the Object Hallucination (OH) problem of LVLMs. Inspired by prior studies leveraging the idea of Visual Contrastive Decoding (VCD), RVCD maximizes the potential of VCD through negative and positive logits generated from explicit images retrieved from a single-concept image database. Comprehensive experiments demonstrate that RVCD significantly outperforms other state-of-the-art (SOTA) decoding methods.

8 Limitations

RVCD adjusts the final output by leveraging additional information from multiple negative and positive logits generated from an image that explicitly represents a single concept. This approach requires generating logits for the number of words in the evaluation dictionary that match the set of objects mentioned in the greedy decoded draft caption, excluding duplicates, at each decoding step. Consequently, if the draft caption mentions an overly diverse and large number of words, it may result in a disadvantage in token latency. In future work, we will focus on constructing a more efficient reference image database to mitigate this issue.

Ethics Statement

The purpose of this study is to introduce a decoding method to mitigate Object Hallucination (OH) in Large Vision-Language Models (LVLMs). In this process, we utilized transformer-based LVLMs and a diffusion-based image generation model. While these models, as generative AI, may produce uncontrollable or disloyal outputs. Nevertheless, the goal of RVCD is to mitigate such disloyal outputs and OH, aligning with ethical review standards. Additionally, all our experiments were conducted using public datasets, and every image in the AI-generated image database we provide was manually checked and contains general object representations that do not pose ethical concerns.

9 Acknowledgements

This research was supported by the Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2024 (Project Name: Developing a generative AI story platform for Fanfiction, Project Number: RS-2024-00442270). This research was partly supported by an IITP grant funded by the Korean Government (MSIT) (No. RS-2020-II201361, Artificial Intelligence Graduate School Program (Yonsei University)).

References

Black Forest Labs. 2024. FLUX.1 License. Accessed on February 7, 2025.

Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny. 2023. Minigpt-v2: Large language model as a unified interface for vision-language multi-task learning. *arXiv:2310.09478*.

Xuweiyi Chen, Ziqiao Ma, Xuejun Zhang, Sihan Xu, Shengyi Qian, Jianing Yang, David F. Fouhey, and

Joyce Chai. 2024a. Multi-object hallucination in vision-language models. In *Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS)*. Accepted to NeurIPS 2024.

Zhaorun Chen, Zhuokai Zhao, Hongyin Luo, Huaxiu Yao, Bo Li, and Jiawei Zhou. 2024b. HALC: Object hallucination reduction via adaptive focal-contrast decoding. *arXiv:2403.00425*.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. 2023. Dola: Decoding by contrasting layers improves factuality in large language models. *arXiv:2309.03883*.

Chenhang Cui, Yiyang Zhou, Xinyu Yang, Shirley Wu, Linjun Zhang, James Zou, and Huaxiu Yao. 2023. Holistic analysis of hallucination in GPT-4V (ision): Bias and interference challenges. *arXiv:2311.03287*.

Wenliang Dai, Zihan Liu, Ziwei Ji, Dan Su, and Pascale Fung. 2022. Plausible may not be faithful: Probing object hallucination in vision-language pre-training. *arXiv:2210.07688*.

Alessandro Favero, Luca Zancato, Matthew Trager, Siddharth Choudhary, Pramuditha Perera, Alessandro Achille, Ashwin Swaminathan, and Stefano Soatto. 2024. Multi-modal hallucination control by visual information grounding. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*. ArXiv:2403.14003 [cs.CV].

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. 2023. MME: A comprehensive evaluation benchmark for multimodal large language models. *arXiv:2306.13394*.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. 2023. Hallusionbench: An advanced diagnostic suite for entangled language hallucination & visual illusion in large vision-language models. *arXiv e-prints*, pages arXiv–2310.

Anisha Gunjal, Jihan Yin, and Erhan Bas. 2023. Detecting and preventing hallucinations in large vision language models. *arXiv:2308.06394*.

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

Liqiang Jing and Xinya Du. 2024. FGAIF: Aligning Large Vision-Language Models with Fine-grained AI Feedback. *arXiv:2404.05046*. Submitted on 7 Apr 2024 (v1), last revised 6 May 2025 (v2).

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

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv:1908.03557*.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Evaluating object hallucination in large vision-language models. *arXiv:2305.10355*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014*, pages 740–755. Springer.

