Why do LLaVA Vision-Language Models Reply to Images in English?

Musashi Hinck[1][*], **Carolin Holtermann**[2][†], **Matthew Lyle Olson**[1][†], **Florian Schneider**[2][†]
Sungduk Yu[1], **Anahita Bhiwandiwalla**[3][‡], **Anne Lauscher**[2] **Shaoyen Tseng**[1], **Vasudev Lal**[1],
[1]Intel Labs, [2]University of Hamburg, [3]Nvidia

Correspondence: musashi.hinck@intel.com

Abstract

We identify a novel pathology of multilingual vision-language models (VLMs): adding an image to the input reduces the likelihood that the model will reply in the same language as the query. We term this pathology Image-induced Fidelity Loss (IFL), and study its prevalence, cause and remedies in LLaVA-style VLMs. On prevalence, we show that IFL occurs in four different LLaVA-style VLMs across three sizes and fourteen languages. Systematic experimental ablation of the LLaVA design space shows that among training data language, vision backbone and language backbone, the choice language backbone has the largest impact on IFL. This finding is supported by examination of the input embeddings at the point of multimodal fusion, where visual inputs are encoded separately to textual ones, regardless of language. Finally, we show that a lightweight intervention technique from the mechanistic interpretability literature can reduce IFL. In sum, we formalize a novel challenge arising in multilingual multimodal settings, and comprehensively analyze its prevalence and causes in a popular class of VLMs.

1. Introduction

When generating text in response to a query, *language fidelity* refers to whether the returned text is in the same language as the query. While a seemingly simple task for humans, researchers have found that large language models (LLMs) with multilingual capabilities bias towards generating English text, regardless of the query language[10].

In this work, we identify a surprising parallel pathology: adding an image to the prompt of a VLM increases the likelihood that it will reply in the ”wrong” language. In the first part of this paper we formalize this pathology, and term it *Image-induced Fidelity Loss* (IFL). In the subsequent section, we study *how commonly it occurs* (prevalence) and *why it occurs* (cause). Focusing on LLaVA-style VLMs [16], we measure prevalence across four variants and fourteen languages, comprehensively ablate the design space to identify the training choices that most contribute to IFL, and use mechanistic interpretability (MI) to trace the mechanisms of IFL within the model.

On prevalence, experiments on 7740 evaluation tasks in fourteen languages show the effect of adding an image to the query on the probability of the response to be in the correct language ranges from -0.06 to -0.53. Our experiments indicate that IFL is primarily attributable to the language backbone on the VLM. In order to estimate the effect of three key design choices–training data language, vision backbone and language backbone–we train separate LLaVA models for every possible combination of these three factors. We then show that of the three factors, the choice of language backbone has the greatest effect on IFL in the downstream LLaVA model. These results are corroborated by our examination of the internal representations in LLaVA models. In particular, we find that we are able to mitigate IFL by intervening on the residual stream in the language model using a simple steering technique.

The final section discusses limitations and directions for future work. We focus our analysis on LLaVA-style VLMs for their ubiquity and popularity, and as a ”model organism” to conduct our in-depth analyses. We also make extensive use of machine translation in order to construct parallel training corpora, which we acknowledge introduces bias into our analyses. Future work should extend this analysis to other VLM types such as Flamingo [2].

Our contributions are:

- We formalize a novel pathology of multilingual VLMs.
- We demonstrate its prevalence in LLaVA-style VLMs.
- We identify that choice of language backbone has the strongest design effect on IFL.
- We show that intervention on the residual stream can mitigate IFL.

2. Image-induced Fidelity Loss

Following the success of text-only foundation models [4, 5], researchers have extended foundation models to vi-

[*] First author.

[†] Equal contribution, ordered alphabetically.

[‡] Work done while at Intel Labs.

sual modalities, creating large, pretrained models that can process image and text inputs and generate text outputs [12, 32]. Given the prohibitive costs of full pretraining, researchers have developed frameworks for training VLMs from partially frozen pretrained components [15, 19]. In this work, we focus on LLaVA, a popular example of an efficient framework for creating VLMs [16].

VLMs can take image and/or text as inputs, and generate text in response. When the language of the input and output are the same, we refer to this as having *language fidelity*. The phenomenon of interest in this paper is the *decrease* in fidelity associated with adding an image to the input of a VLM, which we call *Image-induced Fidelity Loss* (IFL).

Definitions Given an input x (text and/or image), we define the function $L(x)$ as returning the (natural) language of the input. The fidelity of a given text-generating model $\theta(\cdot)$ and input x is defined as a binary indicator of whether the language of the input $L(x)$ equals the language of the output $L(\theta(x))$:

$$F(\mathbf{x}) = \begin{cases} 1 & \text{if } L(\mathbf{x}) = L(\theta(\mathbf{x})), \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

We investigate the impact of including an image on fidelity. We compare inputs containing an image (x_{image}) against inputs where the image is replaced with a textual description of the image's content, $x_{description}$. The rationale for substituting the image with a textual description of the image's content is to maintain the semantic value of the input constant.

Thus, for each document (consisting of image and text pair) in our evaluation dataset, we define the Image Fidelity Loss (IFL) as:

$$\text{IFL} = F(x_{description}) - F(x_{image}) \tag{2}$$

representing the fidelity loss incurred by substituting a text description of an image with the actual image.

Language Detection and Bias Correction We use the GlotLID model [11] to predict the language of the model output ($L(x)$). In order to correct for errors in the GlotLID predictions, we use the bias-corrected estimators from the design-based supervised learning [8, DSL] framework. DSL leverages a small number of randomly sampled expert annotations to correct for bias in downstream estimators. We manually label a stratified random sample of 1000 examples to use as our gold standard. The debiased results can be interpreted as being the results that would have been obtained if we had used expert annotation for all datasets. We provide details of the sampling weights and annotation method in the supplementary materials.

Multilingual Multimodal Datasets These image-text pairs are drawn from the three multilingual VQA benchmarks: MaXM [6], PALO-LLaVAW [17, hereafter LLaVAW] and ViSIT [3], and summarized in Table 4 in the SI. These datasets all include a textual query referring to an image, plus a textual description of the image. In the case of ViSIT this description is generated conditional on the task instruction and verified by human annotators. In total, for each model we collect 15480 responses spanning fourteen languages (7740 times 2 conditions; see Table 5 in the SI for a breakdown).

Prevalence of IFL We apply the experimental design outline above to four popular LLaVA-style VLMs: LLaVA-v1.5-7b, LLaVA-v1.5-13b [16], BakLLaVA [27] and LLaVA-Gemma-2B [9].

