# Mitigating Hallucination in Multimodal LLMs with Layer Contrastive Decoding

# Bingkui Tong<sup>1</sup> Jiaer Xia<sup>2</sup> Kaiyang Zhou<sup>2</sup>

<sup>1</sup>Mohamed bin Zayed University of Artificial Intelligence <sup>2</sup>Hong Kong Baptist University

## **Abstract**

Multimodal Large Language Models (MLLMs) have shown impressive perception and reasoning capabilities, yet they often suffer from hallucinations—generating outputs that are linguistically coherent but inconsistent with the context of the input image, including inaccuracies in objects, attributes, and relations. To address this challenge, we propose a simple approach called Layer Contrastive Decoding (LayerCD). Our design is motivated by the observation that shallow visual features are much more likely than deep visual features to cause an MLLM to hallucinate as they only capture biased, low-level information that is insufficient for high-level reasoning. Therefore, LayerCD aims to filter out hallucinations by contrasting the output distributions generated from visual features of different levels, specifically those from the shallow and deep layers of the vision encoder, respectively. We conduct extensive experiments on two hallucination benchmarks and show that LayerCD significantly outperforms current state-of-the-art. The code for LayerCD is available at maifoundations/LayerCD.

## 1 Introduction

LLMs have long struggled with hallucination, a phenomenon where model outputs appear plausible but are factually incorrect or fabricated Maynez et al. [2020]. Unfortunately, multimodal LLMs (MLLMs)—which incorporate an additional vision module to process images—also face this issue. In MLLMs, hallucination occurs when the model generates responses that are fluent and coherent yet misaligned with the visual input Liu et al. [2024b]. These inconsistencies often manifest as inaccuracies in identifying objects, attributes, or relationships, limiting the model's ability to accurately interpret images and posing a significant challenge for real-world deployment and practical applications.

We propose Layer Contrastive Decoding (LayerCD), a simple, inference-time method that requires no architectural changes. Our approach is motivated by the key observation that MLLMs are more prone to hallucination when conditioned on shallow versus deep visual features. By contrasting the output distributions from

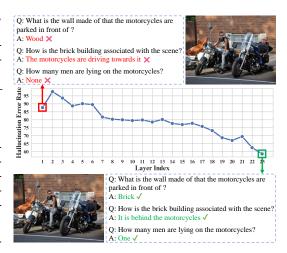


Figure 1: Evaluation of LLaVA 1.5, a state-of-theart MLLM, using different levels of visual features. The results suggest that shallow features lead to significantly higher hallucination error rates compared to deeper features.

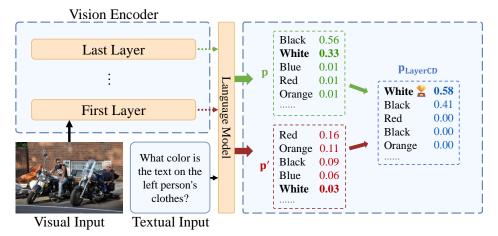


Figure 2: Overview of Layer Contrastive Decoding (LayerCD). The main idea is to factor out hallucinations by contrasting output distributions derived from different levels of visual features, specifically those from the shallow and deep layers of the vision encoder, respectively.

these two feature levels, LayerCD effectively reduces hallucinations. A preliminary study using LLaVA 1.5 Liu et al. [2024c] validates this; as shown in Fig. 1, using shallow features leads to significantly higher hallucination error rates. This is because shallow features capture only low-level characteristics like edges and colors, which are insufficient for high-level reasoning and thus more likely to cause hallucinations Zeiler and Fergus [2014], Yosinski et al. [2014].

Fig. 2 illustrates the LayerCD mechanism and the crucial differences in the resulting output distributions. While a model using deep features may produce hallucinations (e.g., "Black", Blue"), it still assigns non-trivial probability to the correct token (e.g., "White"). In contrast, conditioning on shallow features yields more high-confidence hallucinations while suppressing the correct token. Drawing inspiration from Contrastive Decoding Li et al. [2022], which was designed to reduce repetition in text, LayerCD contrasts these two distributions. This readjustment cancels out hallucinations and allows the correct token ("White") to emerge with the highest confidence.