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

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. *arXiv:2304.08485*.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. 2023c. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. *arXiv:2303.05499*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2015. You Only Look Once: Unified, Real-Time Object Detection. *arXiv:1506.02640*.

Joseph Redmon and Ali Farhadi. 2018. YOLOv3: An Incremental Improvement. *arXiv:1804.02767*. Tech Report.

Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. 2024. Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks. *arXiv:2401.14159*.

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

Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. 2023. Aligning Large Multimodal Models with Factually Augmented RLHF. *arXiv:2309.14525*. Preprint, submitted on 25 Sep 2023.

Haoqin Tu, Chenhang Cui, Zijun Wang, Yiyang Zhou, Bingchen Zhao, Junlin Han, Wangchunshu Zhou, Huaxiu Yao, and Cihang Xie. 2023. How many unicorns are in this image? A safety evaluation benchmark for vision LLMS. *arXiv:2311.16101*.

Ultralytics. 2023. Yolo by ultralytics. Accessed on January 24, 2025.

Ultralytics. 2024. Ultralytics yolov8 turns one: A year of breakthroughs and innovations. Accessed on January 24, 2025.

Xiyao Wang, Yuhang Zhou, Xiaoyu Liu, Hongjin Lu, Yuancheng Xu, Feihong He, Jaehong Yoon, Taixi Lu, Gedas Bertasius, Mohit Bansal, Huaxiu Yao, and Furong Huang. 2024. Mementos: A comprehensive benchmark for multimodal large language model reasoning over image sequences. *arXiv:2401.10529*.

Yun Xing, Yiheng Li, Ivan Laptev, and Shijian Lu. 2024. Mitigating Object Hallucination via Concentric Causal Attention. *arXiv:2410.15926*. To appear at NeurIPS 2024, submitted on 21 Oct 2024.

Chenglin Yang, Celong Liu, Xueqing Deng, Dongwon Kim, Xing Mei, Xiaohui Shen, and Liang-Chieh Chen. 2024a. 1.58-bit FLUX: The first successful approach to quantizing the state-of-the-art text-to-image generation model, FLUX.1-dev, using 1.58-bit weights. *arXiv:2412.18653*.

Dingchen Yang, Bowen Cao, Guang Chen, and Changjun Jiang. 2024b. Pensieve: Retrospect-then-Compare Mitigates Visual Hallucination. *arXiv:2403.14401*. Submitted on 21 Mar 2024 (v1), last revised 1 Sep 2024 (v2).

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. 2023. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. *arXiv:2311.04257*.

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

Bohan Zhai, Shijia Yang, Xiangchen Zhao, Chenfeng Xu, Sheng Shen, Dongdi Zhao, Kurt Keutzer, Manling Li, Tan Yan, and Xiangjun Fan. 2023. Halle-switch: Controlling object hallucination in large vision language models. *arXiv e-prints*, pages arXiv–2310.

Yiming Zhang, Zhuokai Zhao, Zhaorun Chen, Zhili Feng, Zenghui Ding, and Yining Sun. 2024. Rankclip: Ranking-consistent language-image pretraining. *arXiv:2404.09387*.

Yiyang Zhou, Chenhang Cui, Rafael Rafailov, Chelsea Finn, and Huaxiu Yao. 2024a. Aligning modalities in vision large language models via preference fine-tuning. *arXiv:2402.11411*.

Yiyang Zhou, Chenhang Cui, Jaehong Yoon, Linjun Zhang, Zhun Deng, Chelsea Finn, Mohit Bansal, and Huaxiu Yao. 2023. Analyzing and mitigating object hallucination in large vision-language models. *arXiv:2310.00754*.

Yiyang Zhou, Zhiyuan Fan, Dongjie Cheng, Sihan Yang, Zhaorun Chen, Chenhang Cui, Xiyao Wang, Yun Li, Linjun Zhang, and Huaxiu Yao. 2024b. Calibrated self-rewarding vision language models. *arXiv:2405.14622*.

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

Xianwei Zhuang, Zhihong Zhu, Zhanpeng Chen, Yuxin Xie, Liming Liang, , and Yuexian Zou. 2024. Game on tree: Visual hallucination mitigation via coarse-to-fine view tree and game theory. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17984–18003, Miami, Florida, USA. Association for Computational Linguistics.