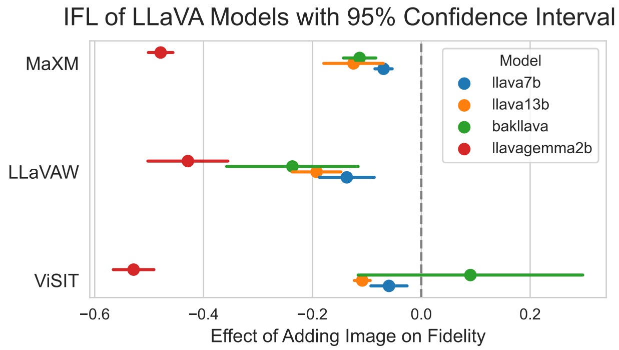

Figure 1. **IFL prevalence among existing LLaVA models.** Effect of adding image to query on response fidelity (IFL) with 95% confidence intervals. All values are DSL bias-corrected estimates of change in probability, aggregated over all languages.

Figure 1 shows the debiased estimated effect and 95% confidence interval of adding an image on fidelity for each model and benchmark, aggregated across languages.

The magnitude varies by model and benchmark. The single largest drop is by LLaVA-Gemma-2b on ViSIT, where the response is 52.9 percentage points more likely to be in a different language than the query when an image is included in the input. With the exception of BakLLaVA on ViSIT, all effects are statistically significant at an alpha of 0.95. Because we use the DSL framework for estimation, these claims are statistically robust to errors from the language identification model.

Collectively, these results provide concrete evidence of a systematic issue: LLaVA models are more likely to reply in the *wrong* language when the user includes an image in the query. The remainder of this paper explores the source of this issue.

3. Effects of Design Choices

The LLaVA architecture combines a pretrained vision encoder and language model by using a small multi-layer perceptron (MLP) to project the penultimate hidden states of the vision encoder into the input embedding space of the language model [16]. This architecture is fine-tuned with two stages of training. First the vision and language models are frozen and the projection MLP is trained on 558k image-caption pairs. Next the vision encoder is kept frozen and the projection MLP and language model are trained on 665k visual instruction-following and examples [16].

Design Space The base LLaVA-1.5 model uses Vicuna-v1.5-7b [33] as the language backbone, CLIP [24] as the vision encoder and English for more than 99% its training examples. There are *a priori* reasons to think that any of these decisions could induce an "English bias" in the model. Vicuna is published as an English-language LLM trained primarily on English-language examples. The captions used to train the CLIP vision encoder are filtered for non-English texts [24, p.3], meaning that the representations produced by CLIP may be "biased" towards English language representations of visual data. Finetuning the model with primarily English data may "teach" the language model to reply to visual inputs from the vision encoder/MLP in English.

We ablate these design choices individually to disentangle their effects. For our experiments, we focus on Chinese and German because these are languages for which there is an LLM at a similar size and architecture to Vicuna-7b that is not directly finetuned from Vicuna-7b. For Chinese, we use the Yi-6b-chat, a 6B-parameter LLM trained from scratch on a bilingual Chinese-English data mixture [1]. For German, we use LeoLM-7b-chat, a 7B-parameter LLM finetuned from Llama-2 [30] on 65B German tokens [22]. For the vision encoder, we test the effect of substituting CLIP for DINOv2 [21] because the latter is trained using a self-supervised training objective that does not incorporate language, while still using a ViT [7]. We use NLLB-1.7-distilled [28] to machine translate all ∼1.2M training observations used in the LLaVA training data into Chinese and German. We provide estimates of the machine translation quality following techniques in [23] in the supplement.

Language	Vision	EN	ZH	DE
Vicuna-v1.5-7b	CLIP	✓	✓	✓
	DINOv2	✓	✓	✓
Yi-6b-chat	CLIP	✓	✓	–
	DINOv2	✓	✓	–
Leo-7b-chat	CLIP	✓	–	✓
	DINOv2	✓	–	✓

Table 1. Supported configurations of language backbone, vision backbone and training data language.

This design yields a total of fourteen combinations (Table 1). All designs used the same training parameters as the original LLaVA-v1.5-7B model. We provide further training details in the supplementary materials.

Model	IFL	Accuracy
	Chinese	
LLM	0.17 [0.15, 0.19]	0.21 [-0.07, 0.50]
VE	-0.20 [-0.22, -0.18]	0.15 [-0.13, 0.43]
Data	-0.16 [-0.17, -0.14]	0.01 [-0.27, 0.30]
	German	
LLM	0.07 [0.04, 0.10]	0.28 [-0.35, 0.91]
VE	-0.11 [-0.15, -0.08]	-0.10 [-0.73, 0.53]
Data	-0.37 [-0.40, -0.33]	-0.24 [-0.87, 0.40]

Table 2. Point estimate and 95% confidence interval of the effect of changing the design feature (LLM, vision encoder or training data language, corresponding to β_2, β_3 and β_4 in equation 3) on IFL (left-hand column) and accuracy (right-hand column). Chinese and German are reported top and bottom respectively.

Design Effects For each set of experiments (Yi/Chinese and Leo/German), we measure the effect of training choices on IFL using the following regression model with first-order interactions:

$$\begin{aligned} \mathit{Fidelity} = \beta_0 &+ \beta_1 \mathit{Image} \\ &+ \beta_2 \mathit{Image} \times \mathit{LLM} \\ &+ \beta_3 \mathit{Image} \times \mathit{VE} \\ &+ \beta_4 \mathit{Image} \times \mathit{Data} + \epsilon \end{aligned} \tag{3}$$

where:

- *Fidelity* is a binary indicator for whether a completion has fidelity (Equation 1).
- $\beta_2 \mathit{Image} \times \mathit{LLM}$ is change in IFL when the LLM is changed from from Vicuna to Yi or Leo
- $\beta_3 \mathit{Image} \times \mathit{VE}$ is change in IFL when the vision backbone is changed from CLIP to DINOv2
- $\beta_4 \mathit{Image} \times \mathit{Data}$ is change in IFL when the training language is changed from English to Chinese or German
- β_0, β_1 and ϵ are not relevant to the analysis

Coefficients β_2, β_3 and β_4 with the corresponding 95% confidence interval are reported in the left-hand side of Table 2. We see similar patterns for both languages. Changing the language model from Vicuna to Yi/Leo improved the performance of the model, reducing IFL by 17 and 7 pp for Chinese and German respectively. Changing the vision encoder from CLIP to DinoV2 worsened IFL, increasing it by 20 and 11 pp respectively. Changing the training data language worsened IFL considerably, increasing it by 16 and 37 pp respectively. We discuss the effects on accuracy in the supplement.

