Controlling Multimodal LLMs via Reward-guided Decoding

Oscar Mañas[1,2,4], Pierluca D'Oro[1,2,4], Koustuv Sinha[4],
Adriana Romero-Soriano[1,3,4,5], Michal Drozdzal[4], Aishwarya Agrawal[1,2,5]

[1]Mila - Quebec AI Institute, [2]Université de Montréal, [3]McGill University,
[4]Meta FAIR, [5]Canada CIFAR AI Chair

oscar.manas@mila.quebec

Abstract

As Multimodal Large Language Models (MLLMs) gain widespread applicability, it is becoming increasingly desirable to adapt them for diverse user needs. In this paper, we study the adaptation of MLLMs through controlled decoding. To achieve this, we introduce the first method for reward-guided decoding of MLLMs and demonstrate its application in improving their visual grounding. Our method involves building reward models for visual grounding and using them to guide the MLLM's decoding process. Concretely, we build two separate reward models to independently control the degree of object precision and recall in the model's output. Our approach enables on-the-fly controllability of an MLLM's inference process in two ways: first, by giving control over the relative importance of each reward function during decoding, allowing a user to dynamically trade off object precision for recall in image captioning tasks; second, by giving control over the breadth of the search during decoding, allowing the user to control the trade-off between the amount of test-time compute and the degree of visual grounding. We evaluate our method on standard object hallucination benchmarks, showing that it provides significant controllability over MLLM inference, while matching or surpassing the performance of existing hallucination mitigation methods.

1. Introduction

Multimodal Large Language Models (MLLMs) have shown great potential to solve a wide range of visiolinguistic tasks, while offering a general language interface to users [5, 9]. As the adoption of MLLMs increases [1, 13, 39], the demand to easily control their behavior to satisfy diverse user needs is emerging. Two needs, in particular, arise among the most important for users of MLLMs: a) control over the precision and thoroughness of their output (e.g., object recall), and b) control over the amount of compute spent to generate those outputs. For instance, a user with visual impairment using the system to understand their surroundings may want the MLLM to respond with highly precise outputs (as hallucinations might be highly undesirable), while avoiding overly high latency on limited compute (e.g., on a smartphone); instead, a user leveraging the MLLM to generate synthetic captions to train downstream models may prioritize more diverse and detailed outputs (even if it means tolerating lower precision) while having the flexibility to spend more compute.

In this paper, we tackle this problem and propose a method for inference-time alignment of MLLMs. Our method, called multimodal reward-guided decoding (MRGD), employs two reward functions, one tailored for hallucination reduction [3] and one tailored for improving object recall. Using these reward functions as criteria for searching for better outputs, our method gives control over the two axes mentioned above: by giving the option to set a relative weight for each reward, it allows to control the trade-off between object precision and recall of the MLLM's outputs; by varying the breadth of the search, we can control the trade off between the amount of test-time compute and the degree of visual grounding (which encompasses both object precision and recall).

Previous works explored reducing hallucinations in MLLMs by using methods such as prompting [50], supervised fine-tuning (SFT) [24] and RLHF fine-tuning [38, 45, 53]. However, these methods do not allow fine-grained inference-time controllability of the MLLM's behavior: prompting relies on very coarse control by means of prompt engineering, while SFT and RLHF allow no controllability at all during inference. For text-only LLMs, reward-guided decoding has been shown to be an effective way of obtaining fine-grained controllability [12, 18, 29, 37], but there is a lack of such techniques for MLLMs. Compared to the text-only case, for which a reward model processes data from a single modality, reward models guiding MLLMs face unique challenges, as they need to process both visual

and textual information at the same time. In particular, multimodal reward models need to understand the interaction between the generated text output and an input from a different modality (an image). This interaction can cause specific types of hallucinations to emerge [54] and should be addressed by tailored solutions [38].

In summary, the main contributions of our paper are:

- We propose a novel approach for reward-guided decoding for MLLMs, based on building reward models for different aspects of visual grounding and combining them to guide the search for high-quality outputs at test time.
- Through extensive experiments, we show that in MLLMs there exists an inherent trade-off between object precision and recall, as well as between compute and visual grounding quality. Our proposed method allows a user to specify a desired balance between these factors, enabling adaptive control over precision and recall, as well as between compute and visual grounding quality trade-offs, depending on task requirements and resource constraints.
- We demonstrate on standard hallucination benchmarks that our proposed method matches or outperforms existing hallucination mitigation approaches, while allowing test-time controllability of an MLLM's outputs.

2. Related work

Guided decoding of LLMs. In the text-only setting, several works have explored guiding the decoding process of LLMs with a reward model to control output features such as helpfulness and harmlessness, and summary quality. [10, 18] train a reward model to evaluate full responses, and apply it at each decoding step to evaluate response prefixes and modulate the next-token probability distribution before sampling. Instead, [12, 15, 29, 33, 43] explicitly train a scoring function to predict the expected reward of response prefixes, also known as value function. [7, 21, 22, 26, 37] explore sampling strategies such as best-of-k, beam search, or Monte Carlo tree search, which are based on generating multiple responses and selecting the best one with a reward model or value function. Unlike existing methods for LLMs, we build *multimodal* reward models to evaluate responses to *multimodal* instructions, which additionally contain images, and focus on evaluating MLLMs on visual grounding tasks. These models require processing both visual and textual information simultaneously, which can lead to different types of hallucinations that are specific to the multimodal nature of the inputs and hence need to be addressed with tailored solutions.

Mitigating hallucinations of MLLMs. Prior work on mitigating visual hallucinations of MLLMs has focused on supervised fine-tuning [24, 27], preference fine-tuning with RLHF/RLAIF [2, 36, 38, 41, 45, 46, 52, 53, 55] or prompting [50]. Reward-guided decoding can be more powerful than fine-tuning or prompting, as it directly optimizes the output, making it more likely to produce the desired results [12, 18, 29, 37]. In contrast, the principles learned during fine-tuning or specified in (system) prompts may not always be respected at generation time. In addition, reward-guided decoding can be combined with prompting or fine-tuning, and readily applied to many base models without retraining. Other methods also propose to refine the MLLM's output, either via post-hoc rectification [44, 49, 54] or specialized decoding strategies [14, 20, 40, 51]. Most similar to our method, CGD [11] uses an off-the-shelf CLIP-like model to guide decoding. However, we show that training a multimodal reward model on preference data using a stronger backbone is more effective at mitigating hallucinations. Furthermore, we propose to guide the decoding process with a combination of reward models for visual grounding, which allows the user to control the trade-off between object hallucination and recall in the MLLM's outputs.

3. Method

We propose a multimodal reward-guided decoding strategy to improve the controllability of MLLMs at inference time. We first build small yet effective multimodal reward models to evaluate different aspects of visual grounding, and later combine them for search-based guided decoding.