In summary, we make the following contributions in this paper: (1) We provide an important insight that shallow visual features are more prone to causing hallucinations in MLLMs than deep visual features; (2) Based on the insight, we propose a simple contrastive decoding approach that contrasts visual features of different layers to factor out hallucinations; (3) We demonstrate the effectiveness of our approach on two hallucination benchmarks where our approach significantly outperforms current state-of-the-art.

## 2 Methodology

Our proposed Layer Contrastive Decoding (LayerCD) is an inference-time, architecture-free method to mitigate hallucination. It is motivated by the observation that contrasting the outputs from shallow visual features (high confidence on hallucinations, low on correct tokens) and deep features (high confidence on both) can isolate and suppress erroneous tokens. During decoding, an MLLM takes visual features  $\boldsymbol{z}$  and text input  $\boldsymbol{x}$  to produce the next-token probability, conditioned on previous tokens  $\boldsymbol{y}_{< t}$ :  $p(y_t|\boldsymbol{x},\boldsymbol{z},\boldsymbol{y}_{< t})$ 

### 2.1 Layer Contrastive Decoding

Contrastive Probability Distribution As discussed previously, shallow visual features are more prone than deep features to cause hallucinations and therefore the two output distributions derived from these two types of features can form a contrastive pair, which allows the model to pinpoint and then cancel out hallucinations in the output. Let  $z_s$  denote image features extracted from a shallow layer (e.g., the first layer) in the vision encoder,  $z_d$  image features extracted from a deep layer (e.g.,

the last layer), and  $f(\cdot)$  the LLM, the contrastive probability distribution can be formulated as

$$p(\mathbf{y}|\mathbf{x}, \mathbf{z}_d, \mathbf{z}_s) = \sigma[(1+\alpha)f(\mathbf{x}, \mathbf{z}_d) - \alpha f(\mathbf{x}, \mathbf{z}_s)], \tag{1}$$

where  $\sigma[\cdot]$  is the softmax function, and  $\alpha$  is a hyperparameter that controls the amplification of the difference between the two distributions. In particular, a larger  $\alpha$  indicates a stronger contrast between the two distributions. When  $\alpha = 0$ , the formulation returns to regular decoding.

Since LayerCD only modifies the next-token probability distribution, it can be easily combined with existing decoding methods, such as nucleus sampling, beam search, and others. It is worth noting that the formulation in Eq. 1 does not require modifying the internal model parameters. Unless otherwise specified, we select the first and last vision encoder layer for extracting  $z_s$  and  $z_d$ , respectively.

Adaptive Plausibility Constraint The formulation in Eq. 1 indiscriminately penalizes all outputs from the shallow-feature model, an assumption that is too strong in practice. Common tokens like articles (e.g., "a", "the") are often predicted with high confidence regardless of feature depth, and should not be penalized. To prevent the generation of implausible outputs, we introduce an adaptive plausibility constraint, inspired by Li et al. Li et al. [2022]. At each inference step, we dynamically filter the vocabulary by removing tokens whose confidence in the original deep-feature distribution falls below a threshold based on the maximum confidence. This ensures that LayerCD is computed using only this updated set of plausible tokens:

$$\mathcal{V}_{\text{head}}(\boldsymbol{y}_{< t}) = \{ y_t \in \mathcal{V} : p(y_t | \boldsymbol{x}, \boldsymbol{z}_d, \boldsymbol{y}_{< t}) \ge \beta \max_{w} p(w | \boldsymbol{x}, \boldsymbol{z}_d, \boldsymbol{y}_{< t}) \},$$
(2)

where V is the vocabulary of the MLLM and  $\beta$  is a hyperparameter between 0 and 1 that controls how aggressively low-confidence tokens are pruned. A larger  $\beta$  results in a more aggressive truncation, retaining only tokens with higher confidence.<sup>1</sup>