4. Locating the Cause of IFL

Embeddings Analysis To understand the interaction between image and text embeddings in the input space, we employ Uniform Manifold Approximation and Projection

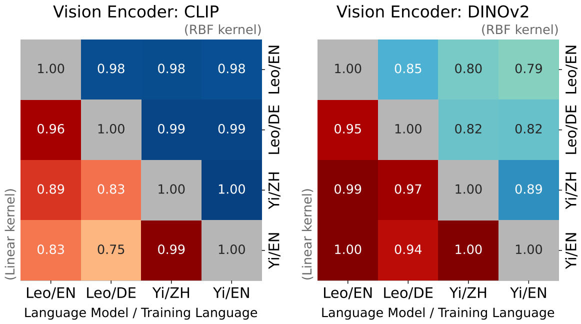

Figure 2. Centered Kernel Alignment (CKA) heatmap showing the similarity of vision embeddings across two differently trained language models. Linear kernel CKA is shown in the lower triangle; RBF kernel CKA is shown in the upper triangle.

(UMAP) for dimensionality reduction [18], a non-linear dimensionality reduction technique that preserves global data structure. Figure 3 (SI) illustrates that image embeddings cluster distinctly from text, demonstrating a demarcated separation in the latent space. This segregation suggests image embeddings occupy a unique region of the embedding space, indicating they are not directly embedding in the same area as any particular language.

To further understand the image embeddings, we use Centered Kernel Alignment (CKA) to measure the similarity of internal representations across differently trained models [13]. CKA measures the similarity between two sets of data by comparing kernel matrices, which transform data into a high-dimensional space. A CKA score close to 1 indicates high similarity between datasets, while a score near 0 suggests low similarity. We measure how the vision embeddings compare between two separately trained VLMs: LLaVA-Yi trained in Chinese and LLaVA-Leo in German.

Figure 2 shows that vision embeddings maintain a consistent structure in the latent space across various models, regardless of the language backbone or the training data specifics. This supports the finding in the UMAP visualization that image embeddings are in their own region of the input space. This suggests that the language model is "responsible" for interpreting out-of-distribution embeddings, and the MLP adaptor is not placing the image embeddings closer to a particular language.

Mechanistic Intervention Drawing from recent work in Mechanistic Interpretability, we propose a simple training-free intervention for improving fidelity that uses just one text example per language.

Our steering mechanism works by computing a language attribute a_{lang} in an intermediate layer, then applying that attribute to every generated token, following ActAdd[31]. The attribute is computed as follows:

$$a_{lang} = LLM_l(x_{lang}) - LLM_l(x_{en}) \tag{4}$$

Model	IFL	IFL + Remedy	Diff.	Relative Increase
llava7b	-0.085	-0.030	0.055	65%
llava13b	-0.175	-0.103	0.073	42%
bakllava	-0.073	0.098	0.170	233%
llava-gemma2b	-0.681	-0.513	0.168	25%

Table 3. Fidelity improvements by using mechanistic intervention (Remedy). Across all pretrained models, we find significant reduction in IFL by interventing on the LLM's intermediate layer.

where LLM_l represents the output at layer l, x_{en} is the sentence "Describe this image in detail.", and x_{lang} is the translated version of that sentence.

During inference, this direction is added to the output of layer l, effectively steering the model's behavior towards the desired language:

$$LLM{*}_l = LLM_l(o_{l-1}) + a_{lang} \tag{5}$$

where o_{l-1} is the output of the previous layer and $LLM{*}_l$ is the new, intervened layer. For our experiments layer l is selected to be partway through computation at one third depth (e.g., layer 10 out of 30).

We find large relative reductions in the prevalence of IFL across the board, which we report in table 3. While this strategy requires knowledge about which language attribute to select, it provides strong evidence supporting the hypothesis that the LLM is responsible for IFL. Moreover, the successful application of a targeted mechanistic intervention highlights the potential of this approach to mitigate issues related to IFL, an area of research we will explore further.

5. Limitations and Conclusion

Our work has several limitations. First, we focus on a single VLM "type"–future work will extend the investigation to other kinds of VLMs. Second, our estimates of fidelity are based on a silver-standard language identification model. We account for this by mixing in gold-standard annotations using DSL. Third, we use machine translation to construct parallel training and evaluation corpora, which may introduce noise into our results.

Nevertheless, our paper provides strong evidence of three results: 1) IFL is prevalent across LLaVA-style VLMs, 2) the mechanism for IFL occurs in the language backbone and 3) IFL may be mitigated with a simple steering technique involving inference-time intervention on the language backbone residual stream. We hope that our work will draw attention to the unique challenges of developing multilingual multimodal foundation models, and the scientific opportunities for systematic inquiry that these provide.

References

[1] 01. AI, :, Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, Kaidong Yu, Peng Liu, Qiang Liu, Shawn Yue, Senbin Yang, Shiming Yang, Tao Yu, Wen Xie, Wenhao Huang, Xiaohui Hu, Xiaoyi Ren, Xinyao Niu, Pengcheng Nie, Yuchi Xu, Yudong Liu, Yue Wang, Yuxuan Cai, Zhenyu Gu, Zhiyuan Liu, and Zonghong Dai. Yi: Open foundation models by 01.ai, 2024. 3, 9

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning, 2022. 1

[3] Yonatan Bitton, Hritik Bansal, Jack Hessel, Rulin Shao, Wanrong Zhu, Anas Awadalla, Josh Gardner, Rohan Taori, and Ludwig Schimdt. VisIT-Bench: A Benchmark for Vision-Language Instruction Following Inspired by Real-World Use. *arXiv preprint arXiv:2308.06595*, 2023. 2, 8

[4] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the opportunities and risks of foundation models, 2022. 1

[5] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. 1

[6] Soravit Changpinyo, Linting Xue, Michal Yarom, Ashish Thapliyal, Idan Szpektor, Julien Amelot, Xi Chen, and Radu Soricut. MaXM: Towards Multilingual Visual Question Answering. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2667–2682, Singapore, 2023. 2, 8

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 3

[8] Naoki Egami, Musashi Hinck, Brandon Stewart, and Hanying Wei. Using imperfect surrogates for downstream inference: Design-based supervised learning for social science applications of large language models. In *Advances in Neural Information Processing Systems*, pages 68589–68601. Curran Associates, Inc., 2023. 2