3.1. Building multimodal reward models

The effectiveness of our guided decoding strategy hinges on the existence of a reward function capable of successfully evaluating how well a response satisfies a certain objective. Unlike for math or coding problems [7], there are no automatic verifiers for the open-ended responses generated by MLLMs. We want to build a method that gives controllability over the outputs of an MLLM by trading off object precision and recall at inference time. To achieve so, we build two reward models (RMs), that allow to incentivize precision and recall respectively: (1) an object hallucination reward model r^{θ}_{hal} (shortened to r_{hal} when omitting the parameters is clear), trained from preference data obtained from a mixture of datasets, and (2) a recall reward model r_{rec}, obtained by combining pre-trained modules. We next describe in detail how we build these two reward models.

3.1.1. Training r_{hal} from preference data

Given a dataset of multimodal preference data for object hallucinations $D = \{x_v, x_q, y^+, y^-\}_i$, where y^+ and y^- are the chosen and rejected responses respectively, we train our reward model for object hallucination r^{θ}_{hal} as a classifier that predicts the preference probability following the

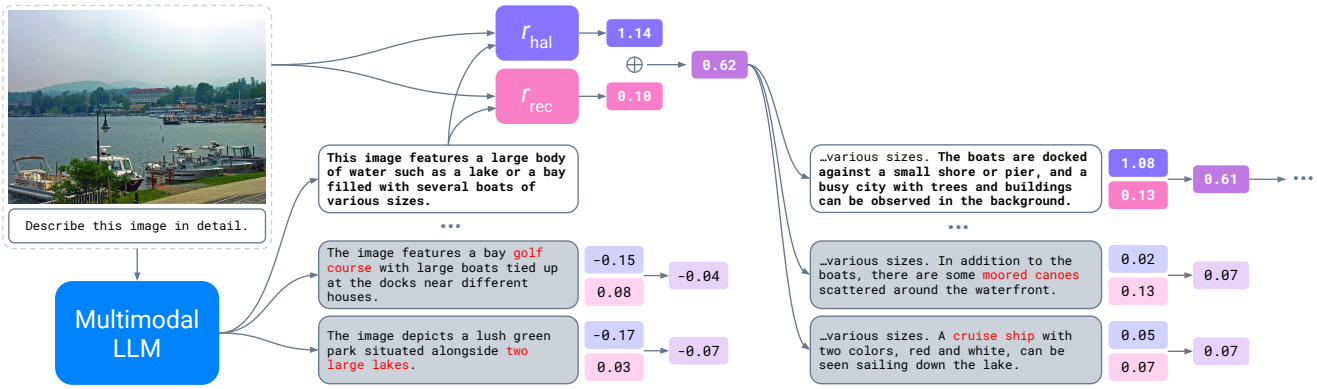

Figure 1. Illustration of multimodal reward-guided decoding (MRGD) for MLLMs. At each iteration, k candidate completions (sentences in our case) to a partial response are sampled from the MLLM and evaluated according to a linear combination of rewards (the process is illustrated for the first selected completion and omitted elsewhere). The completion with largest score is selected and added to the context to generate the next k candidates, until the <EOS> token is encountered.

Bradley-Terry model [6, 30]:

$$\mathcal{L}_{RM}(x_v, x_q, y^+, y^-; \theta) = -\log \sigma(r^\theta_{\text{hal}}(x_v, x_q, y^+) - r^\theta_{\text{hal}}(x_v, x_q, y^-)) \tag{1}$$

To facilitate combining multiple rewards, it is desirable that $r^\theta_{\text{hal}}(x_v, x_q, y) \in [0, 1]$. Therefore, we add a pair of mean-squared error loss terms to encourage $r^\theta_{\text{hal}}(x_v, x_q, y^+)$ to be close to 1 and $r^\theta_{\text{hal}}(x_v, x_q, y^-)$ to be close to 0, while simultaneously avoiding the gradient saturation pitfalls of squashing activation functions. Ultimately, this leads to the following loss function:

$$\mathcal{L}(\theta) = \mathop{\mathbb{E}}_{(x,y^+,y^-)\sim D}[\mathcal{L}_{RM}(x, y^+, y^-; \theta) + (r^\theta_{\text{hal}}(x, y^+) - 1)^2 + r^\theta_{\text{hal}}(x, y^-)^2], \tag{2}$$

where $x = (x_v, x_q)$.

We use PaliGemma [4] as the backbone of our reward model for object hallucination, and add to it a regression head consisting of a linear layer projecting the last output token embedding to a single scalar. During our initial exploration, we also considered CLIP [31] as a potential backbone for our reward model, but we ultimately discarded it due to the limited context length of CLIP's text encoder, which was insufficient to handle the longer responses present in preference data.

3.1.2. Building r_{rec} by composing off-the-shelf modules

We build our reward model for object recall r_{rec} from three off-the-shelf modules: a pre-trained object detector, a pre-trained word embedding model, and classical NLP tools. Given an image x_v and a generated caption y, we first extract the reference objects from the image with the object detector, denoted as $O_{ref} = \{o_1, o_2, ..., o_n\}$, where n is the number of detected reference objects. We also extract the predicted objects from the caption with a POS tagger, denoted as $O_{pred} = \{p_1, p_2, ..., p_m\}$, where m is the number of generated objects. We embed both reference and predicted objects with word embeddings $f_e : \mathcal{W} \to \mathbb{R}^d$, where $\mathcal{W}$ is the set of all words and d is the dimensionality of the embedding space. This results in embedded reference objects $E_{ref} = \{f_e(o_1), f_e(o_2), ..., f_e(o_n)\}$ and embedded predicted objects $E_{pred} = \{f_e(p_1), f_e(p_2), ..., f_e(p_m)\}$. Next, we compute the all-to-all semantic similarity between the embedded reference and predicted objects using a similarity function $\text{sim} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$. Specifically, for each predicted object p_i, we compute its similarity with each reference object o_j as $\text{sim}_{ij} = f_e(p_i) \cdot f_e(o_j)^T$. We consider a predicted object p_i as a true positive if its maximum semantic similarity with any reference object is above a threshold τ, i.e., $\max_{j=1,...,n} \text{sim}_{ij} > \tau$. Finally, we sum all true positives and divide by the number of reference objects to obtain the estimated object recall r_{rec}:

$$r_{\text{rec}} = \frac{\sum_{i=1}^{m} \mathbb{I}(\max_{j=1,...,n} \text{sim}_{ij} > \tau)}{n}, \tag{3}$$

where $\mathbb{I}(\cdot)$ is the indicator function.

