Enhanced Visual Instruction Tuning for Text-Rich Image Understanding

Anonymous ACL submission

Abstract

Instruction tuning enhances the capability of Large Language Models (LLMs) to interact with humans. Furthermore, recent instruction-following datasets include images as visual input, collecting responses for image-based instructions. However, current visual instruction-tuned models cannot comprehend textual details within images well. This work enhances the current open-source visual instruction tuning models with text-rich images (*e.g.*, movie posters, book covers, etc.). Specifically, we first used publicly available OCR tools to collect results on 422K text-rich images from the LAION dataset. Furthermore, we prompt text-only GPT-4 with recognized text and image captions to generate 16K conversations, each containing question-answer pairs for text-rich images. Using the above-collected data, we substantially improve (up to 20% accuracy improvement) the zero-shot capability of two open-source backbone models on seven datasets (text-based VQA, Information Extraction, ChartQA, etc.). The GPT-4-based instruction-following evaluation also demonstrates the improvement of our model on both natural images and text-rich images. We will make our code/data/models publicly available.

1 Introduction

Instruction tuning (Ouyang et al., 2022; Chung et al., 2022) improves generalization to unseen tasks by formulating various tasks into instructions. Such open-ended question-answering capability fosters the recent chatbot boom since ChatGPT. Recently, visual instruction-tuned models (Liu et al., 2023b; Li et al., 2023a; Li, 2023) further augment conversation agents with visual encoders such as CLIP-ViT (Dosovitskiy et al., 2020; Radford et al., 2021), enabling human-agent interaction based on images. However, possibly due to the dominance of natural images in training data (e.g., Conceptual Captions (Changpinyo et al., 2021) and COCO (Lin et al., 2015)), they struggle with understanding texts within images (Liu et al., 2023d). However, textual understanding is integral to visual perception in everyday life.

Fortunately, tools such as Optical Character Recognition (OCR, Mori et al., 1992) allow us to recognize text in images. One naive way to utilize this is to add recognized texts to the input of visual instruction-tuned models (Gao et al., 2023). However, such an approach significantly increases the computation (longer context lengths) and might not fully leverage the encoding capability of visual encoders. To this end, we propose to enhance the end-to-end visual instruction-tuned model by collecting instruction-following data that require understanding texts within images.

Specifically, we first collect 422K noisy instruction-following data using text-rich[1] images by combining manually written instructions (e.g., "Identify any text visible in the provided image.") and the OCR results. Such large-scale noisy-aligned data effectively improve feature alignment between visual features and the language decoder. Furthermore, we prompt text-only GPT-4 (OpenAI, 2023) with OCR results and image captions to generate 16K conversations, where each conversation can be multiple turns of question & answer pairs, as high-quality instruction-following examples. This process requires GPT-4 to de-noise the OCR results and develop specific questions to create complex instructions based on the input (Figure 1).

We evaluate the effectiveness of the collected data based on two backbones (LLaVA, Liu et al. 2023b and mPLUG-Owl, Ye et al. 2023) and seven datasets (including text-based VQA, Furkan Biten et al. 2019; Mishra et al. 2019; Singh et al. 2019; Mathew et al. 2020, OCR, Risnumawan et al.

[1]In this work, we use the phrase "text-rich images" to describe images with text in them, such as posters and book covers. In contrast, we refer to images without text as "natural images".

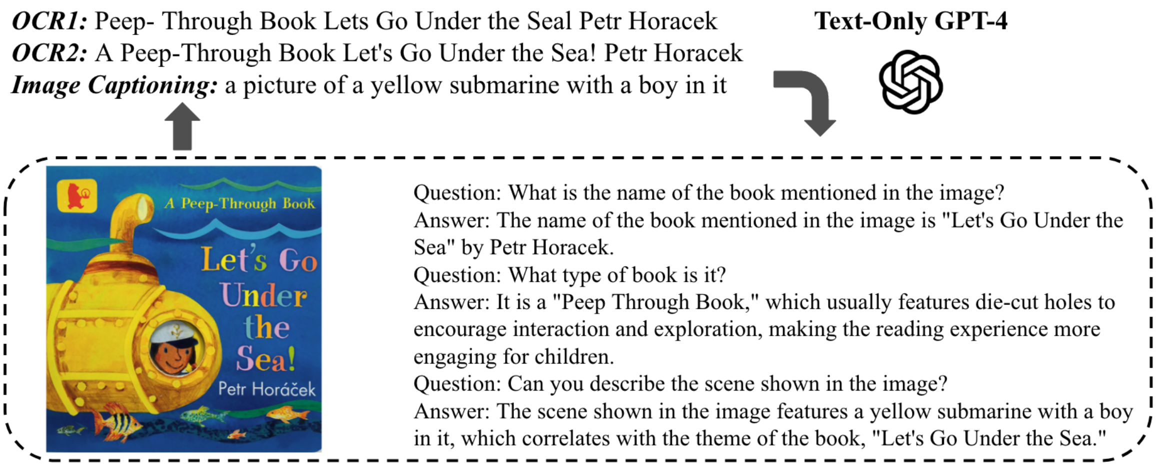


Figure 1: The process of collecting high-quality instruction-following data.

2014a, Information Extraction, Kuang et al. 2023, and ChartQA, Masry et al. 2022) following the evaluation protocol from Liu et al. (2023d). We also demonstrate that our data leads to more significant improvements when the visual encoder accepts higher-resolution images. Moreover, GPT-4-based evaluation favors models trained with our data on following free-style instructions based on images from COCO (Lin et al., 2015; Liu et al., 2023b) and LAION (Schuhmann et al., 2022). To sum up, our contributions are as follows:

- We collected 422K noisy instruction-following data and 16K high-quality instruction-following data to enhance text-rich image understanding.
- Our data significantly enhances the model's performance across seven datasets (text-based VQA, Information Extraction, ChartQA, etc.) and GPT-4 instruction-following evaluations.
- We open source the training and evaluation data together with the model checkpoints[2].

2 Related Work

Instruction Tuning Following natural language instructions is the key capability for an agent to interact with real-world users. Instruction tuning starts from collecting human-preferred feedback for human written instructions (Ouyang et al., 2022) or formulating multi-task training in a multi-task instruction-following manner (Chung et al., 2022; Wang et al., 2022b). However, large, capable instruction-tuned models are usually closed-sourced and serve as commercial APIs only. Recently, Alpaca (Wang et al., 2022a; Taori et al., 2023), Vicuna (Chiang et al., 2023), and Baize (Xu et al., 2023) start the trend of generating high-quality instruction-following data based on LLMs such as GPT-3.5 / ChatGPT / GPT-4 and finetuning the open source LLaMA model (Touvron et al., 2023). However, evaluating the ability to follow instructions remains a challenge. While GPT-4 has demonstrated superior evaluation capabilities (Liu et al., 2023c), there are still a number of concerns, such as bias toward response length (Xu et al., 2023) and lack of robustness to the order of examples (Wang et al., 2023). Following Chiang et al. (2023); Liu et al. (2023b); Dubois et al. (2023), we use GPT-4-based instruction-following evaluation in this work.

Multimodal Instruction Tuning Recently, instruction tuning has been expanded to the multimodal setting, including image, video (Zhang et al., 2023b; Maaz et al., 2023), and audio (Huang et al., 2023; Zhang et al., 2023a). For image-based instruction tuning, MiniGPT-4 (Zhu et al., 2023) employs ChatGPT to curate and improve detailed captions for high-quality instruction-following data. LLaVA (Liu et al., 2023b) generates multimodal instruction-following data by prompting text-only GPT-4 with captions and object's bounding boxes. LLaMA-Adapter (Zhang et al., 2023c; Gao et al., 2023) uses COCO data for text-image feature alignment and utilizes textual data only for instruction tuning. mPLUG-owl (Ye et al., 2023) combines more than 1000M image-text pairs for pretrain-

[2]The release of data and models are subject to OpenAI Terms of Use and the original license of used models.

ing and a 400K mixture of text-only/multimodal instruction-following data for finetuning. However, according to Liu et al. (2023d), most of these models struggle to accomplish tasks requiring OCR capability. InstructBLIP (Dai et al., 2023) transforms 13 vision-language tasks (including OCR-VQA (Mishra et al., 2019)) into the instruction-following format for instruction tuning. Cream (Kim et al., 2023) applies multi-task learning that includes predicting masked texts in images. A more comprehensive survey can be found in Li et al. (2023b). In this work, we select LLaVA as our baseline, which is the most data-efficient and powerful model, and demonstrate the effectiveness of our proposed pipeline.