[9] Musashi Hinck, Matthew L. Olson, David Cobbley, Shao-Yen Tseng, and Vasudev Lal. Llava-gemma: Accelerating multimodal foundation models with a compact language model, 2024. 2

[10] Carolin Holtermann, Paul Röttger, Timm Dill, and Anne Lauscher. Evaluating the elementary multilingual capabilities of large language models with multiq, 2024. 1, 8

[11] Amir Kargaran, Ayyoob Imani, François Yvon, and Hinrich Schuetze. GlotLID: Language identification for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6155–6218, Singapore, 2023. Association for Computational Linguistics. 2, 8

[12] Wonjae Kim, Bokyung Son, and Ildoo Kim. ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision, 2021. arXiv:2102.03334 [cs, stat]. 2

[13] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR, 2019. 4

[14] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The Open Images Dataset v4: Unified Image Classification, Object Detection, and Visual Relationship Detection at Scale. *International journal of computer vision*, 128(7):1956–1981, 2020. 8

[15] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping Language-Image Pre-training with

Frozen Image Encoders and Large Language Models, 2023. arXiv:2301.12597 [cs]. 2

[16] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual Instruction Tuning. In *Advances in Neural Information Processing Systems*, pages 34892–34916, New Orleans, LT, USA, 2023. Curran Associates, Inc. 1, 2, 3, 8, 9

[17] Muhammad Maaz, Hanoona Rasheed, Abdelrahman Shaker, Salman Khan, Hisham Cholakal, Rao M Anwer, Tim Baldwin, Michael Felsberg, and Fahad S Khan. PALO: A Polyglot Large Multimodal Model for 5B People. *arXiv preprint arXiv:2402.14818*, 2024. 2, 8

[18] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018. 4

[19] Jack Merullo, Louis Castricato, Carsten Eickhoff, and Ellie Pavlick. Linearly Mapping from Image to Text Space, 2023. arXiv:2209.15162 [cs]. 2

[20] OpenAI. Gpt-4o: Documentation, 2024. Accessed: 2024-06-15. 7

[21] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. 3, 9

[22] Björn Plüster. LeoLM: Igniting German-Language LLM Research, 2023. Accessed: 2024-06-15. 3, 9

[23] Chen Qiu, Dan Oneață, Emanuele Bugliarello, Stella Frank, and Desmond Elliott. Multilingual multimodal learning with machine translated text. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4178–4193, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics. 3, 9

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. ISSN: 2640-3498. 3, 9

[25] Florian Schneider and Sunayana Sitaram. M5 – a diverse benchmark to assess the performance of large multimodal models across multilingual and multicultural vision-language tasks, 2024. 12

[26] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia, 2018. Association for Computational Linguistics. 10

[27] SkunkworksAI. Bakllava-1, 2023. Hugging Face model repository. 2, 9

[28] NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. No language left behind: Scaling human-centered machine translation, 2022. 3, 9

[29] Ashish V. Thapliyal, Jordi Pont Tuset, Xi Chen, and Radu Soricut. Crossmodal-3600: A Massively Multilingual Multimodal Evaluation Dataset. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 715–729, Abu Dhabi, United Arab Emirates, 2022. 8

[30] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. 3

[31] Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023. 4

[32] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. SimVLM: Simple Visual Language Model Pretraining with Weak Supervision, 2022. arXiv:2108.10904 [cs]. 2

[33] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. 3

A. Figures and Tables Referenced in Article

Dataset	MM	#Langs.	Size
PALO-LLaVAW	yes	10	600
MaXM	yes	7	2142
ViSIT	yes	10	5740
MultiQ	no	119	27400

Table 4. Overview of employed datasets, indicating multimodality (MM), number of languages (#Langs.), and total observations.

Language	N	Language	N
Chinese (zh)	862	Japanese (ja)	585
Hindi (hi)	845	Spanish (es)	585
English (en)	842	German (de)	525
Hebrew (he)	805	French (fr)	324
Thai (th)	793	Romanian (ro)	284
Arabic (ar)	585	Russian (ru)	60
Bengali (bn)	585	Urdu (ur)	60

Table 5. Number of tasks per language in the datasets we study.

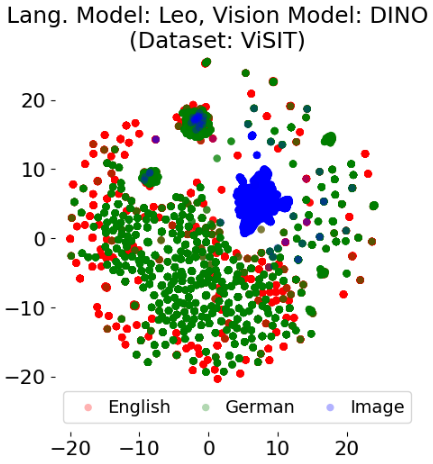

Figure 3. UMAP visualization of image and text embeddings from a multimodal language model. Image embeddings are shown clustering distinctly from text embeddings, indicating a unique separation in the latent space.

B. Additional Analysis: Effect on Accuracy

Although the above section provides insights into reducing IFL, how do these design decisions affect the factual accuracy of responses? To measure this, we used GPT-4o [20] to generate zero-shot predictions of the accuracy. Our GPT-4o prompt gave the question, dataset ground truth (where available) and model completion and asked if the completion is correct given the question and ground truth label. We then used the DSL procedure to debias these evaluations, whereby the authors manually annotated 1000 observations to provide a gold standard.

We use the same regression setup as Equation 3, substituting the outcome *Fidelity* for *Accuracy*, a binary variable indicating whether a given response is correct given the question and dataset ground truth. The right-hand column of table 2 displays the effect of each design decision on the accuracy of responses in the target language.

We find no evidence for a systematic effect of any of the design decisions on accuracy. All estimated coefficients are statistically indistinguishable from 0, meaning our data does not support the hypothesis that changing the LLM, vision encoder or training data in the way described has a systematic effect on the accuracy of the response.

C. Computational Experiments

Computational Budget The training experiments for this paper were conducted on an internal cluster using nodes with 8 × A6000 Nvidia 48GB GPUs. In total, we trained 32 distinct configurations (not all of which were ultimately used). A single end-to-end training run with a 7-billion parameter LLM backbone takes 25 hours, meaning roughly 800 GPU hours were used for training. Inference experiments were run on a mixture of RTX3090 24GB cards, A6000 24GB cards and A6000 48GB cards. These required roughly an additional 900 GPU hours. Data analysis utilized CPU. The only sizable compute consisted of applying the DSL estimator to large datasets, which required on the order of $\sim$ 500 CPU hours. Finally, the GPT-4o annotation for the roughly 730k completions in our experiments required roughly USD 2k worth of completion calls.