3.2. Multimodal reward-guided decoding

Our goal is to guide the generation process of an MLLM where the generated response is modulated using the two reward functions described above. Given an image x_v and a visual instruction x_q, an MLLM π generates a text response $y = \{y_1, ..., y_n\}$ autoregressively token-by-token, i.e., $y \sim \pi(x_v, x_q)$. To give a user the possibility of choosing the relative strength of each reward model on-the-fly, we define a score s as the linear combination of the rewards for object hallucination r_{hal} and object recall r_{rec}:

$$s(x_v, x_q, y) = w \cdot r_{\text{hal}}(x_v, x_q, y) + (1 - w) \cdot r_{\text{rec}}(x_v, x_q, y), \tag{4}$$

Algorithm 1 Multimodal reward-guided decoding

$y \leftarrow$ " "
while <EOS> $\notin y$ **do**
  $Y \leftarrow \emptyset$
  for $j = 1$ to k **do**
    $y' \sim \pi(x_v, x_q, y)$
    $Y \leftarrow Y \cup \{y'\}$
  end for
  $y' = \arg\max_{y' \in Y} s(x_v, x_q, [y; y'])$
  $y \leftarrow [y; y']$
end while

where $w \in [0, 1]$ is a guidance strength hyperparameter chosen at inference time. A user can modulate the strength of the reward guidance by varying w. At the extremes, for w=1, the best response is chosen entirely by following the reward model for object hallucination, while for w=0 only the reward model for object recall is applied.

Given the score $s(x_v, x_1, y)$, we search for a response by expanding a search tree of partial responses and deciding which partial response to complete depending on the reward-based selection criterion. At each iteration, we sample k candidate completions $Y = \{y^j_{i..i+m}\}^k_{j=1}$ from a single partial response, with $(m < n)$, evaluate each of them with a reward-based score $s(x_v, x_q, y^j_{1..i+m})$, select the one with the maximum score, and add it to the context. We then iterate this process until the <EOS> token is generated (see Algorithm 1).

Since a reward model's score for a partial response also depends on how well-formed is the text of that response, evaluating a partial response at an arbitrary token can produce a lower score for a partial response that may be in reality more aligned than others (due to, e.g., incomplete words). To address this potential issue, we take advantage of the fact that captions are typically composed of multiple sentences, and evaluate the output of the MLLM every T sentences. As T grows, the reward model will evaluate longer and longer outputs. As T gets larger than the largest output length (equivalent behavior to $T = \infty$), only complete outputs concluded with an <EOS> token are evaluated, and one complete output is selected among them: this strategy is usually referred to as *rejection sampling* or *best-of-k* in the literature [7, 37]. Figure 1 provides a summary of our method.

4. Experiments

We evaluate our multimodal reward-guided decoding strategy in mitigating object hallucinations in long captions, and study the trade-offs between object precision and recall, and between visual grounding and test-time compute.

4.1. Experimental setup

Training data. We train our reward model for evaluating object hallucination on a mixture of publicly available multimodal preference datasets where responses without hallucinations are preferred over responses with hallucinations: LLaVA-RLHF [38] (9.4k), RLHF-V [45] (5.7k), POVID [53] (17k), RLAIF-V [46] (83k). In addition, we repurpose SugarCrepe [16] (7.5k) as preference data[1]. We use an 80/20% train/validation split for each dataset. To handle varying dataset sizes, each minibatch contains roughly the same amount of examples from each dataset.

Implementation details. We initialize our object hallucination reward model's backbone from PaliGemma[2], train the linear regression head from scratch and finetune the backbone with LoRA [17]. We use an effective minibatch size of 256, warm up the learning rate from 0 to $1e^{-3}$ during the first 5% of the first epoch and decay it to zero with a cosine schedule. We train the reward model for a single epoch. For the object recall reward model, we use the open-vocabulary object detector OWLv2[3] [28], the word embedding model Sentence-BERT[4] [34] and the POS tagger from the Natural Language Toolkit (NLTK). We set the object similarity threshold τ=0.8. We use LLaVA-1.5$_{7B}$[5] [25] and Llama-3.2-Vision$_{11B}$[6] [13] as our base MLLMs. We caption images with the prompt "`Describe this image in detail`" for LLaVA-1.5 and "`Describe this image in a few sentences`" for Llama-3.2-Vision. For guided decoding, unless otherwise specified, we use a sampling temperature of t=1.0 for LLaVA-1.5 and t=0.2 for Llama-3.2-Vision.

Evaluation setup. We evaluate our method on two standard object hallucination benchmarks, CHAIR [35] (5k) and AMBER [42] (1k), and report instance-level and sentence-level hallucination rates (the inverse of object precision), using the metrics used in respective benchmarks – C_i and CHAIR for instance-level, and C_s and Hal. for sentence level. We also report object recall using the Rec. and Cov. (short for coverage) metrics, and caption length (denoted by Len.) to ensure our method generates meaningful captions rather than degenerating into object-less outputs (more details in Appendix 6.1).

4.2. Reward model evaluation

We first evaluate the performance of r_{hal} on a held-out validation set from our preference data. We define accuracy as the fraction of times the reward model assigns higher scores to chosen vs. rejected responses, i.e. $r^{\theta}_{\text{hal}}(x_v, x_q, y^+) >$

[1] We use the instruction "`Describe this image`".
[2] `google/paligemma-3b-pt-224`
[3] `google/owlv2-base-patch16-ensemble`
[4] `sentence-transformers/all-mpnet-base-v2`
[5] `llava-hf/llava-1.5-7b-hf`
[6] `meta-llama/Llama-3.2-11B-Vision-Instruct`

Table 1. Results on object hallucination benchmarks for LLaVA-1.5. C_i/CHAIR: instance-level hallucination rate, C_s/Hal.: sentence-level hallucination rate, Rec./Cov.: object recall/coverage. MRGD with k=30 and T=1. BS@k: beam search with k beams, $*$: reported results from [36], †: results computed by us running the original code, ‡: results from our reimplementation (more details in Appendix 6.2), ?: the decoding strategy used is unclear from the paper. Bolded values indicate the best performance among methods in the same family.