3 Data Collection

Starting from the LAION-5B (Schuhmann et al., 2022) dataset [3], our goal is only to keep images that are text-rich. Considering that documents usually contain plenty of text, we first obtained a binary classification dataset by combining natural images and document data. Subsequently, we trained an image classifier using a DiT (Li et al., 2022)-base backbone, which was fine-tuned on the RVL-CDIP dataset (Harley et al., 2015). Hopefully, such a classifier can predict whether an image contains text or not. We first build a subset by selecting images with a predicted probability greater than 0.8 while also satisfying $p(\text{watermark}) < 0.8$ and $p(\text{unsafe}) < 0.5$ [4]. The derived subset is noisy due to the limitation of the classifier. To further clean up the data and incorporate human judgment,

We randomly sampled 50K images and clustered them into 100 clusters based on `CLIP-ViT-B/32` visual features. After inspecting the clustering results, we carefully select 14 clusters (see Figure 11 in the Appendix for examples) containing diverse text-rich images ranging from posters, covers, advertisements, infographics, educational materials, and logos. The cluster model is then used as the filter to collect images for constructing our instruction-following examples. As a reference, we provide a CLIP (Radford et al., 2021)-based categorization (see Appendix A for details.) to illustrate the distribution of images for both two types of data we collected in Figure 2. We provide a comparison between our collected data and LLaVA's data in

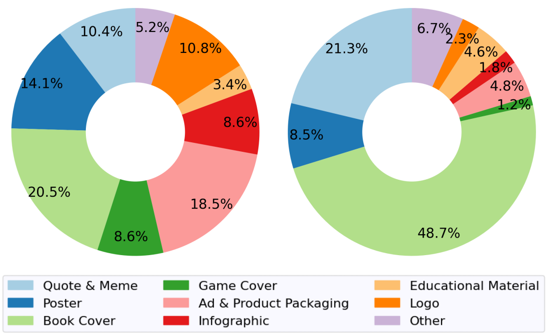


Figure 2: CLIP-based categorization of our collected images. The left refers to images used to collect noisy data, and the right refers to images used in the GPT-4 prompting. Both pie charts are based on 10K sampled images from the corresponding datasets.

Table 1.

Noisy Instruction-following Data Using the clustering model as a filter, we collect 422K deduplicated images that belong to the 14 preferred clusters. To balance the examples from different categories, we keep at most 52K examples for one cluster. We run all images through PaddleOCR [5]. Note that running OCR at the original resolution (e.g.,1024^2) might recognize small fonts that are not visible by visual encoders like CLIP ViT (Dosovitskiy et al., 2020; Radford et al., 2021, resolution up to 336^2). To ensure the recognition of visible fonts while maintaining OCR accuracy, we perform OCR on the image after downsampling (the short edge is resized to 384 pixels if longer than that.) to extract the text. Then, based on the geometric relationships between the recognized words, we merge them into paragraphs before concatenating them. As a robust instruction-following model should react similarly to instructions with similar meanings, we reword "Identify any text visible in the provided image." into ten distinct instructions (Table 5 in Appendix). We then create a single-turn conversation for a given image by **(i)** randomly sampling an ***input instruction*** and **(ii)** using recognized texts as the desired ***output response***. Such instruction-following data is noisy because of the relatively limited performance of OCR tools on diverse fonts and colorful backgrounds.

GPT-4-based Instruction-following Data Compared to high-quality instruction-following data, there are mainly two issues for the noisy data collected above. **(i)** Responses should contain orga-

[3] `https://huggingface.co/datasets/laion/laion-high-resolution`

[4] Both are from the LAION dataset's metadata.

[5] `https://github.com/PaddlePaddle/PaddleOCR`

Data	Image	Instruction	# Conv	Avg Ins Len	Avg Res Len
LLaVA pretraining	CC3M	Caption	595K	15.9	15.4
$R_{pretraining}$ (Ours)	LAION	PaddleOCR	422K	17.2	48.8
LLaVA finetuning	COCO	GPT-4	158K	15.9	93.1
$R_{finetuning}$ (Ours)	LAION	GPT-4	16K	15.1	40.5

Table 1: Compairson between our data and LLaVA data. $R_{pretraining}$ and $R_{finetuning}$ denote the additional pre-training / finetuning data we collected. The average instruction and response length are calculated after LLaMA tokenization.

nized sentences instead of raw OCR results with missing words and grammar errors. **(ii)** Instructions should be diverse, suitable and specific to the given image instead of monotonously asking for all visible texts. To address these issues, we follow Liu et al. (2023b) to generate instruction-following data by prompting text-only GPT-4 (OpenAI, 2023) with OCR results and captions.

It is challenging to prompt GPT-4 with fragmented OCR results in a few words to generate non-trivial instructions. To this end, we carefully select 4 of the 14 previously mentioned clusters (the 3rd, 4th, 6th and 9th clusters in Figure 11) to collect images with enough visible and coherent sentences. As shown in Figure 2, such filtering dramatically increases the percentage of book covers and quote images. We randomly selected 4K examples from each cluster (no overlap with images used for noisy instruction-following data), yielding a total of 16K images. Following prior work (Wang et al., 2022a; Taori et al., 2023; Liu et al., 2023b), we provide the visualization of verb-noun pairs for instructions generated by GPT-4 in Appendix Figure 12. For those instructions without a verb-noun pair, we demonstrate the frequency of objects being asked in Appendix Figure 13.

Furthermore, based on the system message and two in-context few-shot examples (shown in Appendix B), we ask GPT-4 to generate conversational data based on OCR results and image captions (Figure 1). The generated questions are used as ***input instructions***, and answers are used as ***output responses***. Concretely, for a given image, we first provide two OCR results from EasyOCR[6] and PaddleOCR, which can complement each other. To illustrate visual elements other than texts within the image, we also provide the result of BLIP-2 image captioning (Li et al., 2023c). To prevent the caption from focusing on the text, we use OCR bounding boxes to mask the text and then use the inpainting (Telea, 2004) to refill the mask before using generation captions. Note that captioning models might suffer from hallucinations (Rohrbach et al., 2018). We mention this unreliability in our system message and ask GPT-4 only to generate questions with sure answers. We leave the generation of more detailed captions (Rotstein et al., 2023; Hu et al., 2022) for future work.

[6]https://github.com/JaidedAI/EasyOCR

4 Experiments

4.1 Models

We finetune[7] LLaVA and mPLUG-Owl as they have shown promising instruction-following capability compared to other baselines. Please refer to Appendix E for hyperparameters.

LLaVA (Liu et al., 2023b) connects a visual encoder CLIP (Radford et al., 2021) to a language decoder Vicuna-13B (Chiang et al., 2023) by a transformation matrix. Specifically, we use `CLIP-ViT-L/14` for 224^2 resolution and `CLIP-ViT-L/14-336` for 336^2 resolution. The original LLaVA follows a two-stage procedure (See Appendix 5 for details). To augment LLaVA, we combine our noisy instruction-following data with the LLaVA pretraining data and our high-quality instruction-following data with the LLaVA finetuning data. We name the augmented version LLaVA_R.

mPLUG-Owl (Ye et al., 2023) leverages `CLIP-ViT-L/14` and LLaMA-7B (Touvron et al., 2023) while connecting them visual abstractor module. Different from LLaVA, mPLUG-Owl is pretrained on over 1000M captions, including LAION-400M (Schuhmann et al., 2021), COYO-700M (Byeon et al., 2022), Conceptual Captions (Sharma et al., 2018) and MSCOCO (Lin et al.,

[7]Note that the released checkpoints might output unintended harmful and offensive content, which is subject to the base language model and the fact that finetuning might compromise safety (Qi et al., 2023).