By combining the contrastive probability distribution with the adaptive plausibility constraint, the *t*-th token is computed as

$$p(y_t|\mathbf{x}, \mathbf{z}_d, \mathbf{z}_s, \mathbf{y}_{\le t})$$
 s.t.  $y_t \in \mathcal{V}_{\text{head}}(\mathbf{y}_{\le t})$ . (3)

## 3 Experiments on POPE

**POPE** We evaluate hallucination using the POPE benchmark Li et al. [2023a], which contains images from MSCOCO Lin et al. [2014], A-OKVQA Schwenk et al. [2022], and GQA Hudson and Manning [2019]. Models must answer yes/no questions probing for objects that are absent, generated via *random*, *popular* (high-frequency), and *adversarial* (co-occurring) sampling. We report the average Accuracy, Precision, Recall, and F1 scores over five runs.

Models and Baseline To evaluate the robustness and adaptability of our methods, we select three state-of-the-art MLLMs with diverse architectures: LLaVA-v1.5-7B Liu et al. [2024c], Cambrian-8B Tong et al. [2024], and MoLmo-7B-D Deitke et al. [2025]. LLaVA-v1.5-7B utilizes a CLIP ViT-L/14@336 Radford et al. [2021] vision encoder and the Vicuna-v1.5-7B Zheng et al. [2023] LLM. Cambrian-8B combines multiple vision encoders (e.g., CLIP ViT-L/14@336 Radford et al. [2021], ConvNeXt-XXL@1024 Woo et al. [2023], DINOv2 Giant Oquab et al. [2023], and SigLIP ViT-SO400M/14@384 Zhai et al. [2023b]) with the LLaMA3-8B-Instruct Dubey et al. [2024] LLM. We chose MoLmo-7B-D, which pairs a CLIP ViT-L/336 Radford et al. [2021] encoder with the QWen2-7B Yang et al. [2024] LLM, due to its superior performance over QWen-VL Bai et al. [2023] on 11 benchmarks.

For baselines, we compare LayerCD with regular decoding and VCD Leng et al. [2024]. Please refer to Appendix for discussions on the differences between LayerCD and VCD.

Implementation Details For LayerCD, we set the contrastive amplification parameter  $\alpha$  to 1 and the plausibility constraint's truncation parameter  $\beta$  to 0.1. The shallow features  $z_s$  and deep features  $z_d$  are extracted from the first and last default layer of the vision encoder, respectively. The output tokens are sampled from the post-softmax layer after the decoding strategy is applied.

<sup>&</sup>lt;sup>1</sup>Similar to Eq. 1, we compute this constraint using logit value instead of post-softmax probability.