A Data License and Usage

Our experiment was conducted using the MSCOCO validation 2014 dataset (Lin et al., 2014) for CHAIR, BLEU, and POPE. Additionally, we evaluated the quality of image captioning both quantitatively and qualitatively through MME (Fu et al., 2023) and LLAVA-bench (Liu et al., 2023a). We clarify that all of these datasets are publicly available for research purposes and were utilized to assess the image captioning performance of various decoding-based methods. Additionally, our generated single concept AI images were created through the image generation model FLUX.1-dev, which is under Non-Commercial License (Black Forest Labs, 2024).

B Computational Resources

We adopted widely used 7B-sized LVLM backbones in our experimental environment, including MiniGPT-4 V2 with Vicuna-7B (Chen et al., 2023), LLaVA-1.5 (Liu et al., 2023b), and mPLUG-Owl2 (Ye et al., 2023). All experiments were conducted without any training, performing only inference, and were executed on a single NVIDIA A100 GPU.

C Detection Precision on Draft Captions

As shown in Figure 1, the precision of each LVLM when answering a VQA question with "yes" or "no" about the presence of an object mentioned in its generated caption is expressed as GT/Hal (LVLM). Similarly, when treating objects detected/not detected by YOLO as equivalent to the LVLM's "yes" or "no" answers, YOLO's precision is represented as GT/Hal (YOLO). Hal (LVLM) is noticeably worse, indicating that errors occurred in most cases where LVLMs gave negative answers in the VQA task. In other words, there were many False Negatives where the objects actually existed, but the LVLMs stated that they did not. This suggests that the object detection capability of LVLMs is insufficient to self-correct hallucinated objects in the draft captions.

Table 5 represents the statistical details of Figure 1.

Table 5: Statistical details of detection precision on draft captions. The mean and standard deviation of five samples, each consisting of 500 instances, sampled with replacement from the MSCOCO 2014 validation dataset.

Precision	**LLaVA-1.5**	**MiniGPT-4**	**mPLUG-Owl2**
Hal (YOLO)	90.05±0.85	91.75±4.39	91.23±2.95
Hal (LVLM)	28.77±3.01	57.49±5.24	18.10±2.57
GT (YOLO)	91.12±0.28	91.23±0.78	91.94±0.47
GT (LVLM)	98.21±0.38	89.08±1.35	99.05±0.37

D Comprehensive POPE Results

In the POPE evaluation, to create a fair environment similar to previous studies (Chen et al., 2024b; Zhuang et al., 2024), we combined the entire query of POPE with an initially greedy decoded answer (yes/no) and used it as a draft caption for RVCD. Accordingly, the detector determines whether the object mentioned in the draft caption actually exists in the image and conveys this judgment to the LVLMs. This is similar to how HALC (Chen et al., 2024b) combines the entire query with an initial answer (yes/no) to form a text prompt, allowing the detection model to provide grounding for the focal area of the query in their POPE evaluation. The statistical details are presented in Table 6 and Table 7.

E MME Experiment Details

For reliability, in MME evaluations, instead of using the offline approach employed in previous studies (Chen et al., 2024b; Zhuang et al., 2024), we utilized a query concatenating the prompt "Please describe this image and then answer the question. " with the original questions from MME to enable automated evaluation. Subsequently, we assessed whether positive/negative words were present in the output captions. Detailed information is provided in Table 8.

Table 8: Comparison of Decoding Methods Performances on MME Sub-tasks: Existence, Color, Count, Position.

Model	Decoder	Existence	Color	Count	Position	Tokens	Samples
LLaVA-1.5	RVCD	123.33	95.0	60.0	65.0	128	120
	Greedy	105.0	65.0	60.0	50.0	128	120
	VCD	100.0	85.0	56.66	60.0	128	120
	Opera	88.33	80.0	60.0	53.33	128	120
	DoLa	75.0	60.0	53.33	55.00	128	120
	HALC	73.33	65.0	58.33	53.33	128	120
MiniGPT-4	RVCD	130.0	70.0	51.66	53.33	128	120
	Greedy	108.33	78.33	48.33	56.66	128	120
	VCD	125.0	65.0	53.33	51.66	128	120
	Opera	71.66	53.33	46.66	56.66	128	120
	DoLa	128.33	78.33	60.0	60.0	128	120
	HALC	138.33	78.33	63.33	58.33	128	120
mPLUG-Owl2	RVCD	130.0	125.0	70.0	53.33	128	120
	Greedy	111.66	120.0	68.33	70.0	128	120
	VCD	126.66	91.66	81.66	55.0	128	120
	Opera	105.0	90.0	63.33	50.0	128	120
	DoLa	111.66	120.0	65.0	65.0	128	120
	HALC	98.33	115.0	65.0	53.33	128	120