Model	Decoding strategy	COCO				AMBER		
		C_i (↓)	C_s (↓)	Rec. (↑)	Len.	CHAIR (↓)	Hal. (↓)	Cov. (↑)
Baselines								
LLaVA-1.5$_{7B}$ [25]	Greedy	15.05	48.94	81.30	90.12	7.6	31.8	49.3
	Greedy + Prompting	13.50	44.00	80.38	92.98	6.7	29.1	49.4
	BS@30	15.68	55.00	81.62	101.89	11.2	41.4	46.1
Fine-tuning approaches								
LLaVA-RLHF$^{\dagger}_{7B}$ [38]	Greedy	16.09	57.24	81.34	119.82	10.2	48.7	53.0
HA-DPO* [52]	BS@5	11.0	38.2	-	91.0	6.7	30.9	49.8
POVID [53]	?	5.4	31.8	-	-	-	-	-
EOS* [47]	Greedy	12.3	40.2	-	79.7	5.1	22.7	49.1
HALVA$_{7B}$ [36]	?	11.7	41.4	-	92.2	6.6	32.2	53.0
CSR [55]	BS@5	7.3	28.0	-	-	-	-	-
Liu et al. [27]	?	14.5	55.0	79.2	107.5	6.5	31.7	50.9
mDPO [41]	?	9.8	35.7	-	-	4.4	24.5	52.4
Guided decoding approaches								
LLaVA-1.5$_{7B}$	VCD† [20]	15.76	54.18	81.66	102.91	9.7	42.8	51.6
	CGD‡ [11]	10.44	41.76	80.43	92.26	5.9	28.5	49.2
	MRGD$_{w=1.0}$	**6.83**	**26.38**	74.52	93.28	**5.3**	**25.8**	46.8
	MRGD$_{w=0.75}$	7.64	28.87	76.02	93.38	6.3	30.3	52.7
	MRGD$_{w=0.5}$	7.83	29.68	77.54	94.26	6.3	33.2	57.1
	MRGD$_{w=0.25}$	9.76	37.20	79.96	95.93	7.3	39.0	60.0
	MRGD$_{w=0.0}$	26.36	73.88	**85.11**	109.07	14.4	63.8	**64.0**

$r^{\theta}_{\text{hal}}(x_v, x_q, y^-)$. We obtain an average validation accuracy of 77.54%, which is in line with the performance of well-behaved reward models [19].

4.3. Comparison to baselines and existing methods

We compare MRGD with existing hallucination mitigation methods based on fine-tuning [27, 36, 41, 47, 52, 53, 55] and guided decoding [11, 20] for the LLaVA-1.5 base model. Existing methods were selected based on recency, comparability and code/checkpoint availability. We also implement a prompting baseline based on requesting the desired response properties in the input prompt (more details in Appendix 6.3). For MRGD, we choose the best performing variant w.r.t. the number of samples k, the reward evaluation period T, and the temperature t.

Table 1 shows MRGD with w=1 considerably reduces object hallucinations w.r.t. greedy decoding, at the expense of a moderate decrease in object recall/coverage. For instance, on the COCO benchmark, CHAIR$_i$ is reduced by 54.6% (from 15.05% with greedy decoding to 6.83% with MRGD) while recall is only reduced by 8.3%. By combining both reward models with w=0.5, recall is substantially increased w.r.t. w=1.0 (∼3% on COCO and ∼7% on AMBER), without overly increasing the hallucination rate (∼1% on COCO and AMBER). When w=0, MRGD achieves state-of-the-art results on object recall/coverage at the cost of a higher hallucination rate (e.g., CHAIR$_i$ is increased by 75% w.r.t. greedy decoding).

We also observe the optimal operating point w^*—mitigating object hallucinations without loosing recall—varies slightly across benchmarks, with $w^*{\approx}0.25$ for COCO and $w^* \in [0.75, 1]$ for AMBER. An analysis of 500 images from COCO and AMBER reveals that COCO images have an average of 21.4 detected objects, compared to 9.9 for AMBER, resulting in systematically lower r_{rec} values for COCO. Therefore, the optimal w assigns more weight to r_{rec} (lower w) for COCO than for AMBER.

Compared to prior visual hallucination mitigation methods, MRGD matches or surpasses the performance of methods which fine-tune the base MLLM, while offering greater flexibility and more granular control over the MLLM's behavior. For instance, MRGD outperforms LLaVA-RLHF [38], HA-DPO [52], EOS [47], HALVA [36], CSR [55], Liu et al. [27] and mDPO [41], while matching the performance of POVID [53] on COCO. On AMBER, MRGD outperforms LLaVA-RLHF,

Table 2. Results on object hallucination benchmarks for Llama-3.2-Vision. C_i/CHAIR: instance-level hallucination rate, C_s/Hal.: sentence-level hallucination rate, Rec./Cov.: object recall/coverage. MRGD with k=10 and T=1.

Model	Decoding strategy	COCO				AMBER		
		C_i (↓)	C_s (↓)	Rec. (↑)	Len.	CHAIR (↓)	Hal. (↓)	Cov. (↑)
Llama-3.2-Vision$_{11B}$ [13]	Greedy	6.09	22.44	72.69	107.34	6.1	39.6	66.7
	Greedy + Prompting	5.70	27.23	72.99	178.02	5.6	43.3	65.4
Llama-3.2-Vision$_{11B}$	MRGD$_{w=1.0}$	**5.13**	**19.50**	71.62	103.10	**5.3**	**34.2**	66.7
	MRGD$_{w=0.5}$	5.78	21.14	72.42	103.59	**5.3**	34.9	68.4
	MRGD$_{w=0.0}$	6.88	26.84	**74.67**	107.82	7.0	45.4	**71.2**

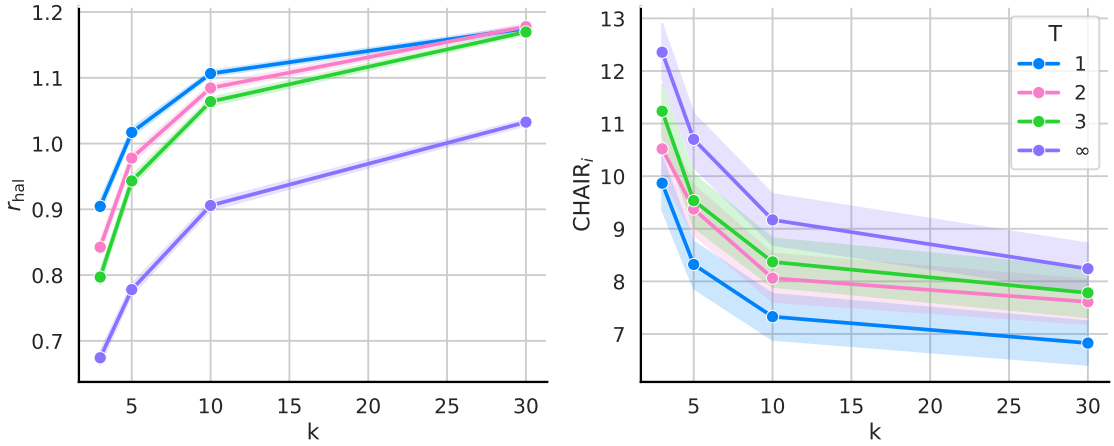

Figure 2. Reward value r_{hal} (left) and CHAIR$_i$ (right) for LLaVA-1.5 on COCO varying k and T, with w=1.0 and t=1.0. Leveraging the reward model to guide the generation more often (lower T) improves compute-efficiency.