Setting	Model	Decoding	Accuracy	Precision	Recall	F1 Score
	LLaVA1.5	Regular VCD LayerCD	$83.21_{\pm 0.49} \ 82.33_{\pm 0.16} \ 85.77_{\pm 0.25}$	$92.21_{\pm 0.70}$ $95.42_{\pm 0.52}$ $96.39_{\pm 0.32}$	$72.55_{\pm 0.63} \\ 67.93_{\pm 0.30} \\ \textbf{74.32}_{\pm 0.59}$	$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$
Random	Cambrian	Regular VCD LayerCD	$62.47_{\pm 0.40} $ $64.59_{\pm 0.56} $ <b>75.65</b> <sub><math>\pm 0.33</math></sub>	$81.22_{\pm 1.58}$ $94.54_{\pm 1.43}$ <b>97.21</b> <sub><math>\pm 0.22</math></sub>	$32.45_{\pm 0.55}$ $30.96_{\pm 0.82}$ $52.81_{\pm 0.75}$	$\begin{array}{c} 46.37_{\pm 0.56} \\ 46.64_{\pm 1.04} \\ \textbf{68.44}_{\pm 0.60} \end{array}$
	Molmo	Regular VCD LayerCD	$45.67_{\pm 4.08} \\ 50.29_{\pm 0.47} \\ \textbf{60.79}_{\pm 0.05}$	$39.87_{\pm 8.55}$ $50.58_{\pm 0.96}$ <b>63.49</b> <sub>±0.11</sub>	$18.85_{\pm 6.24}$ $24.80_{\pm 1.04}$ $50.81_{\pm 0.56}$	$\begin{array}{c c} 25.54_{\pm 7.44} \\ 33.28_{\pm 1.14} \\ \textbf{56.45}_{\pm 0.30} \end{array}$
	LLaVA1.5	Regular VCD LayerCD	$81.83_{\pm 0.47}$ $80.99_{\pm 0.17}$ $84.25_{\pm 0.25}$	$88.99_{\pm 0.53}$ $91.87_{\pm 0.57}$ <b>92.60</b> <sub><math>\pm 0.17</math></sub>	$72.64_{\pm 0.66} \\ 68.01_{\pm 0.28} \\ \textbf{74.44}_{\pm 0.59}$	$ \begin{array}{c c} 79.99_{\pm 0.55} \\ 78.16_{\pm 0.16} \\ \textbf{82.53}_{\pm 0.34} \end{array} $
Popular	Cambrian	Regular VCD LayerCD	$61.18_{\pm 0.56} \ 62.91_{\pm 0.68} \ \textbf{73.87}_{\pm 0.30}$	$76.19_{\pm 1.84} \ 88.19_{\pm 1.28} \ $ <b>91.45</b> $_{\pm 0.71}$	$32.56_{\pm 0.55}$ $29.81_{\pm 1.18}$ <b>52.68</b> $_{\pm 0.68}$	$\begin{array}{c} 45.61_{\pm 0.64} \\ 44.55_{\pm 1.42} \\ \textbf{66.84}_{\pm 0.50} \end{array}$
	Molmo	Regular VCD LayerCD	$44.78_{\pm 3.23} \\ 49.21_{\pm 0.32} \\ \textbf{58.83}_{\pm 0.33}$	$37.52_{\pm 7.52} \ 48.22_{\pm 0.77} \ 61.60_{\pm 0.45}$	$16.43_{\pm 4.45} \\ 21.47_{\pm 0.77} \\ \textbf{46.92}_{\pm 0.26}$	$\begin{array}{ c c c c c }\hline 22.83{\scriptstyle \pm 5.68}\\ 29.70{\scriptstyle \pm 0.89}\\ \textbf{53.27}{\scriptstyle \pm 0.34}\\ \hline\end{array}$
	LLaVA1.5	Regular VCD LayerCD	$79.04_{\pm 0.43} \\ 78.32_{\pm 0.35} \\ \textbf{82.09}_{\pm 0.43}$	$83.35_{\pm 0.41} \ 85.72_{\pm 0.93} \ $ 87.96 $_{\pm 0.82}$	$72.57_{\pm 0.62}$ $67.97_{\pm 0.33}$ $74.36_{\pm 0.50}$	$\begin{array}{ c c c c c c }\hline 77.59_{\pm 0.50} \\ 75.82_{\pm 0.26} \\ \textbf{80.59}_{\pm 0.42} \end{array}$
Adversarial	Cambrian	Regular VCD LayerCD	$61.23_{\pm 0.44} \ 62.31_{\pm 0.36} \ \textbf{73.97}_{\pm 0.28}$	$76.66_{\pm 1.45} \\ 84.67_{\pm 1.71} \\ \textbf{91.94}_{\pm 0.56}$	$32.31_{\pm 0.67} \ 30.12_{\pm 1.32} \ $ 52.56 $_{\pm 0.66}$	$\begin{array}{c} 45.45_{\pm 0.69} \\ 44.40_{\pm 1.29} \\ \textbf{66.88}_{\pm 0.50} \end{array}$
	Molmo	Regular VCD LayerCD	$44.81_{\pm 3.29} \\ 48.43_{\pm 0.35} \\ \textbf{58.06}_{\pm 0.62}$	$39.05_{\pm 6.78} \ 46.87_{\pm 0.77} \ $ <b>59.92</b> $_{\pm 0.76}$	$19.25_{\pm 4.51} \\ 23.59_{\pm 0.96} \\ \textbf{48.69}_{\pm 0.62}$	$\begin{array}{c c} 25.77_{\pm 5.50} \\ 31.38_{\pm 1.00} \\ \textbf{53.73}_{\pm 0.68} \end{array}$

Table 1: Results on POPE-MSCOCO. Our LayerCD outperforms VCD and regular decoding by significant margins. The results on POPE-A-OKVQA and POPE-GQA are provided in the supplementary where the conclusion remains the same.