D. Expert Annotation

Sampling Weights We stratified on evaluation benchmark (i.e. we weighted the probability by the inverse proportion of the originating benchmark to the full dataset) and then upweighted German by a factor of 4, Chinese and Hindi by 2, and downweighted Romanian, Russian and Urdu by a factor of 2. We sampled a total of 1000 observations (without replacement) using these weights.

Annotation Procedure The 1000 observations were uploaded into spreadsheets for the authors to manually annotate. Where possible, annotations were matched to authors who could read the language used in the query. The annotation consisted of three questions: what language is the answer, does the model completion match the gold standard, and is the answer correct. The latter two questions were restricted to three categories: true, false and NA. NA was used where the model did not provide coherent output.

E. Automated Evaluation

GlotLID We use the GlotLID v3 [11] model for automated language identification. We take the most-likely language as predicted by GlotLID, and then manually process the label to collapse what we thought were common misclassifications by the model, such as classifying Mandarin Chinese into various languages and dialects using the simplfied Chinese script when the outputs contained a mix of non-Chinese punctuation characters and Chinese glyphs.

The full parsing rule is as follows:

```
def parse_glotlid(lang: str) -> str:
    iso, script = tuple(lang.split("_"))
    match script:
        case "Hani":
            return "chinese"
        case "Jpan":
            return "japanese"
        case "Deva":
            return "hindi"
        case "Beng":
            return "bengali"
        case "Hebr":
            return "hebrew"
        case "Thai":
            return "thai"
        case "Cyrl":
            return "russian"
        case "Zzzz":
            return "none"
        case "Arab":
            match iso:
                case "urd":
                    return "urdu"
                case _:
                    return "arabic"
        case "Latn":
            match iso:
                case "deu":
                    return "german"
                case "eng":
                    return "english"
                case "spa":
                    return "spanish"
                case "ron":
                    return "romanian"
                case "fra":
                    return "french"
                case _:
                    return "other_latin"
        case _:
            return "other"
```

F. Datasets Used

Here we provide an overview on the datasets we employ in our study.

MaXM was introduced by Changpinyo et al. [6] and is a VQA dataset comprising seven languages in five scripts. In MaXM, the questions and their respective answers are in the same language. Moreover, in MaXM, the images are a subset of the XM3600 [29] dataset and are chosen to match a region where the language of the question-answer pair is spoken. To increase the cultural diversity, the images selected to match the region where the language of the question-answer pair is spoken.

VisIT-Bench stands for **Vis**ual **I**nstruction **T**uning **Bench**mark [3]. The dataset consists of 592 vision-language tasks written by human researchers, with GPT-4-generated responses and dense instruction-conditioned captions of the image that are rated by human coders. The 562 images are taken from the OpenImages [14] v7 dataset. In this work we use 525 examples where the GPT-4 generated responses are rated as correct by human annotators. We machine translate these examples into Arabic, Bengali, Chinese, German, Hebrew, Hindi, Japanese, Spanish and Thai using the Azure Translation API.

To check the translation quality, we inspected 25 randomly sampled translations in Chinese, Hindi, Hebrew, German, Japanese and Spanish (languages where the authors had access to native speakers). Among these, the majority (19 out of 25) of translations were deemed to not significantly change the meaning of the original. In the remainder, issues observed included omitting details (such as not mentioning an object or descriptor), or constructing words that were understandable but not "natural" in the target language. In general the question/instruction was correctly translated, but the translation of the gold standard varied in quality. This presents a limitation for this research, but one that cannot be overcome without greater resources for expert/higher quality translation.

PALO-LLaVA-Bench-In-The-Wild dataset is a multilingual VQA dataset created by the PALO authors [17] by machine translating the original LLaVA-Bench-In-The-Wild [16] in 10 languages using a fine-tuned GPT-3.5 instance. The dataset comprises of 60 questions per language considering 24 diverse images with a caption describing the visual content.

MultiQ is an evaluation dataset for open-ended question answering covering 137 typologically diverse languages. It is specifically constructed to only contain questions that are simple, factual, and target common knowledge to only test the multilingual capabilities of language models, and no complex reasoning [10].

F.1. Machine Translation of Training Data

As noted in the main body, we machine translate (MT) the LLaVA training data into Chinese and German using the NLLB-1.7-distilled model [28]. The choice of this model was primarily motivated by resource availability for translating 1.2M texts into two languages.

We apply two automated translation quality checks for the training data based on the MT checks in Qiu et al. [23]. The first is the token-type-ratio (TTR) of each of the languages. A value close to 0 indicates a high degree of repetition, which is an observed pathology of neural MT models. The second is the BLEU score between the source and MT texts. A BLEU score close to 1 indicates the presence of copied English text.

The highest BLEU score for source to target across all translated examples is $1.6e-231$, indicating that copying did not occur. Figure 4 shows the values for the TTR check. We find that in both cases our MT data has a higher cumulative TTR curve than the English data; this indicates less token repetition. It is hard to directly interpret this value, given that baseline TTR should vary between languages, but the lack of an obvious negative pattern is reassuring.

G. Models Used

Here we provide an overview of the models we employed in our study.

OpenAI/CLIP is a jointly optimized vision and text feature extractor trained using large-scale image-caption pairs [24]. CLIP is focused on learning image representations from scratch that are trivially transferable to many downstream tasks without the need for domain specific training.

DINOv2 is a series of image encoders trained on curated data using unsupervised learning [21]. Through an improved training recipe and larger dataset, followed by a distillation process of larger to smaller models, DINOv2 is positioned as a ViT-based general-purpose image encoder that surpasses OpenAI/CLIP on most benchmarks.

LLaVA-v1.5 is a large multimodal model trained end-to-end with visual instruction following [16]. The model combines a vision model — OpenAI/CLIP — with a large language model — Vicuna-v1.5 — achieving impressive visual and language understanding results that were state-of-the-art at its release. In this work we used the 7b and 13b variants of the model.

BakLLaVA is a large multimodal model based on the LLaVA-v1.5 architecture using Mistral-7b as the base LLM [27]. The model utilizes training data from LLaVA-v1.5 as well as additional sources including ShareGPT and private data with a permissive license.

Yi-6b-chat is a large language model trained from scratch on English and Chinese corpora [1]. In this work, we use the 6b variant that has been extended with chat-style training.