2015) and then finetuned on a mixture of language-only instruction-following data (242K) and visual instruction-following data from LLaVA (158K). Due to the magnitude gap, instead of combining our data with the training data of mPLUG-Owl, we further finetune the released checkpoint on our high-quality instruction-following data. We name the augmented version mPLUG-Owl$_\mathrm{R}$.

4.2 Settings

The GPT-4 model refers to the `gpt-4-0314` version, while the cost to collect finetuning data is around $300. The temperature used to sample GPT-4 is set to 1.0 for the generation of training data, 0.7 for the generation of evaluation data, and 0.2 for the evaluation based on GPT-4. All experiments are run on NVIDIA A100 80GB GPUs. During the evaluation, the temperature used to sample from our model is set at 0.9 for text-based VQA, 0.7 for GPT-4-based instruction-following evaluation, and 0.2 for other qualitative demonstrations.

Following the evaluation protocol in Liu et al. (2023d), we evaluate the performance of augmented models on seven datasets: ST-VQA (Furkan Biten et al., 2019), OCR-VQA (Mishra et al., 2019), TextVQA (Singh et al., 2019), and DocVQA (Mathew et al., 2020), CT80 (Risnumawan et al., 2014b), POIE (Singh et al., 2019), ChartQA (Masry et al., 2022) representing various domains (see Appendix C for more details). In the main paper, we use the VQA accuracy as our main metric, while we provide a comparison using more metrics Appendix D. Note that InstructBLIP (Dai et al., 2023) and LLaVA 1.5 (Liu et al., 2023a) includes OCR-VQA in its training sets, making it incomparable with our settings.

4.3 Results

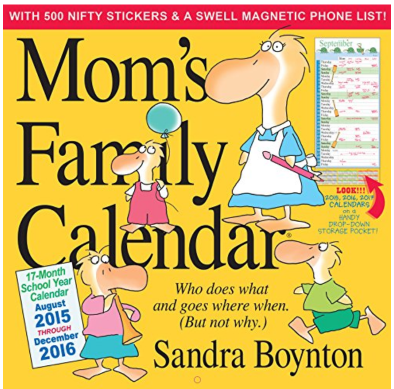
We present the results of the baseline models and our models in Table 2. In 224^2 resolution, LLaVA$_\mathrm{R}$ and mPLUG-Owl$_\mathrm{R}$ substantially improve the LLaVA and mPLUG-Owl baseline, demonstrating that our collected data can bring about a robust improvement. Specifically, the improvement is more significant on LLaVA, making it comparable to mPLUG-Owl, which has a much larger scale of training data. We believe this highlights the effectiveness and data efficiency of collected examples. Furthermore, our best model, 336^2-based LLaVA$_\mathrm{R}$, performs best in 3 out of 4 evaluated datasets. Note that this is not a fair comparison. Some key factors include different language decoders, resolutions, and magnitudes of text-image training data. In the following paragraphs, we conduct ablation studies on our best model, 336^2-based LLaVA$_\mathrm{R}$, by using four datasets in Table 2.

Ablation Study on pretraining/finetuning data We report the result in Table 3 and Figure 3. **(i)** Based on variants (2) and (3), we find that the collected data can benefit the pretraining stage ($\mathrm{R_p}$) and finetuning stage ($\mathrm{R_f}$) separately while being complementary to each other in most cases. More importantly, enhancing the pretraining stage alone achieves the second-best overall performance, indicating the potential to boost textual detail understanding without dependence on GPT-4-generated high-quality data. **(ii)** Using pretraining images, we obtain $\mathrm{C_p}$ by replacing the pretraining instructions with questions & captions, the same pattern as LLaVA. Surprisingly, variant (4) largely improves the (1) baseline, suggesting training on captions of text-rich images also helps. However, as variant (4) is not as good as (2), we can conclude that OCR is more advantageous than captions. **(iii)** We further validate the value of GPT-4 generated data by generating noisy finetuning data ($\mathrm{N_f}$), similar to pretraining data. Variant (5) achieves comparable accuracy as variant (3). However, as shown in Figure 3, such noisy finetuning data hurts the instruction-following capability: (5) responds with all recognized texts while ignoring the questions.

Recognizable Font Size We collect 825 examples from OCR-VQA, which have answers directly presented in the image and are detectable by the OCR tool. This guarantees that answering such questions requires recognizing the answer in the image without complex reasoning. By rescaling the images, we test the model's performance in answering these questions while the vertical heights of answers range from 3 pixels to 19 pixels. We report the result in Fig 4. **(i)** For the baseline model LLaVA, it struggles to provide correct answers in all scenarios for both 224^2-based and 336^2-based versions. **(ii)** Our model LLaVA$_\mathrm{R}$ achieves significantly better results in all scales. We observe a threshold for recognizable texts for both 224^2-based and 336^2-based versions as the accuracy sharply decreases when the height is smaller than 7 pixels. More interestingly, the 224^2-based version performs better on small texts with 3 pixels height, while the 336^2-based version performs better on large texts with more than 7 pixels. We assume the extra training stage of CLIP 336^2 makes it better

	Res.	ST-VQA	OCR-VQA	TextVQA	DocVQA	CT80	POIE	ChartQA
BLIP-2 (2023c) †		21.7	30.7	32.2	4.9	80.9	2.5	7.2
OpenFlamingo (2023) †	224^2	19.3	27.8	29.1	5.1	67.7	2.1	9.1
MiniGPT4 (2023) †		14.0	11.5	18.7	3.0	57.3	1.3	4.3
LLaVA (2023b) ‡	224^2	22.1	11.4	28.9	4.5	61.1	2.1	7.3
LLaVA_R		30.2	23.4	39.5	6.2	81.6	5.7	10.2
mPLUG-Owl (2023) †	224^2	29.3	28.6	40.3	6.9	81.9	3.3	9.5
$\text{mPLUG-Owl}_\text{R}$		29.6	31.2	40.8	7.0	84.7	3.7	10.2
LLaVA (2023b) ‡	336^2	28.9	11.0	36.7	6.9	64.9	2.5	10.2
LLaVA_R		39.2	23.8	48.5	11.6	83.0	8.7	13.5

Table 2: Results (accuracy %) on text-based VQA. We use † to refer to the results obtained from Liu et al. (2023d) and ‡ to refer to our reproduced results. The accuracy metric used by Liu et al. (2023d) only counts for whether the ground truth appears in the response.



Question: Who wrote this book?	
GT	Sandra Boynton
(1)	The book was written by Sandra Boynton.
(2)	The book was written by Sandra Boyton. It is a children's book that features a family calendar with an accompanying activity guide. The book provides information (...) The cover of the book displays a picture of the family from the calendar, which helps to visually represent the topic of the book.
(3)	The book is written by Sandra Byington.
(4)	The book was written by Sandra Boynton.
(5)	With 500+ Nity StickerS &aSwEllMagnetic Phone List! Moms 291573016171800 FAMII (Pocket) Calendar (...) 2016 SANDRA BOYNTON
(6)	The book was written by Sandra Boynton.

Figure 3: Ablation study based an example from OCR-VQA. GT refers to ground truth, and (1) - (6) are different model variants from Table 3. We replace the excessively long response with (...).

	ST-VQA	OCR-VQA	TextVQA	DocVQA
(1) LLaVA	28.9	11.0	36.7	6.9
(2) + R_p	36.7	26.1	46.5	9.6
(3) + R_f	34.1	21.6	43.6	9.5
(4) + C_p	35.4	27.0	45.6	9.2
(5) + N_f	34.1	25.9	43.3	10.2
(6) + R_p + R_f	39.2	23.8	48.5	11.6

Table 3: Ablation Study on pretraining/finetuning data. All results are from 336^2-based models. R_p and R_f denote the extra pretraining/finetuning data we collected. C_p refers to using captions instead of OCR results as responses during pretraining. N_f refers to using written questions + raw OCR results instead of GPT-generated QA for finetuning.

on the larger scale but worse on the smaller scale.