**Results on POPE** Table 1 summarizes the results of applying different decoding strategies to different base models in the three distinct sampling settings. Overall, LayerCD achieves robust performance across all sampling settings and with different base models. Compared with regular decoding and VCD, LayerCD gains significant improvements consistently across all settings. These results strongly demonstrate the effectiveness of using shallow features to filter out object hallucinations (POPE only focuses on object hallucinations). It is worth noting that all models have relatively weak performance in terms of recall. This is probably due to the training data bias that causes the model to answer "No". Nonetheless, LayerCD still demonstrates huge gains against the baselines.

## 4 Conclusion and Limitations

In this work, we propose Layer Contrastive Decoding (LayerCD), a simple and effective approach built on the key observation that shallow visual features are significantly more prone to inducing MLLM hallucinations than deep ones. By leveraging the contrast between these feature levels, LayerCD effectively pinpoints and cancels out erroneous content.

A primary limitation of our approach is the doubled computational cost, as it requires separate forward passes for shallow and deep features—a drawback inherited from the original contrastive decoding design. This presents a challenge for extremely large models, and we leave the development of more efficient contrastive frameworks for future work.

# 5 Acknowledgements

This research is supported by Hong Kong Research Grants Council Early Career Scheme (No. 22200824).

#### References

- Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 1(2):3, 2023.
- Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28, 2015.
- Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024.
- Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. pages 91–104, June 2025.
- 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.
- 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.
- Markus Freitag and Yaser Al-Onaizan. Beam search strategies for neural machine translation. *arXiv preprint* arXiv:1702.01806, 2017.
- 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. Mme: A comprehensive evaluation benchmark for multimodal large language models, 2024. URL https://arxiv.org/abs/2306.13394.
- Ulrich Germann. Greedy decoding for statistical machine translation in almost linear time. In Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, pages 72–79, 2003.
- Anisha Gunjal, Jihan Yin, and Erhan Bas. Detecting and preventing hallucinations in large vision language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18135–18143, 2024.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. arXiv preprint arXiv:1904.09751, 2019.
- Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13418–13427, 2024.
- Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and* pattern recognition, pages 6700–6709, 2019.
- Jitesh Jain, Jianwei Yang, and Humphrey Shi. Vcoder: Versatile vision encoders for multimodal large language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 27992–28002, 2024.

- Chaoya Jiang, Haiyang Xu, Mengfan Dong, Jiaxing Chen, Wei Ye, Ming Yan, Qinghao Ye, Ji Zhang, Fei Huang, and Shikun Zhang. Hallucination augmented contrastive learning for multimodal large language model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 27036–27046, 2024.
- Rémi Lebret, David Grangier, and Michael Auli. Neural text generation from structured data with application to the biography domain. *arXiv preprint arXiv:1603.07771*, 2016.
- Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. arXiv preprint arXiv:2107.06499, 2021.
- Nayeon Lee, Wei Ping, Peng Xu, Mostofa Patwary, Pascale N Fung, Mohammad Shoeybi, and Bryan Catanzaro. Factuality enhanced language models for open-ended text generation. Advances in Neural Information Processing Systems, 35:34586–34599, 2022.
- 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.
- Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. arXiv preprint arXiv:2210.15097, 2022.
- Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023a.
- Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-modal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26763–26773, 2024.
- Zuchao Li, Shitou Zhang, Hai Zhao, Yifei Yang, and Dongjie Yang. Batgpt: A bidirectional autoregessive talker from generative pre-trained transformer. *arXiv preprint arXiv:2307.00360*, 2023b.
- Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. arXiv preprint arXiv:2109.07958, 2021.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.
- Bingbin Liu, Jordan Ash, Surbhi Goel, Akshay Krishnamurthy, and Cyril Zhang. Exposing attention glitches with flip-flop language modeling. *Advances in Neural Information Processing Systems*, 36, 2024a.
- Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Mitigating hallucination in large multi-modal models via robust instruction tuning. In *The Twelfth International Conference on Learning Representations*, 2023.
- Hanchao Liu, Wenyuan Xue, Yifei Chen, Dapeng Chen, Xiutian Zhao, Ke Wang, Liping Hou, Rongjun Li, and Wei Peng. A survey on hallucination in large vision-language models. arXiv preprint arXiv:2402.00253, 2024b.
- 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, 2024c
- Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. Entity-based knowledge conflicts in question answering. *arXiv preprint arXiv:2109.05052*, 2021.
- Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. arXiv preprint arXiv:2005.00661, 2020.
- Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193, 2023.
- Ankur P Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. Totto: A controlled table-to-text generation dataset. *arXiv preprint arXiv:2004.14373*, 2020.