F Token Probabilities of Single Concept Images

The bar graph depicted in Figure 4 visualizes the probability distribution of the first token when LVLM describes single concept images with the prompt, "What is this? Answer in one word." The probabilities are min-max scaled for better visualization. Tokens that start with an underbar represent the first token, and the subsequent remaining tokens forming the word are represented in italic with an underline. The remaining tokens forming the words in Figure 4 were inferred through next-token prediction, where the LVLM's input consists of the prompt followed by the first token of the word. Table 9 represents the original output token probabilities for the first tokens of example images depicted in Figure 4.

G Hyperparameters settings

The hyperparameter settings of RVCD are shown in Table 10. The experimental configurations for other decoding methods with evaluation of CHAIR and BLEU scores based on natural language processing were aligned with the detailed hyperparameter settings and evaluation settings specified in HALC (Chen et al., 2024b).

Unlike previous studies, we clarify that RVCD does not adopt the adaptive plausibility threshold. Contrastive decoding-based prior studies propose an adaptive plausibility threshold (Chen et al., 2024b; Leng et al., 2023; Chuang et al., 2023) to mitigate situations where their method promotes the generation of implausible tokens. However, we find that this approach does not provide significant benefits when applied to RVCD. Since our RVCD already achieves high-quality outputs with state-of-the-art performance without relying on this additional condition, we determine that incorporating it is unnecessary.

We selected YOLO as our object detection model for several key reasons. First, it is both lightweight and computationally efficient (Redmon et al., 2015). Second, because RVCD only requires identifying which objects are present in the input image, the prompt understanding of Grounding DINO (Liu et al., 2023c) or the precise segmentation capabilities of Grounded-SAM (Ren et al., 2024) are not essential, each utilized in respective previous decoding-based studies (Chen et al., 2024b; Zhuang et al., 2024). Third, to investigate the influence of detection model improvements on RVCD performance, we needed multiple versions of the same model. YOLO has been a cornerstone in the deep learning community for nearly a decade, offering a wide range of open-source releases that made it particularly well-suited to our study. For state-of-the-art experiments, we employed YOLOv8x (Ultralytics, 2023), while for the detector ablation study, we additionally used YOLOv3 (Redmon and Farhadi, 2018). In both cases, the confidence threshold was set to the default value of 0.25.

Table 10: RVCD Default Hyperparameter Settings

Parameters	Value
Negative Logits Regulation Factor α	1
Positive Logits Recovery Factor β	0.1
Object Detection Model, Confidence Threshold	YOLOv8x (Ultralytics, 2023), 0.25

G.1 α and β ablation study detail

For statistical details, refer to Table 11 and Table 12. As the β value increases from 0, the CHAIR_I and CHAIR_S scores improve, whereas the BLEU score peaks at 0.1 and then starts to deteriorate. This indicates that the influence of positive logits should not be excessive. Therefore, we set α and β to 1 and 0.1, respectively. We propose this as the default setting for RVCD.

H Experiment Results on LLaVA-Bench.

We utilize LLaVA-Bench (Liu et al., 2023a) for qualitative case studies (Figure 7, Figure 8, Figure 9, Figure 10). Captions generated by RVCD and other decoding methods designed to mitigate OH are presented, with red fonts indicating occurrences of OH and highlighting cases where hallucinations occurred in object existence, attributes, or relationships. The identical examples of HALC and Greedy in Figure 9 are actual decoded result.

Table 6: Comparison of the mean of five POPE results on MSCOCO dataset with different decoding baselines under the 'random' and 'popular' settings. Higher accuracy (Acc.), precision (Prec.), and F_1 score indicate better performance.