Ablation Study on visual encoders We report the average accuracy on four datasets for models trained with or without our data in Figure 5. Since we have already shown that font size can be a bottleneck to the performance of text-rich image understanding, we further test two visual encoders with higher resolution input: **(i)** `Pix2Struct-base` (Lee et al., 2022) is a visual encoder trained on screenshot to HTML transformation. It supports up to 2048 patches with size 16^2, equivalent to $1024 * 512$. **(ii)** `ConcatCLIP` refers to using 16 `CLIP-ViT-L/14` models to encode the $4 * 4$ grids of images separately and then concatenate the extracted features together, supporting 896^2 resolution. Instead of adding transformed features to the context, we add cross-attention modules to the language decoder to attend to such high-res features. (For more details and results on the high-res encoder, please refer to Appendix E.) We find that using a better visual encoder with higher resolution without changing the data will not improve the performance. However, higher-resolution visual encoders benefit more from our collected data, suggesting our data can be better utilized by higher-resolution visual encoders.

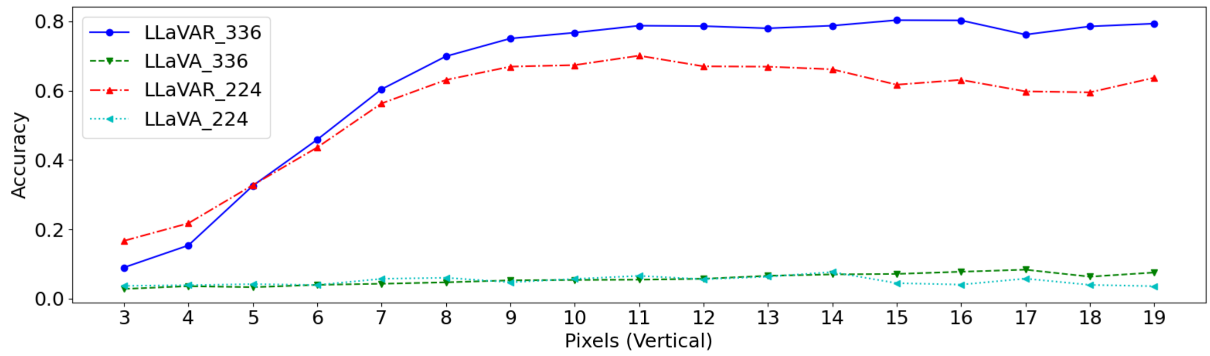


Figure 4: Case study of the recognizable font size, in which the x-axis refers to the height of ground truth answers in the image and the y-axis stands for the answer accuracy of models. We plot the results for both 224^2-based models and 336^2-based models.

	Res	Conversation	Detail	Complex	Read
LLaVA (Original)	224^2	83.1	75.3	96.5	-
LLaVA	336^2	83.9	78.2	95.3	87.9
LLaVA_R	336^2	84.5	78.9	96.5	91.7

Table 4: Relative scores (w.r.t. text-only GPT-4) for instruction-following questions, where the first three dimensions are based on natural images, the last dimension ("Read") is based on text-rich images. In the first row, we show the original results (224^2-based) fetched from Liu et al. (2023b). We report our reproduced LLaVA on 336^2 resolution for a fair comparison.

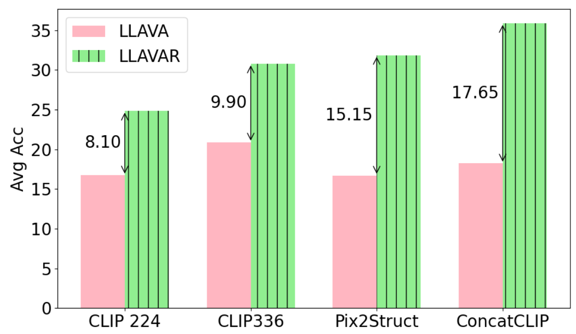
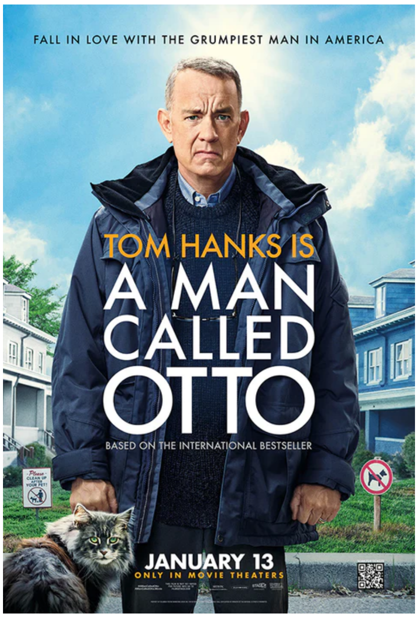


Figure 5: Ablation Study on visual encoders. We report the average accuracy of ST-VQA, OCR-VQA, TextVQA, and DocVQA and show the performance gain of using our data.

4.4 GPT-4-based instruction-following evaluation

We also report the GPT-4 evaluation results on instruction-following questions in Table 4. Specifically, we provide text-only GPT-4 with detailed descriptions of the image (human-written captions, OCR results) and collect feedback as oracles on related questions. To calculate the score, we provide text-only GPT-4 with the detailed description again, together with one question and two answers (one from text-only GPT-4, one from the model we want to test), and ask GPT-4 to give scores to the two answers (from 1 to 10). The final score is the ratio between the average score of the tested model and the average score of GPT-4. **(i) Natural Images**: 90 questions based on 30 COCO validation images from Liu et al. (2023b), including three aspects: conversation, detail description, and complex reasoning. This aims to test whether our collected data will hurt, maintain, or improve the model's performance on natural images. First of all, using a higher resolution brings improvement (+2.9) in the performance of detail description, which is intuitive. Furthermore, LLaVA_R achieves a better trade-off and increases the performance of all three aspects (+1.6 on average). More details are in Appendix G. **(ii) Text-Rich Images**: Similar to collecting the finetuning data, we leverage 50 text-rich images from LAION to collect instruction-following questions based on OCR results and human-annotated captions. We then collect responses from our trained model and use GPT-4 to calculate the relative score w.r.t GPT-4 responses. We add this as an extra dimension "**Read**" to Table 4, where our model demonstrates a significant (+3.8) improvement. The Appendix provides an example in Table 12.



H: When will this movie be released?
L: The movie "The Call of the Ottoman" starring Daniel Craig is expected to be released on February 14th, 2023.
R: The movie will be released on January 13.
H: Who starred in the movie?
L: The movie stars Tom Cruise and Cobie Smulders.
R: Tom Hanks starred in the movie.
H: What is this movie about?
L: The movie "A Man Called Ove" is a comedy-drama film directed by Hannes Holm. (...)
R: The movie is about Tom Hanks, who is called Ottol, and it is an international bestseller based on a true story. (...)
H: Can you describe the given image?
L: The image is a movie poster featuring a man and a cat. The man, who appears to be a character from the movie “The Ugly Truth,” (...)
R: The image features a man standing in front of a house with a cat beside him.
H: Who directed the film?
L: The film was directed by Florian Henckel von Donnersmarck.
R: The film was directed by Tom Hanks, who made his directorial debut with this movie.

Figure 6: Qualitative Comparison. We color the human inputs (***H***) in blue, LLaVA's responses (***L***) in pink, and LLaVA$_R$'s responses (***R***) in green. Both are 336^2-based models. We replace the excessively long response with (...).

4.5 Case Study

We use a recent movie poster [8] to demonstrate the difference between LLaVA and LLaVA$_R$ when interacting with humans based on text-rich images. LLaVA, without augmenting textual understanding within images, suffers from hallucinations when answering these questions. Some mentioned movies, like “A Man Called Ove” and “The Ugly Truth”, are real movies, suggesting that the language decoder is hallucinating its internal knowledge, while the visual encoder cannot encode helpful information. Alternatively, LLaVA$_R$ can correctly answer many of the provided questions with faithful information clearly grounded in the image. However, some limitations remain, such as the spelling error “ottol” (We provide more statistics related to such spelling errors in Appendix F). Also, the final question asks for information that is not observable from the given poster, where an expected response should express such uncertainty instead of giving concrete answers. Nevertheless, neither model correctly answers the question.