- 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.
- Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*, 2020.
- Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European conference on computer vision*, pages 146–162. Springer, 2022.
- 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.
- Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*, 2024.
- Xintong Wang, Jingheng Pan, Liang Ding, and Chris Biemann. Mitigating hallucinations in large vision-language models with instruction contrastive decoding. arXiv preprint arXiv:2403.18715, 2024.
- Sam Wiseman, Stuart M Shieber, and Alexander M Rush. Challenges in data-to-document generation. *arXiv* preprint arXiv:1707.08052, 2017.
- Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16133–16142, 2023.
- An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. arXiv preprint arXiv:2407.10671, 2024.
- Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? Advances in neural information processing systems, 27, 2014.
- Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13, pages 818–833. Springer, 2014.
- Bohan Zhai, Shijia Yang, Xiangchen Zhao, Chenfeng Xu, Sheng Shen, Dongdi Zhao, Kurt Keutzer, Manling Li, Tan Yan, and Xiangjun Fan. Halle-switch: Rethinking and controlling object existence hallucinations in large vision language models for detailed caption. *arXiv* preprint arXiv:2310.01779, 2023a.
- Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pretraining. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 11975–11986, 2023b.
- Muru Zhang, Ofir Press, William Merrill, Alisa Liu, and Noah A Smith. How language model hallucinations can snowball. *arXiv preprint arXiv:2305.13534*, 2023.
- 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
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. Advances in Neural Information Processing Systems, 36:46595–46623, 2023.
- Yiyang Zhou, Chenhang Cui, Jaehong Yoon, Linjun Zhang, Zhun Deng, Chelsea Finn, Mohit Bansal, and Huaxiu Yao. Analyzing and mitigating object hallucination in large vision-language models. arXiv preprint arXiv:2310.00754, 2023.

# A Appendix

#### A.1 Related Work

Hallucination in LLMs The causes of hallucinations in LLMs are multifaceted, with studies linking them to both training data quality Lin et al. [2021] and architectural limitations Li et al. [2023b], Liu et al. [2024a]. For instance, heuristically paired data during dataset construction can sometimes result in inconsistent or unsupported outputs, exacerbating the hallucination issue Lebret et al. [2016], Wiseman et al. [2017]. Additionally, limited diversity in training data, such as repetitive patterns Lee et al. [2021], can bias model outputs and increase the likelihood of hallucinations. Another significant cause for hallucination lies within the model architecture, where issues in representation learning and token embedding processing can distort models' understanding and amplify hallucinations Parikh et al. [2020]. Furthermore, models often prioritize memorized parametric knowledge over real-time input due to reliance on information encoded during training, further compounding the problem Roberts et al. [2020], Longpre et al. [2021]. Decoding strategies also play a role, with some methods introducing early-generation errors that accumulate rather than being corrected Zhang et al. [2023], Lee et al. [2022], Bengio et al. [2015].

Hallucination in MLLMs The causes of hallucinations in MLLMs are more complex than in LLMs due to the integration of visual inputs. Specifically, limitations in the vision encoder, such as low resolution or a bias toward salient objects, have been identified as major contributors to hallucinations Zhai et al. [2023a], Li et al. [2024], Jain et al. [2024]. Additionally, the alignment process between visual and textual representations often fails to accurately synchronize these inputs, further exacerbating hallucination errors Jiang et al. [2024], Chen et al. [2024]. Furthermore, when visual input is incorporated into the model's self-attention mechanism, it often receives insufficient focus, causing the model to rely more on pre-trained knowledge within the LLM component than on the actual visual content Favero et al. [2024], Leng et al. [2024].