Setting	Model	Decoding	Accuracy	Precision	Recall	F_1 Score
Random	LLaVA-1.5	Greedy	79.93±0.74	72.37±0.74	96.82±0.37	82.83±0.55
		Beam Search	83.65±0.29	78.05±0.36	93.64±0.52	85.13±0.26
		DoLa	80.22±0.76	72.69±0.77	96.81±0.34	83.03±0.57
		OPERA	79.35±0.88	72.24±0.86	95.36±0.54	82.20±0.68
		VCD	74.28±0.41	67.30±0.35	94.45±0.21	78.60±0.30
		HALC	80.22±0.76	72.69±0.77	96.81±0.34	83.03±0.57
		RVCD	91.33±0.34	95.16±0.48	87.09±0.82	90.94±0.39
	MiniGPT-4	Greedy	75.70±0.41	68.63±0.44	94.68±0.37	79.58±0.27
		Beam Search	77.58±0.92	71.97±1.06	90.40±0.31	80.13±0.65
		DoLa	77.55±0.65	84.84±0.96	67.10±1.58	74.92±0.92
		OPERA	76.49±0.69	70.53±0.80	91.02±0.41	79.47±0.46
		VCD	66.25±0.76	63.03±0.70	78.61±0.90	69.96±0.62
		HALC	77.55±0.65	84.84±0.96	67.10±1.58	74.92±0.92
		RVCD	87.96±1.01	91.84±2.25	83.37±0.44	87.38±0.91
	mPLUG-Owl2	Greedy	81.68±0.95	74.33±0.99	96.80±0.37	84.08±0.72
		Beam Search	86.22±0.47	81.65±0.61	93.42±0.25	87.14±0.40
		DoLa	81.82±0.87	74.53±0.94	96.72±0.32	84.18±0.66
		OPERA	85.88±0.64	80.70±0.75	94.34±0.52	86.99±0.56
		VCD	78.45±0.61	72.10±0.71	92.80±0.51	81.15±0.44
		HALC	81.81±0.87	74.51±0.95	96.72±0.28	84.17±0.65
		RVCD	89.05±0.48	90.75±0.83	86.97±0.26	88.82±0.45
Popular	LLaVA-1.5	Greedy	70.72±1.47	63.63±1.20	96.82±0.37	76.78±0.92
		Beam Search	77.94±1.06	71.27±1.12	93.64±0.52	80.93±0.75
		DoLa	71.03±1.44	63.89±1.19	96.81±0.34	76.97±0.91
		OPERA	73.96±1.39	66.79±1.25	95.36±0.54	78.55±0.95
		VCD	68.98±1.03	62.59±0.85	94.41±0.42	75.27±0.62
		HALC	71.03±1.44	63.89±1.19	96.81±0.34	76.97±0.91
		RVCD	88.94±0.68	90.43±0.76	87.09±0.82	88.73±0.69
	MiniGPT-4	Greedy	56.5±1.35	53.69±0.84	94.68±0.37	68.52±0.65
		Beam Search	63.24±1.31	58.59±0.98	90.40±0.31	71.09±0.77
		DoLa	70.23±0.61	71.62±1.42	67.10±1.58	69.27±0.51
		OPERA	64.03±0.99	59.12±0.77	91.02±0.41	71.68±0.55
		VCD	59.80±0.99	57.13±0.81	78.61±0.90	66.17±0.68
		HALC	70.23±0.61	71.62±1.42	67.10±1.58	69.27±0.51
		RVCD	86.87±0.55	89.65±1.03	83.37±0.44	86.39±0.51
	mPLUG-Owl2	Greedy	73.2±1.15	65.77±1.04	96.8±0.37	78.32±0.73
		Beam Search	80.02±0.70	73.67±0.83	93.42±0.25	82.38±0.51
		DoLa	73.42±1.13	65.99±1.04	96.72±0.32	78.45±0.72
		OPERA	78.40±0.64	71.54±0.73	94.34±0.52	81.37±0.45
		VCD	72.87±0.44	66.43±0.53	92.49±0.56	77.32±0.22
		HALC	73.42±1.12	65.99±1.04	96.72±0.28	78.45±0.71
		RVCD	87.82±0.91	88.50±1.64	86.97±0.26	87.72±0.80

Table 7: Comparison of the mean of five POPE results on MSCOCO dataset with different decoding baselines under the 'adversarial' settings. Higher accuracy (Acc.), precision (Prec.), and F_1 score indicate better performance.