HA-DPO, HALVA, and Liu et al., while being on par with EOS and mDPO. For guided decoding approaches, we see MRGD outperforms our reimplementation of CGD [11] in terms of hallucination mitigation. Surprisingly, LLaVA-RLHF and VCD [20] exhibit a higher hallucination rate than greedy decoding on captioning hallucination benchmarks, which were not considered in the original papers; instead, they limited their evaluation to discriminative hallucination benchmarks consisting of visual questions. This suggests that generative (captioning) and discriminative (VQA) hallucination benchmarks may not be as strongly correlated as previously assumed.

4.4. Applying MRGD on top of RLHF

While the hallucination mitigation literature focuses primarily on the instruction fine-tuned LLaVA-1.5 model, here we assess the effectiveness of our method with more recent MLLMs that have been already fine-tuned with RLHF. We apply MRGD on top of Llama-3.2-Vision, which has undergone a preference alignment phase (with DPO [32]) after instruction fine-tuning. Crucially, its multimodal preference data includes visual grounding examples [13], which makes it less prone to hallucinations.

Table 2 shows that MRGD is also effective when applied to Llama-3.2-Vision. As expected, the improvement in object precision and recall is smaller compared to LLaVA-1.5 since Llama-3.2-Vision already starts from a better level of visual grounding. However, MRGD can further mitigate object hallucinations: when guiding decoding with the reward model for object hallucinations (w=1.0), we observe a ∼1% reduction in instance-level hallucinations, a ∼3% and ∼5% decrease in sentence-level hallucinations for COCO and AMBER respectively, and only a slight decrease in object recall for COCO (∼1%), while it is maintained for AMBER.

One interesting observation is that Llama-3.2-Vision's object recall on COCO is considerably lower (-8.6%) than that of LLaVA-1.5, which may be due to more conservative outputs as a consequence of preference and safety fine-tuning. Guiding decoding with the reward model for object recall (w=0.0) boosts object recall in the generated captions by ∼1% on COCO and ∼4% on AMBER.

4.5. Visual grounding vs. compute trade-off

By varying the number of samples k and the evaluation period T, we can control the trade-off between the degree of visual grounding in the generated outputs and the amount of compute used during decoding, for fixed guidance strength

Greedy: The image features a large open market with a variety of colorful umbrellas and parasols on display. There are at least six umbrellas in the scene, with some placed closer to the foreground and others further back. The umbrellas come in different sizes and colors, creating a vibrant and lively atmosphere. In addition to the umbrellas, there are two **chairs** visible in the market, one located near the center and the other towards the right side. A person can be seen in the background, possibly browsing the market or attending to the umbrellas.

MRGD (w=1.0): The image depicts a market area with many decorative umbrellas displayed outside under a wooden shed. These colorful umbrellas are spread throughout the scene, with some further back and others more to the foreground. A few umbrellas are positioned at various angles on the left side of the image, while others are clustered on the right side of the scene. The market scene looks vibrant, and the umbrellas seem to be the main attraction.

MRGD (w=0.5): The image features a variety of colorful and patterned umbrellas arranged in a row, likely displayed for sale or decorative purposes. There are 8 prominent umbrellas visible in the scene; some are placed on the ground while others are hung or sitting upright. A group of people can be seen on the left side of the image, possibly browsing the array of umbrellas for purchase or admiring the overall visual display. Overall, the scene showcases the creative and vibrant nature of these eye-catching umbrellas.

MRGD (w=0.0): The image showcases a room with a **hardwood floor** filled with numerous Chinese umbrella or parasol items, creating a plethora of colorful and varying designs. In addition to the umbrellas, there's a dining table in the room with **bowls**, **apples**, and a **bottle** on it. A person is present near the table, perhaps admiring the collection of umbrellas or attending to the tableware. A couple of **surfboards** can be spotted on the floor, suggesting that the person occupying the space may also have an interest in water sports or recreational activities. The room also features a **potted plant**, a **chair**, and a **handbag**, further adding to the cozy and inviting atmosphere of the room. Overall, it seems like a combination of an art gallery, a Chinese restaurant, and a casual living space.

(a)

Greedy: The image features a kitchen counter with a variety of appliances and utensils. A **blender** is prominently placed on the counter, along with a food processor and a **mixer**. There are also several **knives** and a **pair of scissors** scattered across the counter. In addition to the kitchen tools, there are a few books on the counter, possibly cookbooks or reference materials. A **bowl** can be seen on the counter as well, possibly containing ingredients or leftovers. The scene gives off a busy and active atmosphere, typical of a kitchen in use.

MRGD (w=1.0): The image displays a kitchen counter laden with several items. A primary focus is a white food processor, standing on top of a dining table. In addition to the food processor, there are a **knife** and a cup also resting on the table, close to the food processor. The dining table is large, stretching long, and occupies the majority of the scene. The food processor stands prominently, creating a sense of emphasis on the appliance.

MRGD (w=0.5): The image shows a kitchen counter with a **blender** sitting on it. The **blender** appears to be broken or inoperable, as evidenced by it sitting alone on the counter. Beneath the **blender**, there are several **knives** spread out on the counter, as well as some fruits, such as an **orange** wedged into the middle. The presence of fruit, **knives**, and the **blender** indicates it might have been a moment of food preparation.

MRGD (w=0.0): The image shows a countertop with various kitchen appliances and **utensils**. A food processor is prominently displayed on the counter, surrounded by several cups, **spoons**, a **knife**, and a **pair of scissors**. There are also numerous books placed on the counter, appearing to be display items or reference material for cooking. In the background, a dining table can be seen along with an **oven** and a **microwave**, indicating that it is a kitchen space used for various cooking tasks. A zebra print item is also visible, adding a touch of style to the kitchen setting.

(b)

Figure 3. Selected qualitative examples when generating captions with LLaVA-1.5 using the default greedy decoding and our proposed MRGD strategy for different values of w (k=30, T=1). Correct objects are highlighted in green and hallucinated objects in red. New lines in captions are omitted for brevity.

w and temperature t: expanding the search space (increasing k) and evaluating more frequently (decreasing T) increases visual grounding but also the required compute.

Figure 2 shows how the reward value r_{hal} and hallucination rate (CHAIR$_i$) evolve as we increase the number of samples k=$\{3, 5, 10, 30\}$ and as we vary the evaluation period T. As expected, we observe the hallucination rate decreases as we increase k. Notably, MRGD (with T=1) is considerably more compute-efficient than naive rejection sampling (equivalent to T=∞). For example, a similar

hallucination rate is achieved with rejection sampling with k=30 and MRGD with k=5, making MRGD $\sim$6$\times$ more compute-efficient than rejection sampling. Note that evaluating more frequently also increases computational cost, but to a much lesser extent than generating, since the reward models are considerably smaller than the base MLLM and evaluation only requires a single forward pass.