Leo-7b-chat is a large language model that extends Llama-2 into German through continued training on a large German corpus [22].

GlotLID v3 is a language identification model that coveres 2102 languages. The data used to train this model was sourced from Wikipedia, news sites, translation corpora, religious text, and storybooks.

NLLB-1.7-distilled is translation model that support direct translation between 200 languages, including many low-resource languages [28]. The datasets used to train NLLB (No Language Left Behind) were sourced from professionally translated sentences in the Wikipedia domain in addition to publicly available translation datasets.

GPT-4o is a commercial large language model provided from OpenAI.

H. Technical Explainers

H.1. Primer on LLaVA

What is LLaVA? Our study analyzes LLaVA, a multimodal model (VLM) that integrates a pretrained vision encoder, denoted as E_V, with a large language model (LLM), using a connecting multilayer perceptron (MLP). The process is defined in two main training stages: pretraining of the MLP and joint finetuning of the MLP with the LLM.

Model Architecture The VLM comprises the following components:

Vision Encoder: The vision encoder E_V processes the visual input X_v to produce a set of embeddings $E_V(X_v)$.

MLP Connector: A connecting MLP, defined as F, transforms the output of E_V into the dimenstionality of the LLM. This transformation is represented as $F(E_V(X_v))$.

LLM: The LLM processes both textual query X_q and the transformed vision embeddings. The combined input to the LLM is given by concatenating the embeddings from the MLP with text embeddings, i.e., $LLM([F(E_V(X_v)); E_L(X_q)])$, where E_L denotes

The VLM is defined as a function that takes an image input X_v and a textual question X_q, and processes these through the vision encoder, MLP connector F, and LLM to produce an output X_a, which is the model's answer to the question based on the visual context. Formally, the VLM can be expressed as:

$$VLM(X_v, X_q) = LLM\left([\mathrm{F}(E_V(X_v)); E_L(X_q)]\right), \quad (6)$$

where $E_V(X_v)$ is the output of the vision encoder for the input image, $\mathrm{F}(E_V(X_v))$ is the transformed visual embedding suitable for the LLM, and $E_L(X_q)$ represents the embedded form of the textual question. The final output X_a

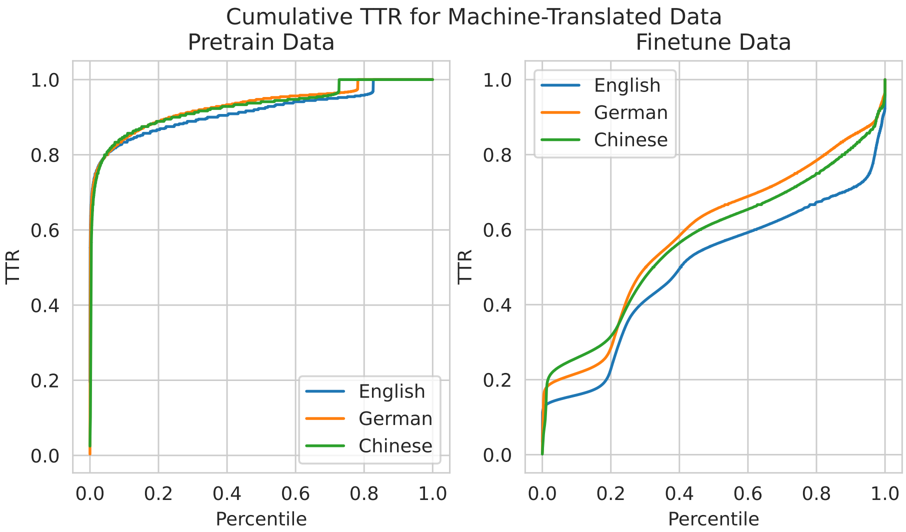

Figure 4. Token-type ratio (TTR) for pretraining (left) and finetuning training datasets.

is generated by the LLM, which synthesizes and integrates both the visual and textual information to produce a contextually appropriate answer.

Training Procedure The training of the VLM is structured into two distinct stages: pretraining and finetuning. During the pretraining stage, the MLP is trained while keeping E_V and the LLM frozen. The objective is to optimize the MLP to map the vision encoder outputs to a representation that is effectively integrable with the LLM. The training uses a custom dataset of 595k samples filtered from CC3M [26]:

$$\mathcal{L}_{\text{MLP}} = \sum_{(X_v, X_c) \in \mathcal{D}} L_{CE}(VLM(X_v, X_q)), \tag{7}$$

where X_c represents the captions associated with X_v, and $\mathcal{D}$ denotes the dataset.

Finetuning In the finetuning stage, the MLP and the LLM are jointly trained with a larger, diverse dataset of 665k multimodal instruction tuning examples, integrating both synthetic and established vision-language training sets. The entire conversation $C = (X_q, X_a)$ is fed into the LLM, with autoregressive masking applied to focus training on the answers using supervised cross-entropy loss L_{CE}:

$$\mathcal{L}_{\text{VLM}} = \sum_{C \in \mathcal{C}} L_{CE}(VLM(X_v, X_q)), \tag{8}$$

where $\mathcal{C}$ represents the conversation dataset, and training focuses exclusively on the answer parts X_a, leveraging the context provided by the entire conversation but training only through the answer segments.

llava7b

dataset	Lang.	IFL	IFL + Remedy	Diff.
llavaw	ar	-0.250	-0.083	0.167
	bn	-0.117	-0.050	0.067
	zh	-0.233	-0.017	0.217
	fr	-0.183	0.000	0.183
	hi	-0.133	-0.033	0.100
	ja	-0.117	-0.050	0.067
	ru	-0.233	-0.017	0.217
	es	-0.200	-0.050	0.150
	ur	-0.050	0.083	0.133
maxm	zh	0.004	0.000	-0.004
	fr	0.004	-0.011	-0.015
	he	-0.132	-0.125	0.007
	hi	-0.042	-0.035	0.008
	ro	0.000	-0.014	-0.014
	th	-0.007	-0.011	-0.004
visitazure	ar	-0.038	-0.047	-0.009
	bn	-0.084	-0.038	0.045
	zh	-0.026	-0.047	-0.021
	de	-0.054	-0.037	0.017
	he	-0.038	-0.037	0.002
	hi	-0.026	-0.009	0.017
	ja	-0.021	-0.051	-0.030
	es	-0.024	-0.042	-0.017
	th	-0.045	-0.010	0.035
average	-	-0.085	-0.030	0.055

Table 6. mechint raw llava7b scores.