Setting	Model	Decoding	Accuracy	Precision	Recall	F_1 Score
Adversarial	LLaVA-1.5	Greedy	65.92±0.88	59.84±0.65	96.82±0.37	73.97±0.51
		Beam Search	73.23±0.90	66.50±0.77	93.64±0.52	77.77±0.66
		DoLa	66.20±0.93	60.05±0.69	96.81±0.34	74.12±0.54
		OPERA	69.98±1.00	63.26±0.78	95.36±0.54	76.06±0.67
		VCD	66.31±0.27	60.42±0.18	94.56±0.37	73.73±0.21
		HALC	66.20±0.93	60.05±0.69	96.81±0.34	74.12±0.54
		RVCD	85.36±0.61	84.17±0.73	87.09±0.82	85.61±0.60
	MiniGPT-4	Greedy	56.72±0.76	53.82±0.48	94.68±0.37	68.63±0.31
		Beam Search	61.70±1.33	57.45±0.97	90.40±0.31	70.25±0.72
		DoLa	69.00±0.89	69.76±1.20	67.10±1.58	68.39±0.95
		OPERA	61.95±1.69	57.57±1.24	91.02±0.41	70.53±0.91
		VCD	59.32±1.15	56.73±0.91	78.61±0.90	65.90±0.76
		HALC	69.00±0.89	69.76±1.20	67.10±1.58	68.39±0.95
		RVCD	83.06±1.36	82.92±2.43	83.37±0.44	83.13±1.10
	mPLUG-Owl2	Greedy	68.20±1.78	61.60±1.38	96.80±0.37	75.28±1.05
		Beam Search	74.28±0.77	67.56±0.79	93.42±0.25	78.41±0.49
		DoLa	68.42±1.77	61.78±1.38	96.72±0.32	75.39±1.04
		OPERA	72.75±1.06	65.90±0.95	94.34±0.52	77.59±0.70
		VCD	69.65±0.76	63.46±0.64	92.66±0.21	75.33±0.47
		HALC	68.40±1.75	61.77±1.37	96.72±0.28	75.38±1.03
		RVCD	85.48±0.41	84.46±0.64	86.97±0.26	85.70±0.36

Image	**Token 1**	**Token 2**	**Token 3**	**Token 4**	**Token 5**
Toilet	0.9600 (**_To***ilet*)	0.0073 (**_Bath**)	0.0067 (**_L***id*)	0.0034 (**_Bowl**)	0.0020 (**_Pot***ty*)
Cat	0.9692 (**_Cat**)	0.0073 (**_K***itten*)	0.0024 (**_Dog**)	0.0017 (**_T***iger*)	0.0015 (**_St***riped*)
Fork	0.6636 (**_F***ork*)	0.0971 (**_Kn***ife*)	0.0745 (**_Ut***ensil*)	0.0272 (**_Spo***on*)	0.0231 (**_Table**)
Spoon	0.9463 (**_Spo***on*)	0.0100 (**_S***poon*)	0.0063 (**_Sp***oon*)	0.0062 (**_Silver**)	0.0056 (**_Bowl**)
Keyboard	0.9419 (**_Key***board*)	0.0349 (**_Computer**)	0.0031 (**_Ke***yboard*)	0.0018 (**_Mouse**)	0.0008 (**_keyboard**)

Table 9: Original output probabilities mapped to first tokens before min-max scaling. Token 1∼5 represents top-5 first tokens of the given single concept image. Bolded parts represent the first tokens, and the subsequent remaining tokens forming the word are represented in italic with an underline.

α	**LLaVA-1.5**			**MiniGPT-4**			**mPLUG-Owl2**		
	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑
0.25	12.20±2.00	3.91±0.50	16.04±0.18	13.28±2.21	4.82±1.00	16.17±0.27	13.48±1.62	4.85±0.38	15.33±0.20
0.5	11.52±0.86	3.66±0.19	15.78±0.21	11.60±1.21	4.21±0.38	16.06±0.21	11.68±1.02	3.89±0.27	15.13±0.26
0.75	11.12±0.39	3.66±0.19	15.48±0.13	9.76±1.05	3.60±0.32	16.19±0.16	11.44±0.50	4.07±0.25	14.90±0.19
1.0	11.08±1.15	3.74±0.33	15.31±0.13	9.28±0.64	3.68±0.32	15.98±0.33	10.92±1.83	3.84±0.39	14.64±0.22

Table 11: Ablation study on the α settings for $CHAIR_S$, $CHAIR_I$, and BLEU metrics. The mean and standard deviation of five samples, each consisting of 500 instances, sampled with replacement from the MSCOCO 2014 validation dataset.