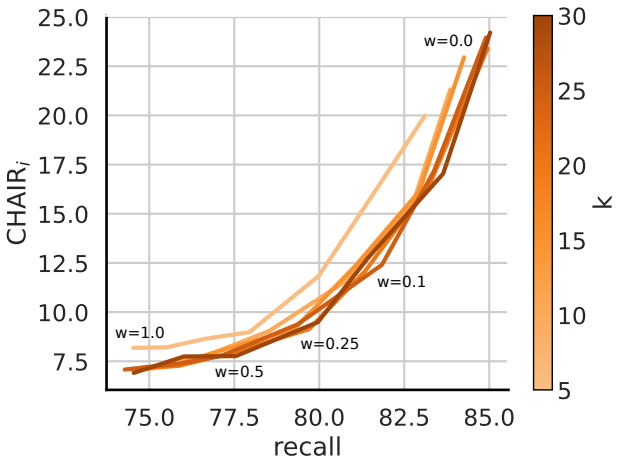

Figure 4. Object precision and recall for LLaVA-1.5 on COCO, with T=1. Each curve represents a different k, and points along a curve represent different w's. Using more compute by increasing k improves both precision and recall, while varying w modulates the precision-recall trade-off for a given amount of compute.

4.6. Object precision vs. recall trade-off

Figure 4 shows the trade-off between hallucination rate (CHAIR$_i$) and object recall when varying the guidance strength w=$\{0.0, 0.05, 0.1, 0.25, 0.5, 0.75, 1.0\}$ for fixed k and T. We observe a lower w leads to higher recall and lower precision (higher CHAIR$_i$) and vice-versa. And the trade-off curve gets closer to the ideal curve (bottom-right edges) with higher k. This suggests that there is an inherent trade-off between precision and recall in MLLMs. However, our approach gives a user the flexibility to choose the operating point (by choosing a value for w and k) that suits their needs at inference time.

4.7. Qualitative analysis

Figure 3 illustrates qualitative differences in generated captions between using the default greedy decoding and our MRGD strategy, for the same input images. Greedy decoding produces captions which contain either more or less objects than those actually present in the images. For instance, in Figure 3a, the model correctly mentions a "market" setting with "umbrellas" and a "person", but also incorrectly adds "chairs" and misses the table or the wooden structure. When guiding the decoding with MRGD and w=1.0, the generated caption does not contain any hallucinated object, but misses the people in the background and the table. When moderately increasing the strength of the reward model for object recall (w=0.5), the generated caption remains free of hallucinated objects while additionally mentioning "a group of people". For w=0.0, the generated caption achieves a higher object recall, mentioning "umbrellas", a "table" and a "person", at the expense of also containing a few hallucinated objects such as "surfboards". Similarly, in figure 3b, the caption generated with greedy decoding correctly mentions a "kitchen counter" with several "appliances", among which only the "food processor" is actually present. It also adds "knives", "scissors" and a "bowl", which are similar to the objects in the image but not completely accurate. When guiding the decoding with MRGD and w=1.0, the hallucinated objects are significantly reduced to only "knife", while maintaining a good object recall. When w=0.5, the generated caption also contains "fruits", which are visible in the background, but it incorrectly specifies "orange". Finally, for w=0.0, the generated caption achieves a higher object recall including "cups", "books" and even a "zebra print item", while also introducing additional hallucinated objects. Overall, MRGD considerably mitigates object hallucinations in image captions while enabling control over the object precision/recall trade-off at inference time.

5. Conclusion

In this paper, we presented MRGD, a reward-guided decoding method for MLLMs based on multimodal reward models for visual grounding. We built two reward models – one evaluating object precision in captions, another evaluating object recall – used in an iterative search process that evaluates candidate responses against their combined reward values. Our methodology enables on-the-fly controllability of MLLM-generated captions along two axes: controlling the object precision/recall trade-off by adjusting the weight of each reward model, and balancing test-time compute vs. visual grounding by varying the search breadth and frequency. Our method provides significant controllability over MLLM inference while matching or surpassing existing hallucination mitigation methods.

Limitations and future work. In this work, we focused on mitigating *object* hallucinations primarily due to their ease of automatic evaluation, but other important visual hallucinations exist – related to attributes, count, spatial relationships, negation, and more – which we leave for future work. In addition, we would like to continue exploring: (1) evaluating semantically complete partial outputs within sentences, (2) building reward models for semantically incomplete outputs, (3) extending MRGD to discriminative hallucination tasks (e.g., POPE [23]), and (4) gradient-based optimization instead of search-based approaches.

Acknowledgments

We thank Saba Ahmadi, Qian Yang and Shravan Nayak for providing valuable feedback on an earlier draft of this work. During this project, Aishwarya Agrawal was supported by the Canada CIFAR AI Chair award.

References

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

[2] Elmira Amirloo, Jean-Philippe Fauconnier, Christoph Roesmann, Christian Kerl, Rinu Boney, Yusu Qian, Zirui Wang, Afshin Dehghan, Yinfei Yang, Zhe Gan, et al. Understanding alignment in multimodal llms: A comprehensive study. *arXiv preprint arXiv:2407.02477*, 2024. 2

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

[4] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024. 3

[5] Florian Bordes, Richard Yuanzhe Pang, Anurag Ajay, Alexander C Li, Adrien Bardes, Suzanne Petryk, Oscar Mañas, Zhiqiu Lin, Anas Mahmoud, Bargav Jayaraman, et al. An introduction to vision-language modeling. *arXiv preprint arXiv:2405.17247*, 2024. 1

[6] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. 3

[7] Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024. 2, 4

[8] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 1

[9] Davide Caffagni, Federico Cocchi, Luca Barsellotti, Nicholas Moratelli, Sara Sarto, Lorenzo Baraldi, Marcella Cornia, and Rita Cucchiara. The (r) evolution of multimodal large language models: A survey. *arXiv preprint arXiv:2402.12451*, 2024. 1

[10] Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*. 2

[11] Ailin Deng, Zhirui Chen, and Bryan Hooi. Seeing is believing: Mitigating hallucination in large vision-language models via clip-guided decoding. *arXiv preprint arXiv:2402.15300*, 2024. 2, 5, 6, 1

[12] Haikang Deng and Colin Raffel. Reward-augmented decoding: Efficient controlled text generation with a unidirectional reward model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11781–11791, 2023. 1, 2

[13] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 1, 4, 6

[14] 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. 2

[15] Seungwook Han, Idan Shenfeld, Akash Srivastava, Yoon Kim, and Pulkit Agrawal. Value augmented sampling for language model alignment and personalization. *arXiv preprint arXiv:2405.06639*, 2024. 2

[16] Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. *Advances in neural information processing systems*, 36, 2024. 4