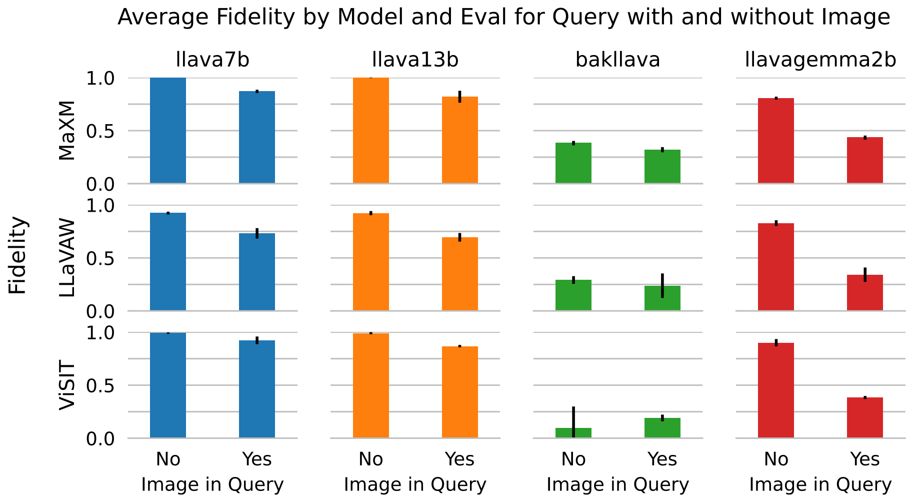

Figure 5. Average fidelity by model and eval for query with and without Images

llava13b

dataset	Lang.	IFL	IFL + Remedy	Diff.
llavaw	ar	-0.183	-0.033	0.150
	bn	-0.233	-0.083	0.150
	zh	-0.133	-0.033	0.100
	fr	-0.200	-0.100	0.100
	hi	-0.317	-0.200	0.117
	ja	-0.183	-0.117	0.067
	ru	-0.433	-0.317	0.117
	es	-0.233	-0.183	0.050
	ur	-0.550	-0.267	0.283
maxm	zh	-0.025	-0.007	0.018
	fr	-0.008	-0.045	-0.038
	he	-0.175	-0.121	0.054
	hi	-0.042	-0.035	0.008
	ro	-0.106	-0.085	0.021
	th	-0.157	-0.093	0.063
visitazure	ar	-0.174	-0.066	0.108
	bn	-0.244	-0.136	0.108
	zh	-0.071	-0.031	0.040
	de	-0.105	-0.094	0.010
	he	-0.125	-0.082	0.044
	hi	-0.136	-0.096	0.040
	ja	-0.056	-0.044	0.012
	es	-0.057	-0.042	0.016
	th	-0.258	-0.155	0.103
average	-	-0.175	-0.103	0.073

Table 7. Mechanistic intervention complete llava13b scores.

bakllava

dataset	Lang.	IFL	IFL + Remedy	Diff.
llavaw	ar	0.000	0.350	0.350
	bn	-0.050	0.217	0.267
	zh	-0.033	-0.067	-0.033
	fr	-0.117	0.000	0.117
	hi	0.000	0.050	0.050
	ja	-0.017	-0.067	-0.050
	ru	0.000	0.000	0.000
	es	-0.117	0.217	0.333
	ur	-0.017	0.183	0.200
maxm	zh	-0.018	0.014	0.032
	fr	-0.318	-0.223	0.095
	he	0.000	0.029	0.029
	hi	0.000	0.135	0.135
	ro	-0.567	-0.299	0.268
	th	-0.119	-0.078	0.041
visitazure	ar	-0.010	0.608	0.618
	bn	-0.012	0.557	0.570
	zh	-0.007	0.019	0.026
	de	-0.136	-0.108	0.028
	he	0.000	0.078	0.078
	hi	-0.007	0.113	0.120
	ja	-0.014	0.026	0.040
	es	-0.183	0.291	0.474
	th	-0.007	0.294	0.301
average	-	-0.073	0.098	0.170

Table 8. Mechanistic intervention complete bakllava scores.

llavagemma2b

dataset	Lang.	IFL	IFL + Remedy	Diff.
llavaw	ar	-0.583	-0.533	0.050
	bn	-0.483	-0.483	0.000
	zh	-0.733	-0.567	0.167
	fr	-0.800	-0.433	0.367
	hi	-0.500	-0.317	0.183
	ja	-0.600	-0.650	-0.050
	ru	-0.883	-0.650	0.233
	es	-0.900	-0.700	0.200
	ur	-0.517	-0.483	0.033
maxm	zh	-0.852	-0.762	0.090
	fr	-0.905	-0.652	0.254
	he	-0.768	-0.482	0.286
	hi	-0.731	-0.546	0.185
	ro	-0.810	-0.637	0.173
	th	-0.646	-0.455	0.190
visitazure	ar	-0.718	-0.578	0.139
	bn	-0.483	-0.420	0.063
	zh	-0.672	-0.552	0.120
	de	-0.688	-0.258	0.430
	he	-0.617	-0.280	0.336
	hi	-0.589	-0.375	0.214
	ja	-0.526	-0.509	0.017
	es	-0.793	-0.608	0.185
	th	-0.538	-0.373	0.166
average	-	-0.681	-0.513	0.168

Table 9. Mechanistic intervention complete LLaVA-Gemma-2b scores.

I. Training

I.1. Hyperparameters

All models were trained using the same hyperparameters as the original LLaVA-v1.5-7b model. This training takes place in two stages, as described above.

In the first ("pretraining") stage, we trained with a global batch size of 256 and a learning rate of $1e-3$. In the second ("finetuning") stage, we used a global batch size of 128 and a learning rate of $2e-5$. For both stages we trained for a single epoch, with a warmup ratio of 0.03 and a cosine annealed learning rate scheduler.

I.2. Convergence

In order to ensure comparability across experiments, we trained every model the same amount (one epoch). However, as a hedge against random failures during training, we monitored the training loss curves. All checkpoints saw similar proportional decrease in training loss from their tenth to final training step, ranging from a 38.9% to 65.8% decrease in training loss. Figure 6 shows the loss curves for each model.

J. Accuracy

J.1. Accuracy of Baseline LLaVA Models

The accuracy of the base LLaVA models is not very high for the languages and benchmarks considered. Table 10 provides a breakdown of accuracy by each of the languages in the benchmarks. We see that the 7B and 13B models fail to exceed even 40 and 50 percent accuracy respectively. These results are consistent with concurrent findings in Schneider and Sitaram [25]. We do not see these results as problematic for our research, as we want to emphasize the goals of fidelity and (factual) accuracy as being independently pursuable.