β	LLaVA-1.5			MiniGPT-4			mPLUG-Owl2		
	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑	$CHAIR_S \downarrow$	$CHAIR_I \downarrow$	BLEU ↑
0	11.08±1.15	3.74±0.33	15.31±0.13	9.28±0.64	3.68±0.32	15.98±0.33	10.92±1.83	3.84±0.39	14.64±0.22
0.0001	11.08±1.14	3.73±0.37	15.32±0.13	9.40±0.65	3.69±0.32	16.00±0.31	10.96±1.80	3.86±0.39	14.64±0.21
0.001	11.20±1.09	3.76±0.32	15.32±0.13	9.36±0.71	3.65±0.30	16.00±0.31	11.00±1.81	3.86±0.45	14.64±0.20
0.1	10.84±0.89	3.70±0.41	15.45±0.16	8.64±0.95	3.32±0.36	16.13±0.14	10.52±1.55	3.63±0.34	14.78±0.21
0.3	9.28±0.76	3.20±0.34	15.39±0.21	9.52±1.19	3.56±0.32	15.87±0.20	10.48±1.51	3.64±0.35	14.74±0.21
0.5	8.24±0.83	3.01±0.40	14.90±0.20	7.96±0.59	3.15±0.32	15.33±0.05	10.20±0.66	3.48±0.41	14.37±0.18
0.7	7.04±1.16	2.65±0.53	14.33±0.24	7.76±0.67	3.49±0.44	14.69±0.19	9.12±0.91	3.60±0.45	13.85±0.17
0.9	7.44±0.89	3.11±0.37	13.92±0.21	7.24±0.85	3.60±0.59	14.12±0.24	8.76±1.51	3.59±0.46	13.39±0.26

Table 12: Ablation study on the β settings for $CHAIR_S$, $CHAIR_I$, and BLEU metrics. The mean and standard deviation of five samples, each consisting of 500 instances, sampled with replacement from the MSCOCO 2014 validation dataset.

RVCD (Ours): Various bowls of Asian food, including noodles and green vegetables, are placed on a wooden dining table. One large black bowl dominates the scene, filled with soup and noodles, including green peppers. Other bowls contain different dishes, such as sushi and greens. Several cups are scattered around the table, and chopsticks are available for use. Green tea is also present, adding to the assortment of beverages.

HALC: The image features a dining table with a variety of bowls and cups filled with different foods. There are three bowls placed on the table, with one of them containing noodles, and the other two containing different types of food. A cup is also present on the table, likely containing a beverage. In addition to the bowls and cups, **there are two spoons on the table,** one of which is placed near the bowl of noodles, **and the other is located closer to the edge of the table.**

Greedy: The image features a dining table with a variety of bowls and cups filled with different types of food. There are three bowls on the table, one of which is filled with noodles, while the other two contain different dishes. A cup is also present on the table, likely containing a beverage. In addition to the bowls and cups, **there are two spoons placed on the table,** one near the center and **the other towards the right side. A fork can be seen on the right side of the table as well.**

VCD: The image displays a table filled with an assortment of food and **drinks,** including several bowls of noodles and a variety of vegetables. Some of these dishes are placed in different bowls and cups, making for an appetizing spread. In addition to the bowls of food, there are a couple of cups, one containing a drink and the other possibly filled with sauce or soup. **Several pieces of broccoli can be seen among the other food items,** indicating the presence of various ingredients in the dishes.

Figure 7: LLaVA-Bench results comparing our RVCD and other methods with LLaVA-1.5 backbone.

RVCD (Ours): The fire animal figurine is wearing glasses, which gives it a humorous and playful appearance. The flames around its body add to the overall effect of the animal being on fire. The color red used for the animal's body and the flames gives it a fiery and intense look. The glasses add a fun and unexpected element to the design, making it stand out from other animal figurines. The overall design is visually appealing and adds a playful touch to the image.

HALC: The logo for LLaVA, Large Language and Vision Assistant, is a cartoon image of a red and orange animal with glasses on its face. The animal appears to be a camel or a giraffe, with a long neck and legs. The animal is wearing a pair of glasses, which gives it a human-like appearance. The animal's body is made of flames, which gives it a fiery appearance. The flames are orange and red, and they are arranged in a way that makes the animal look like it is on fire.