[17] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*. 4

[18] Maxim Khanov, Jirayu Burapacheep, and Yixuan Li. Args: Alignment as reward-guided search. In *The Twelfth International Conference on Learning Representations*. 1, 2

[19] Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. Rewardbench: Evaluating reward models for language modeling. *arXiv preprint arXiv:2403.13787*, 2024. 5

[20] 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. 2, 5, 6, 1

[21] Alexander K Lew, Tan Zhi-Xuan, Gabriel Grand, and Vikash Mansinghka. Sequential monte carlo steering of large language models using probabilistic programs. In *ICML 2023 Workshop: Sampling and Optimization in Discrete Space*. 2

[22] Bolian Li, Yifan Wang, Ananth Grama, and Ruqi Zhang. Cascade reward sampling for efficient decoding-time alignment. In *ICML 2024 Next Generation of AI Safety Workshop*. 2

[23] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305, 2023. 8

[24] 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*. 1, 2

[25] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 4, 5, 2

[26] Jiacheng Liu, Andrew Cohen, Ramakanth Pasunuru, Yejin Choi, Hannaneh Hajishirzi, and Asli Celikyilmaz. Don't throw away your value model! generating more preferable text with value-guided monte-carlo tree search decoding. In *First Conference on Language Modeling*, 2024. 2

[27] Yufang Liu, Tao Ji, Changzhi Sun, Yuanbin Wu, and Aimin Zhou. Investigating and mitigating object hallucinations in pretrained vision-language (clip) models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18288–18301, 2024. 2, 5

[28] Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *Advances in Neural Information Processing Systems*, 36, 2024. 4

[29] Sidharth Mudgal, Jong Lee, Harish Ganapathy, YaGuang Li, Tao Wang, Yanping Huang, Zhifeng Chen, Heng-Tze Cheng, Michael Collins, Trevor Strohman, et al. Controlled decoding from language models. In *Forty-first International Conference on Machine Learning*. 1, 2

[30] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022. 3

[31] 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. 3

[32] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024. 6

[33] Ahmad Rashid, Ruotian Wu, Julia Grosse, Agustinus Kristiadi, and Pascal Poupart. A critical look at tokenwise reward-guided text generation. *arXiv preprint arXiv:2406.07780*, 2024. 2

[34] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2019. 4

[35] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4035–4045, 2018. 4, 1

[36] Pritam Sarkar, Sayna Ebrahimi, Ali Etemad, Ahmad Beirami, Sercan Ö Arık, and Tomas Pfister. Mitigating object hallucination via data augmented contrastive tuning. *arXiv preprint arXiv:2405.18654*, 2024. 2, 5, 1

[37] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024. 1, 2, 4

[38] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, Yiming Yang, et al. Aligning large multimodal models with factually augmented rlhf. *arXiv preprint arXiv:2309.14525*, 2023. 1, 2, 4, 5

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

[40] David Wan, Jaemin Cho, Elias Stengel-Eskin, and Mohit Bansal. Contrastive region guidance: Improving grounding in vision-language models without training. *arXiv preprint arXiv:2403.02325*, 2024. 2

[41] Fei Wang, Wenxuan Zhou, James Y Huang, Nan Xu, Sheng Zhang, Hoifung Poon, and Muhao Chen. mdpo: Conditional preference optimization for multimodal large language models. *arXiv preprint arXiv:2406.11839*, 2024. 2, 5

[42] Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Ming Yan, Ji Zhang, and Jitao Sang. An llm-free multi-dimensional benchmark for mllms hallucination evaluation. *arXiv preprint arXiv:2311.07397*, 2023. 4, 1

[43] Kevin Yang and Dan Klein. Fudge: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535, 2021. 2

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

[45] Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, et al. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13807–13816, 2024. 1, 2, 4

[46] Tianyu Yu, Haoye Zhang, Yuan Yao, Yunkai Dang, Da Chen, Xiaoman Lu, Ganqu Cui, Taiwen He, Zhiyuan Liu, Tat-Seng Chua, et al. Rlaif-v: Aligning mllms through open-source ai feedback for super gpt-4v trustworthiness. *arXiv preprint arXiv:2405.17220*, 2024. 2, 4

[47] Zihao Yue, Liang Zhang, and Qin Jin. Less is more: Mitigating multimodal hallucination from an eos decision perspective. *arXiv preprint arXiv:2402.14545*, 2024. 5, 1

[48] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023. 1

[49] Ce Zhang, Zifu Wan, Zhehan Kan, Martin Q Ma, Simon Stepputtis, Deva Ramanan, Russ Salakhutdinov, Louis-Philippe Morency, Katia Sycara, and Yaqi Xie. Self-correcting decoding with generative feedback for mitigating hallucinations in large vision-language models. *arXiv preprint arXiv:2502.06130*, 2025. 2

[50] Zhuosheng Zhang, Aston Zhang, Mu Li, George Karypis, Alex Smola, et al. Multimodal chain-of-thought reasoning in language models. *Transactions on Machine Learning Research*. 1, 2

[51] Linxi Zhao, Yihe Deng, Weitong Zhang, and Quanquan Gu. Mitigating object hallucination in large vision-language models via classifier-free guidance. *arXiv preprint arXiv:2402.08680*, 2024. 2

[52] Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. Beyond hallucinations: Enhancing lvlms through hallucination-aware direct preference optimization. *arXiv preprint arXiv:2311.16839*, 2023. 2, 5, 1

[53] Yiyang Zhou, Chenhang Cui, Rafael Rafailov, Chelsea Finn, and Huaxiu Yao. Aligning modalities in vision large language models via preference fine-tuning. In *ICLR 2024 Workshop on Reliable and Responsible Foundation Models*, . 1, 2, 4, 5

[54] 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*, . 2

[55] Yiyang Zhou, Zhiyuan Fan, Dongjie Cheng, Sihan Yang, Zhaorun Chen, Chenhang Cui, Xiyao Wang, Yun Li, Linjun Zhang, and Huaxiu Yao. Calibrated self-rewarding vision language models. *arXiv preprint arXiv:2405.14622*, 2024. 2, 5

Controlling Multimodal LLMs via Reward-guided Decoding

Supplementary Material

6. Experiments

6.1. Details on evaluation metrics

We evaluate object precision and recall with standard metrics from the corresponding benchmarks, defined as follows.