Query Language	**LLaVA 7B**	**LLaVA 13B**
English	0.372	0.392
French	0.340	0.417
Urdu	0.317	0.317
Russian	0.183	0.233
Bengali	0.161	0.222
Spanish	0.099	0.162
Japanese	0.115	0.130
Chinese	0.129	0.160
German	0.107	0.154
Romanian	0.025	0.271
Hindi	0.099	0.128
Thai	0.108	0.091
Arabic	0.080	0.087
Hebrew	0.063	0.080

Table 10. Performance of LLaVA Models Across Different Languages

J.2. Accuracy-Fidelity Trade-off

In addition to the findings in the main body of this paper, further experiments indicate weak evidence in support of there being a trade-off between optimizing accuracy and fidelity. Table 11 provides the Pearson correlation coefficient between accuracy and fidelity for each of the models included in our analysis. We find that for only five out of 26 models is there a significant correlation, with the value ranging from -0.514 to 0.541. We do not find any pattern from these results to suggest a systematic finding for a trade-off.

K. Use of AI Tools

The authors of this paper used Github Co-pilot for coding assistance for this research.

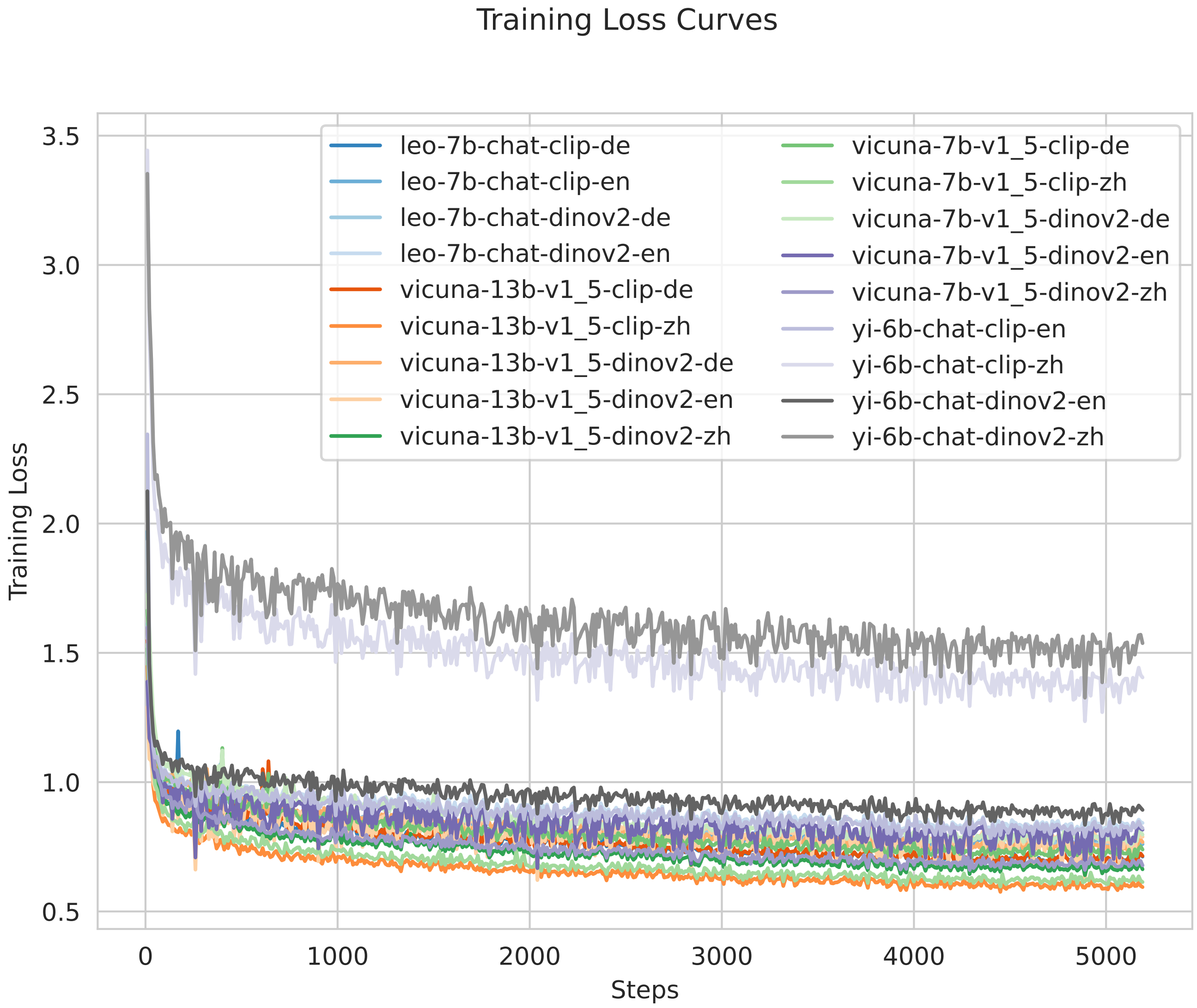

Figure 6. Training loss curves for finetuning. Legend indicates language backbone, vision encoder and training language as two-letter code.

LM	**VE**	**Data**	**Corr**	**p-val**
vicuna13b	dino	en	-0.514	0.006
yi6b	dino	en	-0.432	0.025
yi6b	clip	en	-0.413	0.032
yi6b	dino	en	-0.413	0.032
yi6b	clip	en	-0.386	0.047
leo	dino	en	-0.284	0.151
leo	dino	de	-0.239	0.231
leo	dino	en	-0.218	0.274
vicuna13b	clip	zh	-0.175	0.382
leo	clip	en	-0.152	0.448
leo	clip	de	-0.131	0.515
vicuna13b	clip	de	-0.112	0.579
vicuna13b	dino	zh	-0.060	0.768
yi6b	dino	zh	-0.057	0.776
yi6b	dino	zh	-0.054	0.788
yi6b	clip	zh	-0.050	0.804
vicuna7b	dino	zh	-0.035	0.862
vicuna7b	dino	en	-0.034	0.865
leo7b	clip	en	-0.029	0.885
yi6b	clip	zh	-0.023	0.908
vicuna13b	dino	de	0.023	0.909
vicuna7b	clip	zh	0.069	0.731
vicuna7b	clip	de	0.088	0.664
vicuna7b	dino	de	0.099	0.624
leo	dino	de	0.192	0.338
leo	clip	de	0.225	0.258

Table 11. Correlation between accuracy and fidelity by model.