Recent efforts to reduce hallucinations have employed various strategies, including the development of fine-grained datasets for improved training Gunjal et al. [2024], Liu et al. [2023], enhancements to the vision encoder Jain et al. [2024], better alignment mechanisms Jiang et al. [2024], and optimized decoding strategies Huang et al. [2024], Leng et al. [2024], Wang et al. [2024]. Post-processing methods have also been utilized to address hallucinations after generation Zhou et al. [2023], and reinforcement learning approaches have been explored to align models more closely with human preferences, improving the accuracy of generated outputs Zhao et al. [2023], Sun et al. [2023].

The most closely related work is Visual Contrastive Decoding (VCD) Leng et al. [2024], which essentially contrasts output distributions generated from the original visual input with those generated from input distorted by Gaussian noise. Compared to VCD, our LayerCD approach addresses the problem from a novel perspective: we leverage shallow features to filter out hallucinations. In the experiments, we demonstrate that LayerCD significantly outperforms VCD on two challenging benchmarks. From the computation perspective, LayerCD performs favorably against VCD: VCD requires two forward passes in the vision encoder (one normal input and one distorted input) while LayerCD only needs one.

#### A.2 Experiments on MME

MME The MME benchmark Fu et al. [2024] provides a comprehensive toolbox to evaluate a wide range of capabilities in MLLMs, with a focus on perception and cognitive skills. This benchmark includes 14 different tasks, 10 of which assess perception-related abilities, while the remaining 4 are focused on cognitive processing. Following prior work Leng et al. [2024], we select the *existence* and *count* subsets to assess object-level hallucinations and the *position* and *color* subsets to evaluate hallucinations related to object attributes. Similar to POPE, MME contains binary questions that require "Yes" and "No" responses. Model performance is quantified using a custom scoring formula, which aggregates various accuracy metrics to provide an overall assessment. To ensure fairness, the results reported are averaged over five runs.

**Results on MME** The MME hallucination subsets have been widely used by the hallucination research community. The results on these subsets are presented in Table 2 (left). Overall, LayerCD performs the best among the three decoding methods. In most settings, LayerCD outperforms the

	MME subset						
Model	Decoding	Existence	Count	Position	Color		
LLaVA1.5	Regular VCD LayerCD	$169.67_{\pm 3.71} \\ 168.67_{\pm 11.99} \\ 176.00_{\pm 3.74}$	$113.67_{\pm 7.18} \\ 115.33_{\pm 11.03} \\ 119.00_{\pm 13.97}$	$117.67_{\pm 10.73} \\ 111.00_{\pm 9.10} \\ \textbf{133.33}_{\pm 11.01}$	$140.67_{\pm 6.55} \\ 143.00_{\pm 9.74} \\ \textbf{157.00}_{\pm 5.10}$		
Cambrian	Regular VCD LayerCD	$98.67_{\pm 8.06} \\ 121.33_{\pm 10.19} \\ \textbf{125.33}_{\pm 10.19}$	$85.67_{\pm 18.15} \ 82.67_{\pm 17.47} \ 88.67_{\pm 9.80}$	$66.33_{\pm 7.48}$ $69.33_{\pm 8.60}$ <b>75.68</b> $_{\pm 8.73}$	$74.33_{\pm 8.34} \\ 81.33_{\pm 9.45} \\ \textbf{103.67}_{\pm 5.81}$		
Molmo	Regular VCD LayerCD	$78.33_{\pm 0.00}$ $70.33_{\pm 3.40}$ <b>78.33</b> <sub><math>\pm 0.00</math></sub>	$73.33_{\pm 0.00}$ $75.00_{\pm 2.58}$ $63.33_{\pm 0.00}$	$53.33_{\pm 0.00}$ $53.67_{\pm 4.40}$ $56.67_{\pm 0.00}$	$48.33_{\pm 0.00}$ $56.67_{\pm 2.36}$ $53.33_{\pm 0.00}$		