Greedy: The logo is a cartoon image of a red and orange animal with glasses on its face. The animal appears to be a camel or llama, with a long neck and legs. The animal's eyes are depicted as large, round, and black, with a white pupil and iris. **The animal's nose is depicted as a small, round, and black shape. The animal's mouth is depicted as a small, round, and black shape. The animal's ears are depicted as small, round, and black shapes.**

VCD: The logo is a cartoon image of a llama with flames coming out of its head and red goggles on its face. **The llama is standing on its hind legs and appears to be wearing a sweater.** The flames coming out of the llama's head are bright red and appear to be quite large, with some flames reaching down to its feet. The goggles on the llama's face are clear and appear to be in focus. The overall design of the logo is simple and eye catching, making it easy to recognize and remember.

Figure 8: LLaVA-Bench results comparing our RVCD and other methods with MiniGPT-4 backbone.

RVCD (Ours): Traffic moving on a highway at night is captured in this aerial photograph. Many cars, buses, and trucks can be seen in both directions on the highway, creating long lines of lights as they move along the road. Trees on the sides of the highway cast shadows on the traffic below at night, adding to the overall atmosphere of the scene. Numerous oncoming cars and headlights can be seen, making the highway look busy and congested. The time of day, night, is also evident from the long shadows cast by the trees on the road.

HALC: The image shows a busy highway at night with a lot of cars moving in both directions. The cars are traveling at different speeds, and some of them are braking or slowing down. **The traffic lights are red, yellow, and green, indicating that the traffic is moving in different directions.** The road is well lit, and there are streetlights on both sides of the road. The sky is dark, and **there are some stars visible in the sky.** The image is taken from a high angle, giving a bird's eye view of the highway.

Greedy: The image shows a busy highway at night with a lot of cars moving in both directions. The cars are traveling at different speeds, and some of them are braking or slowing down. **The traffic lights are red, yellow, and green, indicating that the traffic is moving in different directions.** The road is well lit, and there are streetlights on both sides of the road. The sky is dark, and **there are some stars visible in the sky.** The image is taken from a high angle, giving a bird's eye view of the highway.

VCD: The traffic on this highway is heavy, with many cars moving in both directions. The cars are traveling at different speeds, with some moving slowly and others moving quickly. The road is wide and well lit, with street lights illuminating the area. There is a lot of honking and braking, indicating that the traffic is congested. **The sky is dark and the stars are visible,** indicating that it is nighttime. Overall, the image is a representation of heavy traffic on a busy highway at night.

Figure 9: LLaVA-Bench results comparing our RVCD and other methods with MiniGPT-4 backbone.

RVCD (Ours): The Space Needle, a futuristic observation tower, stands tall above Seattle against a blue sky. It is one of the city's most iconic landmarks and offers visitors a panoramic view of the surrounding area. Several antennas are located around the base of the tower, likely used for communication purposes. Trees can be seen in the background, adding a touch of greenery to the cityscape. Buildings are also visible in the distance, emphasizing the Space Needle's prominent position within the urban landscape.

HALC: The image features a large white and green space needle, a famous landmark in Seattle, Washington. The needle is situated at the top of a hill, towering over the city. The structure is surrounded by trees, giving it a picturesque setting. In the background, there are several buildings of varying heights, including a tall building to the left of the space needle and another one to the right. A few smaller buildings can also be seen in the distance. A few people are scattered around the area, with one person standing near the left side of the image.

Greedy: The image features a large white and white-striped structure, possibly a space needle, located in a city. The space needle is situated at the top of a hill, with a clear blue sky in the background. The structure is surrounded by a few trees, adding a touch of greenery to the urban landscape. In the vicinity, there are several cars parked or driving around, with one car located near the left side of the image, another car on the right side, and a third car further back on the right side. Additionally, there are two people visible in the scene.

VCD: The image features a tall, white Space Needle, located in a city with a clear blue sky. The Space Needle is an iconic structure that stands out prominently in the cityscape. Around the Space Needle, there are several parking structures with cars parked within them. Some cars are parked closer to the foreground, while others are situated further back in the scene, creating a sense of depth in the image. The cityscape also includes trees, buildings, and other smaller structures, contributing to the overall urban environment.

Figure 10: LLaVA-Bench results comparing our RVCD and other methods with mPLUG-Owl2 backbone.