4.6 Transferred Instruction-following Capability

According to the dataset statistics (Table 1) and the visualization (Figure 12), our collected instruction-following data is not as diverse and substantial as LLaVA. This can be attributed to the relatively limited information given GPT-4, as the content of text-rich images is less diverse than that of natural images. In Appendix Figure 10, we demonstrate the transferred instruction-following capability of LLaVA, potentially from both the LLaVA data and the Vicuna backbone. While the extra data we add mainly focuses on understanding the visible texts within images, LLaVA$_R$ manages to build its reasoning, writing, and elaboration skills based on the top of its text recognition capability in an end-to-end manner. This allows users to interact with various online content based on simple screenshots.

5 Conclusion

In this work, we enhance visual instruction-tuned models in terms of their capability to read texts in images. Using text-rich images from the LAION dataset, we collect 422K noisy instruction-following examples using OCR results only and 16K high-quality instruction-following data based on text-only GPT-4. These examples are leveraged to augment LLaVA and mPLUG-Owl, where our augmented version demonstrates superior performance in understanding texts within images and following instructions on prior benchmarks. Moreover, our analysis shows that font size is the bottleneck for models to understand texts, while the same augmentation data is more effective with higher-resolution visual encoders. For future work, we encourage further incorporation of the spatial information of OCR results into instruction generation.

[8] https://www.imdb.com/title/tt7405458/

Limitation

From the dataset creation perspective, we collect examples from text-only GPT-4 instead of multimodal models like GPT-4V (OpenAI, 2023) and Gemini (Team et al., 2023). However, we demonstrate that such multimodal capabilities (understanding text within images) can be achieved at a moderate cost with existing human annotation efforts and text-only language models, both of which have many open-sourced alternatives. We believe our data collection and analysis can shed light on building open-sourced multimodal models that are competitive to proprietary models.

References

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *International Conference on Computer Vision (ICCV)*.

Anas Awadalla, Irena Gao, Joshua Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. 2023. Openflamingo.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. 2022. Coyo-700m: Image-text pair dataset. `https://github.com/kakaobrain/coyo-dataset`.

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. 2021. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

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. 2020. An image is worth 16x16 words: Transformers for image recognition at scale.

Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacafarm: A simulation framework for methods that learn from human feedback.

Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marcal Rusinol, Minesh Mathew, C.V. Jawahar, Ernest Valveny, and Dimosthenis Karatzas. 2019. Icdar 2019 competition on scene text visual question answering. *2019 International Conference on Document Analysis and Recognition (ICDAR)*.

Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, Hongsheng Li, and Yu Qiao. 2023. Llama-adapter v2: Parameter-efficient visual instruction model.

Adam W. Harley, Alex Ufkes, and Konstantinos G. Derpanis. 2015. Evaluation of deep convolutional nets for document image classification and retrieval.

Yushi Hu, Hang Hua, Zhengyuan Yang, Weijia Shi, Noah A. Smith, and Jiebo Luo. 2022. Promptcap: Prompt-guided task-aware image captioning.

Rongjie Huang, Mingze Li, Dongchao Yang, Jiatong Shi, Xuankai Chang, Zhenhui Ye, Yuning Wu, Zhiqing Hong, Jia-Bin Huang, Jinglin Liu, Yixiang Ren, Zhou Zhao, and Shinji Watanabe. 2023. Audiogpt: Understanding and generating speech, music, sound, and talking head. *ArXiv*, abs/2304.12995.

Brian Kenji Iwana, Syed Tahseen Raza Rizvi, Sheraz Ahmed, Andreas Dengel, and Seiichi Uchida. 2016. Judging a book by its cover.

Geewook Kim, Hodong Lee, Daehee Kim, Haeji Jung, Sanghee Park, Yoonsik Kim, Sangdoo Yun, Taeho Kil, Bado Lee, and Seunghyun Park. 2023. Cream: Visually-situated natural language understanding with contrastive reading model and frozen large language models.

Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. 2017. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations.

Jianfeng Kuang, Wei Hua, Dingkang Liang, Mingkun Yang, Deqiang Jiang, Bo Ren, and Xiang Bai. 2023. Visual information extraction in the wild: practical dataset and end-to-end solution. In *International Conference on Document Analysis and Recognition*, pages 36–53. Springer.

Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. 2022. Pix2struct: Screenshot parsing as pretraining for visual language understanding.

Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. 2023a. Otter: A multi-modal model with in-context instruction tuning. *arXiv preprint arXiv:2305.03726*.

Chunyuan Li. 2023. Large multimodal models: Notes on cvpr 2023 tutorial. *ArXiv*, abs/2306.14895.

Chunyuan Li, Zhe Gan, Zhengyuan Yang, Jianwei Yang, Linjie Li, Lijuan Wang, and Jianfeng Gao. 2023b. Multimodal foundation models: From specialists to general-purpose assistants.

Junlong Li, Yiheng Xu, Tengchao Lv, Lei Cui, Cha Zhang, and Furu Wei. 2022. Dit: Self-supervised pre-training for document image transformer. *Proceedings of the 30th ACM International Conference on Multimedia*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023c. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft coco: Common objects in context.

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

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

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023c. G-eval: Nlg evaluation using gpt-4 with better human alignment.

Yuliang Liu, Zhang Li, Hongliang Li, Wenwen Yu, Mingxin Huang, Dezhi Peng, Mingyu Liu, Mingrui Chen, Chunyuan Li, Cheng lin Liu, Lianwen Jin, and Xiang Bai. 2023d. On the hidden mystery of ocr in large multimodal models.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. 2023. Video-chatgpt: Towards detailed video understanding via large vision and language models.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. Chartqa: A benchmark for question answering about charts with visual and logical reasoning.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. 2020. Docvqa: A dataset for vqa on document images.

Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. 2019. Ocr-vqa: Visual question answering by reading text in images. In *2019 international conference on document analysis and recognition (ICDAR)*, pages 947–952. IEEE.

S. Mori, C.Y. Suen, and K. Yamamoto. 1992. Historical review of ocr research and development. *Proceedings of the IEEE*, 80(7):1029–1058.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. 2023. Fine-tuning aligned language models compromises safety, even when users do not intend to! *arXiv preprint arXiv:2310.03693*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision.

Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. 2014a. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, 41(18):8027–8048.

Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. 2014b. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, 41(18):8027–8048.

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

Noam Rotstein, David Bensaid, Shaked Brody, Roy Ganz, and Ron Kimmel. 2023. Fusecap: Leveraging large language models to fuse visual data into enriched image captions.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. 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.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

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

Alexandru Telea. 2004. An image inpainting technique based on the fast marching method. *Journal of graphics tools*, 9(1):23–34.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Michael Tschannen, Basil Mustafa, and Neil Houlsby. 2022. Clippo: Image-and-language understanding from pixels only.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2014. Cider: Consensus-based image description evaluation.

Peiyi Wang, Lei Li, Liang Chen, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023. Large language models are not fair evaluators.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022a. Self-instruct: Aligning language models with self-generated instructions.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Maitreya Patel, Kuntal Kumar Pal, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Shailaja Keyur Sampat, Savan Doshi, Siddhartha Mishra, Sujan Reddy, Sumanta Patro, Tanay Dixit, Xudong Shen, Chitta Baral, Yejin Choi, Noah A. Smith, Hannaneh Hajishirzi, and Daniel Khashabi. 2022b. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks.

Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023. Baize: An open-source chat model with parameter-efficient tuning on self-chat data.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, Chenliang Li, Yuanhong Xu, Hehong Chen, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. 2023. mplug-owl: Modularization empowers large language models with multimodality.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023a. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities.

Hang Zhang, Xin Li, and Lidong Bing. 2023b. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*.

Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. 2023c. Llama-adapter: Efficient fine-tuning of language models with zero-init attention.

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

A

CLIP-based categorization Based on the observation of selected clusters, we divide the images used into 8 categories. For each category, we use one or multiple words as labels.

- **Quote & Meme**: "quote", "internet meme".
- **Poster**: "movie poster", "podcast poster", "TV show poster", "event poster", "poster",
- **Book Cover**: "book cover", "magazine cover".
- **Game Cover**: "game cover".
- **Ad & Product Packaging**: "ad", "advertisement", "food packaging", "product packaging".
- **Infographic**: "chart", "bar chart", "pie chart", "scatter plot".
- **Educational Material**: "exam paper", "quiz", "certificate", "book page".
- **Logo**: "logo".