CHAIR_i (C_i) [35], CHAIR [42]. Measure the fraction of hallucinated objects in the generated captions.

$$\text{C}_i/\text{CHAIR} = \frac{|\{\text{hallucinated objects}\}|}{|\{\text{all mentioned objects}\}|}$$

CHAIR_s (C_s) [35], Hal. [42]. Measure what fraction of generated captions include a hallucinated object.

$$\text{C}_s/\text{Hal.} = \frac{|\{\text{captions with a hallucinated object}\}|}{|\{\text{all captions}\}|}$$

Recall (Rec.), Coverage (Cov.) [42]. Measure the fraction of ground-truth objects covered in the generated captions.

$$\text{Rec.}/\text{Cov.} = \frac{|\{\text{correct objects}\}|}{|\{\text{all ground-truth objects}\}|}$$

6.2. Details on reporting results of existing methods

In Table 1, we report results for existing hallucination mitigation methods from the best source available. Unless otherwise specified, values are directly copied from the corresponding papers. For HA-DPO [52] and EOS [47], values are copied from Sarkar et al. [36] since their evaluation setup matches ours. For LLaVA-RLHF [38] and VCD [20], we compute results by generating captions with the original code and evaluating them on CHAIR [35] and AMBER [42], since the original papers do not report hallucination results on these benchmarks. For CGD [11], we reimplement their method with our codebase, since Deng et al. [11]'s evaluation setup does not match ours (they only report results on CHAIR, and on a subset of 500 examples instead of the standard 5000). We match their implementation as closely as possible: we use the same off-the-shelf SigLIP-SoViT-400m[7] [48] as scoring function, we sample outputs with t=0.2 and $top{-}k$=5, and combine SigLip's score with the output likelihood with w=0.99. The only difference we are aware of is that, in the original implementation, a set of N=3 candidates is maintained and for each one the next sentence is sampled M=3 times, so in total 9 candidates are evaluated. In our reimplementation, we maintain a single output and we sample k=9 candidate completions, which should yield similar results given the low temperature value.

[7] `google/siglip-so400m-patch14-384`

6.3. Additional baselines and comparisons with existing methods

Prompting. We propose multimodal reward-guided decoding (MRGD) as a method to control the behavior of MLLMs at inference time. A common approach to steer the behavior of LLMs at inference time is prompting [8]. Here, we apply the same idea to MLLMs as an alternative approach to control their behavior. To mitigate visual hallucinations in image captioning, we use the instruction "`{captioning instruction}. Provide an accurate and objective description, focusing on verifiable visual elements such as colors, textures, shapes, and compositions. Avoid making assumptions, inferences, or introducing information not present in the image`", where the captioning instruction is the one described in Section 4.1: "`Describe this image in detail`" for LLaVA-1.5 and "`Describe this image in a few sentences`" for Llama-3.2-Vision. We maintain greedy decoding for the prompting baselines. In Tables 1 and 2, we observe that prompting slightly reduces object hallucinations compared to greedy decoding for LLaVA-1.5, while for Llama-3.2-Vision, surprisingly, it does not help much and, in fact, it increases the sentence-level hallucination rate (CHAIR_s and Hal.). Instead, with LLaVA-1.5 on COCO, for the same level of object recall (∼80%), MRGD with w=0.25 achieves better object precision by ∼3.7% CHAIR_i and ∼6.8% CHAIR_s compared to prompting. This suggests that prompting is not a very effective strategy to steer MLLMs towards complex behaviors such as reducing visual hallucinations.

Using SigLIP for r_{hal}. CGD [11] can be viewed as a particular instance of MRGD when using off-the-shelf SigLIP as the reward model for object hallucinations and removing the combination of multiple reward models (i.e., setting w=1.0). Therefore, we also conduct an ablation of MRGD replacing PaliGemma fine-tuned on preference data (Section 3.1.1) with off-the-shelf SigLIP-SoViT-400m[7]. Due to SigLIP's limited context length of 64 tokens, we only evaluate the last generated sentence, unlike PaliGemma which receives the full prefix response (which may contain several

Table 3. Additional results on object hallucination benchmarks. MRGD with k=30 and T=1. $\text{MRGD}_{\text{PaliGemma}}$ indicates MRGD using PaliGemma fine-tuned on preference data for r_{hal}, $\text{MRGD}_{\text{SigLIP}}$ indicates MRGD using off-the-shelf SigLIP for r_{hal}, and bolded values indicate the best performance among methods in the same family.

Model	Decoding strategy	COCO				AMBER		
		C_i (↓)	C_s (↓)	Rec. (↑)	Len.	CHAIR (↓)	Hal. (↓)	Cov. (↑)
Baselines								
LLaVA-1.5_{7B} [25]	Greedy	15.05	48.94	81.30	90.12	7.6	31.8	49.3
Guided decoding approaches								
	$\text{MRGD}_{\text{PaliGemma},w=1.0}$	**6.83**	**26.38**	74.52	93.28	**5.3**	**25.8**	46.8
	$\text{MRGD}_{\text{PaliGemma},w=0.75}$	7.64	28.87	76.02	93.38	6.3	30.3	52.7
	$\text{MRGD}_{\text{PaliGemma},w=0.5}$	7.83	29.68	77.54	94.26	6.3	33.2	57.1
	$\text{MRGD}_{\text{PaliGemma},w=0.25}$	9.76	37.20	79.96	95.93	7.3	39.0	60.0
LLaVA-1.5_{7B}	$\text{MRGD}_{w=0.0}$	26.36	73.88	**85.11**	109.07	14.4	63.8	**64.0**
	$\text{MRGD}_{\text{SigLIP},w=1.0}$	7.19	28.00	73.71	92.73	6.0	30.1	48.5
	$\text{MRGD}_{\text{SigLIP},w=0.75}$	7.57	29.58	74.30	93.17	6.1	30.3	50.0
	$\text{MRGD}_{\text{SigLIP},w=0.5}$	8.17	32.88	75.96	94.93	6.3	33.3	53.4
	$\text{MRGD}_{\text{SigLIP},w=0.25}$	10.84	43.58	79.50	99.57	8.5	46.2	57.8

sentences). To ensure that the scores from multiple reward models are comparable and can be combined effectively, we normalize their ranges. In particular, since the effective range of SigLIP scores is much narrower than that of the reward model for object recall ($r_{\text{rec}} \in [0, 1]$), we linearly rescale SigLIP scores $r \in \mathbb{R}^k$ to cover the range $[0, 1]$: $r = (r - \min(r))/(\max(r) - \min(r) + \epsilon)$, where $\min$ and $\max$ are computed across the set of candidate samples Y, and ϵ is a small value to avoid division by zero (in case all candidates obtained the same score). In Table 3, we observe that when using a SigLIP-based r_{hal}, our MRGD strategy is still effective in reducing object hallucinations and enabling the user to trade off object precision and recall on-the-fly at inference time. However, SigLIP does not allow to reach the same level of object precision, and the trade-off with object recall is moderately worse. For instance, when w=1.0, $\text{MRGD}_{\text{PaliGemma}}$ achieves better object precision by $\sim$1.6% CHAIR_s and better Recall by $\sim$0.8% compared to $\text{MRGD}_{\text{SigLIP}}$.