Table 2: Results on the MME hallucination subsets. LayerCD beats the two baselines, i.e., regular decoding and VCD, in most cases except the *count* and *color* subsets when Molmo is used as the base model.

baselines with significant margins. In the *existence* and *position* tasks, LayerCD's gains are more significant. However, when using Molmo in the *count* and *color* tasks, LayerCD shows inferior results than VCD. In particular, we observe that the percentage of "No" answers in these two subsets is exceptionally high for Molmo and LayerCD somehow exacerbates this problem, resulting in weaker performance.

#### A.3 Further Analysis

Combining LayerCD with Traditional Decoding Strategies Since LayerCD is theoretically orthogonal to the traditional decoding strategies, we combine them with LayerCD to see the effects. Specifically, we try the following decoding strategies: greedy decoding Germann [2003], beam search Freitag and Al-Onaizan [2017], and regular sampling with top-p Holtzman et al. [2019], top-k, and temperature normalization. We conduct the experiments using LLaVA 1.5 on POPE based on COCO dataset and using random sampling setting. The results are summarized in Table 3. It is clear that LayerCD works well with all the traditional decoding strategies, with top-p, top-k, and temperature normalization demonstrating the strongest synergy (the gains are significant).

Decoding Strategy	w/ LayerCD	Accuracy
Greedy	<b>√</b>	83.19 <b>85.67</b>
Top P $(P = 0.9)$	<b>√</b>	82.96 <b>84.88</b>
Top K $(K = 50)$	<b>√</b>	83.02 <b>85.23</b>
Top K & Temperature $(K = 50; \text{temperature} = 0.7)$	<b>√</b>	84.04 <b>85.36</b>
Top K & Temperature $(K = 50; temperature = 1.5)$	<b>√</b>	83.95 <b>85.02</b>
Beam Search $(Num_{beams} = 3)$	<b>√</b>	84.20 <b>86.11</b>

Table 3: Results of combining different decoding strategies with LayerCD. LayerCD works well with all of them.

**Impact of Hyperparameters**  $\alpha$  **and**  $\beta$  Table 4 shows the results of varying  $\alpha$  and  $\beta$ , which are LayerCD's hyperparameters for controlling the contrastive amplification and constraint truncation, respectively. As  $\alpha$  increases from 0.2 to 1.0, the performance of LayerCD improves accordingly.

		. — <i>E</i>	3	Accuracy
$\alpha$	Accuracy	0.0	01	81.96
0.2	83.7	0.0	)1	83.01
0.4	84.18	0.	1	85.67
0.6	85.33	0.	2	84.75
0.8	82.45	0.	5	81.96
1.0	85.67	0.	9	80.58
	(a)			(b)

Table 4: Impact of hyperparameters  $\alpha$  and  $\beta$ . See Methodology section for their uses.

The results suggest that a higher  $\alpha$  facilitates the amplification of the difference between the two output distributions derived from shallow and deep visual features, respectively, and hence leads to better performance. For  $\beta$ , the results do not have a clear pattern. Though a higher  $\beta$  leads to better performance, we find in practice that setting  $\beta$  too high may reduce output diversity.

**Impact of Adaptive Plausibility Constraint** Table 5 presents the LayerCD results of LLaVA1.5-7B on POPE based on COCO dataset and using random sampling setting, comparing performance with and without the Adaptive Plausibility Constraint. The results indicate that the Adaptive Plausibility Constraint plays a crucial role in enhancing LayerCD's performance.

w/ APC	Average
	62.21
$\checkmark$	85.67

A higher  $\alpha$  amplifies the difference between the two output distributions derived from shallow and deep visual features and makes desirable tokens easier to be selected. Notably, because POPE re-

Table 5: Impact of the Adaptive Plausibility Constraint. *APC* denotes the Adaptive Plausibility Constraint.

sponses are often single tokens and the highest-probability token is usually correct, a higher  $\beta$  can filters out more distractor tokens and increases performance. However, setting  $\beta$  too high may reduce output diversity.