For each word, we use the following templates to achieve embedding-space ensembling (Radford et al., 2021):

- "a photo of a {}."
- "a blurry photo of a {}."
- "a black and white photo of a {}."
- "a low contrast photo of a {}."
- "a high contrast photo of a {}."
- "a bad photo of a {}."
- "a good photo of a {}."
- "a photo of a small {}."
- "a photo of a big {}."

For each image, we calculate the similarity between the image and all words mentioned above using `CLIP-ViT-L/14`. If the highest similarity is less than 0.15, we then classify the image into **Other**, otherwise we classify into the "super class" (e.g., **Poster**) of the word (e.g., "movie poster") with the highest similarity.

B

System Message adapted from Liu et al. (2023b):
You are an AI visual assistant, and you are seeing a single image. What you see is provided with two OCR results and one image caption describing the information within the same image you are looking at. Image captions might include hallucinations, while OCR results are more accurate. Answer all questions with definite answers as you are seeing the image.
Design a conversation between you and a person asking about this photo. The answers should be in a tone that a visual AI assistant is seeing the image and answering the question. Ask diverse questions and give corresponding answers.
Include questions asking about the visual content of the image (e.g., the man, the sunset, the ocean.) and the texts contained in the image. Only include questions that have definite answers:
(1) one can see the content in the image that the question asks about and can answer confidently;
(2) one can determine confidently from the image that it is not in the image. Do not ask any questions that cannot be answered confidently;
(3) DO NOT mention OCR or image caption in your questions and answers;
(4) DO NOT ask about information from captions while it looks unrelated to or contradicts OCR results.
Also include complex questions that are relevant to the content in the image, for example, asking about background knowledge of the texts in the image, asking to discuss about the design of the image, etc. Again, do not ask about uncertain details. Provide detailed answers when answering complex questions. For example, give detailed examples or reasoning steps to make the content more convincing and well-organized. You can include multiple paragraphs if necessary.

Few-shot Examples used for GPT-4 prompting:
Human:
WHAT F YOUR BLESSINGS COME THROUGH RAINDROPS PDf Induded LAURASTOR Y DEVOHONAL BASED ON THE GRAAAMY VARD WIN ISONG mAT HAS INSEIRED MILLION
WHATIFYOUR BLESSINGS COMETHROUGHRAINDROPS PDF Included LAURASTORY A DEVOTIONAL BASEDON THE GRAMMY AWARD WINNI PISONGTHATHASINSPIREDMILLIONSE

Instructions
Identify any text visible in the image provided.
List all the text you can see in the given image.
Enumerate the words or sentences visible in the picture.
Describe any readable text present in the image.
Report any discernible text you see in the image.
Share any legible words or sentences visible in the picture.
Provide a list of texts observed in the provided image.
Note down any readable words or phrases shown in the photo.
Report on any text that can be clearly read in the image.
Mention any discernable and legible text present in the given picture.

Table 5: Ten instructions asking for OCR results.

a girl is standing in a field with a rainbow

GPT:

Question: What is the name of the devotional mentioned in the image?

Answer: The devotional is called “What If Your Blessings Come Through Raindrops” by Laura Story.

Question: What is special about the song mentioned in the image?

Answer: The song is special because it’s a Grammy Award-winning song that has inspired millions of people."

Human:

One of the hardest things in life to accept is a called third strike Robert Frost te

One of the hardest things in life to accept is a called third strike Robert Frost quotefancy

a close up of a baseball glove

GPT:

Question: Why is the third strike hard to accept? Explain the quote to me.

Answer: The quote Öne of the hardest things in life to accept is a called third strikeättributed to Robert Frost is a metaphor for life situations. In baseball, a called third strike is when the umpire determines that a pitch was a strike, but the batter did not swing. This can be frustrating for the batter because they missed an opportunity to hit the ball due to either a lack of confidence or misjudgment.

In life, this metaphor refers to difficult moments where we might miss opportunities or face setbacks because of our own inaction or hesitation. The quote suggests that it is hard to accept these missed opportunities or challenges, especially when they result from our own choices or lack of action.

C

Details of evaluation datasets used in the main paper:

- ST-VQA (Furkan Biten et al., 2019) contains 31791 questions that require understanding the scene text, based on images from COCO (Lin et al., 2015), Visual Genome (Krishna et al., 2016), ImageNet (Deng et al., 2009), etc.

- TextVQA (Singh et al., 2019) contains 45,336 questions that need reading and reasoning about the text in images to answer, based on images from OpenImages (Krasin et al., 2017).

- OCR-VQA (Mishra et al., 2019) contains more than 1 million questions asking about information from book cover images (Iwana et al., 2016).

- DocVQA (Mathew et al., 2020) contains 50000 questions based on document images.

- CT80 (Risnumawan et al., 2014b) contains 80 images for curved text OCR evaluation. The formats of questions are: (1) “What is written in the image?" for English words. (2) “What is the number in the image?" for digit string.

- POIE (Singh et al., 2019) contains 3000 camera images collected from the Nutrition Facts label of products, together with 111,155 text instances. The format of questions is “What is {entity name} in the image?".

- ChartQA (Masry et al., 2022) includes 4,804 charts with 9608 human-written questions.

D

	Res.	METEOR	ROUGE-L	CIDEr
LLaVA	224^2	7.0	8.2	15.3
LLaVA_R		10.0	11.4	24.5
LLaVA	336^2	8.4	9.9	19.1
LLaVA_R		12.8	14.3	30.9

Table 6: Results on ST-VQA using text-matching metrics.

	Res.	METEOR	ROUGE-L	CIDEr
LLaVA	224^2	8.7	10.5	12.2
LLaVA_R		12.5	14.9	21.4
LLaVA	336^2	9.9	12.1	15.3
LLaVA_R		14.8	17.4	27.0

Table 7: Results on textVQA using text-matching metrics.

	Res.	METEOR	ROUGE-L	CIDEr
LLaVA	224^2	0.2	0.1	0.0
LLaVA_R		0.3	0.1	0.0
LLaVA	336^2	0.3	0.1	0.0
LLaVA_R		0.2	0.1	0.0

Table 8: Results on OCR-VQA using text-matching metrics.

Results of other metrics The metric used for text-based VQA in the main paper is the standard practice in VQA benchmarks (Antol et al., 2015). For STVQA and DocVQA, previous works use ANLS (Average Normalized Levenshtein Similarity) as the metric (Furkan Biten et al., 2019; Mathew et al., 2020), which calculates the average normalized edit distance and only works for supervised models trained to output short and precise answers. It works badly for instruction-following models that usually output long sequences instead of brief answers. For reference, we provide more text-matching metrics [9] (METEOR, Banerjee and Lavie, 2005, ROUGE-L, Lin, 2004, CIDEr, Vedantam et al., 2014) to demonstrate the improvement of our model (Table 6, 7, 8, 9), which works well except for OCR-VQA. We assume these metrics are not valuable for OCR-VQA since the ground truth answers are usually too short.

	Res.	METEOR	ROUGE-L	CIDEr
LLaVA	224^2	3.8	4.8	6.3
LLaVA_R		5.6	6.9	12.7
LLaVA	336^2	4.6	5.6	8.7
LLaVA_R		8.6	10.0	21.5

Table 9: Results on DocVQA using text-matching metrics.

E

E.1 LLaVA Architecture and Training

Architecture For the visual encoder V, we use `CLIP-ViT-L/14` for 224^2 resolution and `CLIP-ViT-L/14-336` for 336^2 resolution. The grid features before the last transformer layer are then transformed into the word embedding space of the language decoder through a trainable projection matrix W. We use Vicuna-13B (Chiang et al., 2023), a LLaMA-based (Touvron et al., 2023) instruction-tuned language model, as the language decoder D.

Settings We use the same training hyperparameters as LLaVA[10], except that **(i)** We set the maximum sequence length to 1024 during pre-training. **(ii)** We first pad any given image to a square shape before resizing it to the desired input size, preventing some image content from cropping during preprocessing.

Training We follow the two-stage training design of LLaVA (Figure 7). The training objectives of both stages are the same: generate ***output responses*** (<*res*>) for the ***input instructions*** (<*ins*>). The transformed image tokens (<*img*>) are added before or after the first input instruction. **(i)** During the first pre-training stage, only the projection matrix W is trained for feature alignment. Since the decoder D is frozen, training tolerates noisy data. In the pre-training stage, we combine the 595K pre-training data from LLaVA with our 422K noisy instruction-following data. **(ii)** Both the projection matrix W and the language decoder D are trained during the finetuning stage, where we merge our 16K instruction-following data into the 158K instruction-following data from LLaVA as the training set. Note that the visual encoder is frozen throughout the training period, which might restrict text recognition performance, as CLIP is trained for general-purpose text-image alignment.

[9] `https://github.com/salaniz/pycocoevalcap`
[10] `https://github.com/haotian-liu/LLaVA`

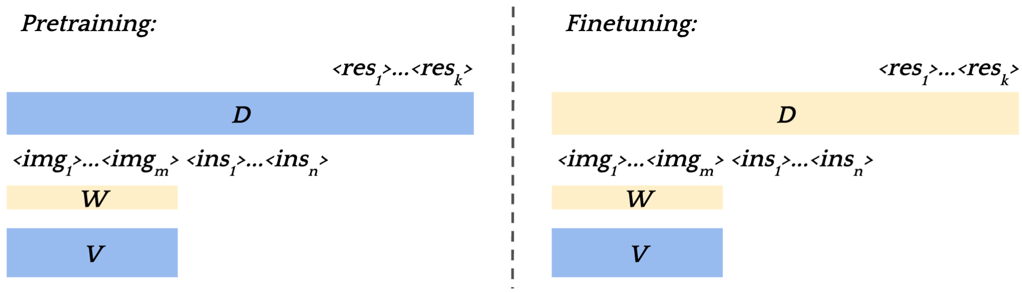


Figure 7: The model training process for the visual encoder V, projection matrix W, and language decoder D. ***Blue blocks*** denote frozen modules and ***yellow blocks*** denote trainable modules. The training input is image tokens (<*img*>) and instruction tokens (<*ins*>), while the target is response tokens (<*res*>).

Better choices of the visual encoder (Tschannen et al., 2022) or CLIP-ViT finetuning (Ye et al., 2023) may further benefit the visual understanding capability, which we leave for future work.

E.2 mPLUG-Owl finetuning

Settings We finetune the mPLUG-Owl checkpoint using batch size 256 and learning rate $1e-5$ for 3 epochs on the collected high-quality instruction-following data.

E.3 High-Resolution LLaVA Architecture and Training

The original version of LLaVA_R only supports up to 336^2 resolution, while our case study has also shown the threshold for the recognizable font size. Both suggest the difficulty of processing real-world high-res images without scaling and cutting. To this end, we test a dual visual encoder system for the high-res variant of LLaVA_R, where a high-res visual encoder is added to work with the standard one. Ideally, the standard visual encoder extracts general, high-level information, while the high-res one specifically helps with detailed information.

Architecture A high-res visual encoder usually outputs thousands of visual features. Simply following LLaVA to feed the transformed visual features into the context of the language decoder is impractical, as the maximum sequence length of the language decoder is usually 2048/4096. To this end, we propose handling high-res visual features with cross-attention modules and standard visual features by feature transformation. We depict the proposed system in Figure 8.

Specifically, given a standard visual encoder V_1, the extracted features are transformed into the word embedding space of the language decoder through a trainable projection matrix W. These transformed features are then concatenated with the word embeddings to build the input embeddings of the language decoder D as Equation 1, where I is the input image, V_1 denotes extracting the grid features before the last transformer layer.

At the same time, we use the high-res visual encoder V_2 to extract high-res visual features, which are then transformed into keys/values as the inputs of the cross-attention module in transformer layers. Given h^j as the hidden state before the cross-attention module in layer j, we calculate the cross-attention using Equation 2, where Q^j, K^j, V^j denotes the query/key/value projection matrix in the j-th transformers layer. In practice, there is a pre-attention LayerNorm before calculating the attention and another output projection matrix O^j to project the aggregated values back to the hidden space.

As the pretrained language decoder D might only have self-attention modules, we manually add another cross-attention module after the original self-attention module in every transformer layer. Considering the random initialization of cross-attention modules might hurt the original language generation capability, we initialize the value projection matrix V^j as a zero matrix and the output projection matrix O^j as an identity matrix.

Implementation We use `CLIP-ViT-L/14` as the standard visual encoder. For the high-resolution encoder, we test two models: **(i)** `Pix2Struct-base` (Lee et al., 2022) is a visual encoder trained on screenshot to HTML transformation. It supports up to 2048 patches with size 16^2, equivalent to $1024 * 512$. **(ii)** `ConcatCLIP` refers to using 16 `CLIP-ViT-L/14` models to encode the $4 * 4$ grids of images separately and then concatenate the extracted features together. In other words, it supports 896^2 resolution.

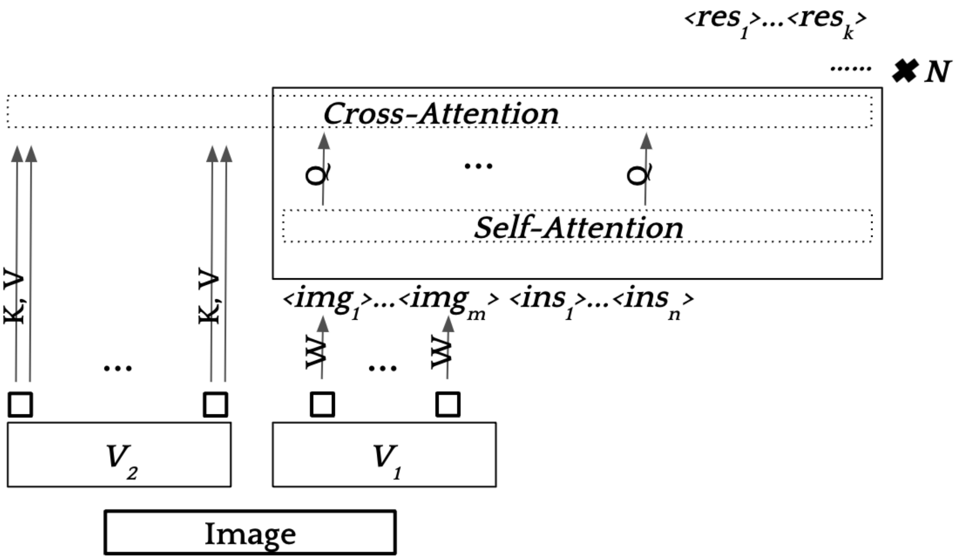


Figure 8: Illustration of the dual visual encoder system. Given an image, it is simultaneously processed by visual encoders V_1 and V_2. V_1 features are transformed by transformation matrix W and directly used as input embeddings to the language model. For V_2 features, they are transformed by transformation matrix K and V and used as keys and values to calculate the cross attention in every transformer layer (assume there are N layers), which uses the transformed hidden states (through Q) from the self-attention module as queries. For the language decoder D, the input is image tokens (*<img>*) and instruction tokens (*<ins>*), while the target is response tokens (*<res>*).

$$\begin{aligned} \mathrm{emb}(\langle \mathrm{img}_1 \rangle), \cdots, \mathrm{emb}(\langle \mathrm{img}_m \rangle) &= WV_1(I) \\ \mathrm{input\_emb} &= \mathbf{concat}([\mathrm{emb}(\langle \mathrm{img}_1 \rangle), \cdots, \mathrm{emb}(\langle \mathrm{img}_m \rangle), \mathrm{emb}(\langle \mathrm{ins}_1 \rangle), \cdots, \mathrm{emb}(\langle \mathrm{ins}_n \rangle)]) \end{aligned} \tag{1}$$

$$\mathrm{CrossAttention}(h, V_2, I) = \mathrm{softmax}(\frac{Q^j h^j (K^j V_2(I))^T}{\sqrt{d}}) V^j V_2(I) \tag{2}$$

Training Only cross-attention modules and the projection matrix W are trained during pretraining, while visual encoders and the language decoder are frozen. Cross-attention modules, the projection matrix W, and the language decoder D are trained during finetuning.

Data To fully unlock the potential of the augmented visual encoder, we also double the number of pretraining examples using the same criteria mentioned in Section 3.

F

The impact of OCR spelling errors We study such OCR errors by studying 1673 examples from OCR-VQA, which have ground truth answers with more than ten characters. We (i) define "correct" as the ground truth answers that are exactly in the predictions, and (ii) define "partially correct" as there exists a substring in the prediction that has high enough similarity with the ground truth but not the same. Specifically, we look at all substrings with the same length of the ground truth in the prediction to calculate ANLS (Average Normalized Levenshtein Similarity) and regard the prediction as "partially correct" if the highest ANLS is greater or equal to 0.5 but smaller than 1.

	Res.	**Correct %**	**Partially Correct%**
LLaVA	224^2	1.6%	8.7%
LLaVA$_R$		6.8%	22.8%
LLaVA	336^2	2.2%	11.2%
LLaVA$_R$		9.0%	26.8%

Table 10: Statistics of correct answers and partially correct answers on OCR-VQA.

We find that many predictions can be considered partially correct, indicating the actual performance of tested models is better than the reported accuracy numbers. However, the percentage of partially

	Conversation	Detail	Complex
LLaVA	83.6	78.1	95.2
LLaVA + $R_{pretrain}$	86.7	79.3	95.1
LLaVA + $R_{finetune}$	79.0	79.4	98.0
$LLaVA_R$	84.4	78.9	96.6

Table 11: Ablation Study on GPT-4-based Instruction-following Evaluation from LLaVA (Liu et al., 2023b).

correct predictions is highly correlated with the percentage of correct predictions. Therefore, we believe that the current metrics can effectively compare the performance of different models.

G

Ablation Study on Instruction-following Evaluation To provide a more detailed understanding of our data, we provide the ablation study results on 90 instruction-following questions from LLaVA (Liu et al., 2023b). We find that including pretraining data improves the conversation capability, probably because longer training data leads to generating longer responses (Table 1), which is favored by GPT-4 (Xu et al., 2023). On the other hand, including only finetuning data hurts the conversation capability but increases complex reasoning. Combining pretraining and finetuning data improves the trade-off between conversation and complex reasoning. Such results prove that incorporating our data will at least not harm the performance of interacting with natural images.

Question: Based on the title and the image on the cover, what can be inferred about the content of "Bo's Lasting Lessons" and its potential target audience?		Relative Score
GPT-4 (Text-Only)	From the title and image, it can be inferred that "Bo's Lasting Lessons" is a book about leadership lessons taught by a legendary coach, likely in the context of sports. The image of men running suggests that the book might be focused on a specific sport, such as football or track and field. The target audience could be individuals interested in coaching, sports, or leadership in general, looking to learn from the experiences and teachings of a successful coach.	-
LLaVA	The book "Bo's Lasting Lessons: The Timeless Teachings of Bo Schembechler" is written by Bo Schembechler and Jim Harshaw. Given the title and the image on the cover, which shows a football coach running across the field with his players in pursuit, it can be inferred that the content of the book is likely focused on football and the strategies, techniques, or principles that can help players improve their performance and achieve success in the sport. The potential target audience for this book could be football players, coaches, and fans who are interested in learning more about the game and the experiences and insights of a seasoned coach like Bo Schembechler.	75%
LLaVA_{R}	"Bo's Lasting Lessons" appears to be a book that shares the legacy coach Bo Schembechler's teachings on leadership and timeless fundamentals. The image on the cover, featuring a football coach leading his team, further supports this inference. The book might be targeted towards sports enthusiasts, particularly fans of football, and people interested in learning about leadership principles and fundamentals from a renowned coach's perspective. The audiobook, read by John H. Mayer and featuring a foreword and conversation with John U. Bacon, suggests that it could also appeal to audiences who prefer listening to books or want to gain more insights from the author and other experts.	129%

Table 12: An example for our collected evaluation data, corresponding to the "**Read**" dimension of GPT-4-based instruction-following evaluation. The corresponding image is Figure 9 in Appendix.

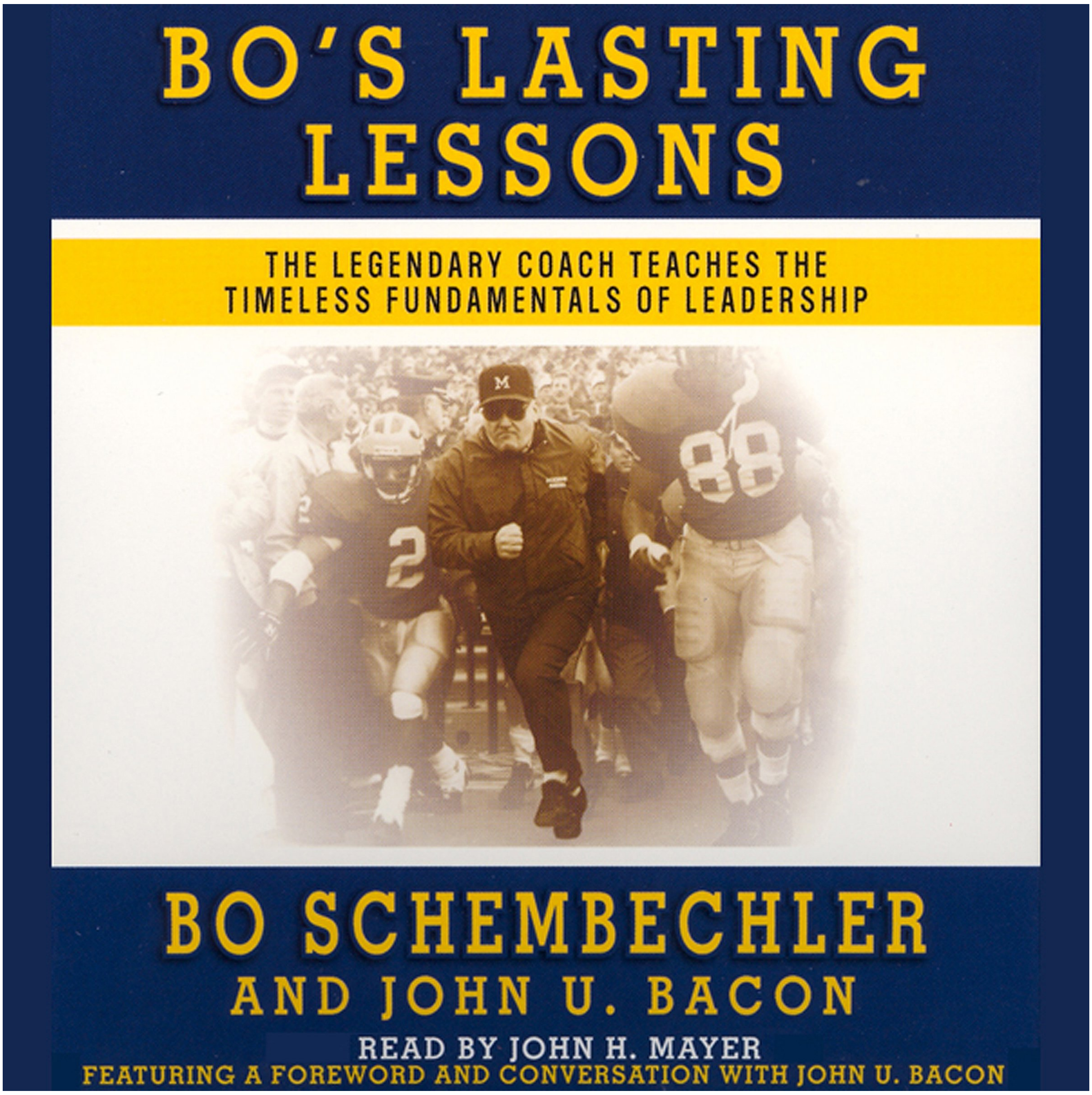
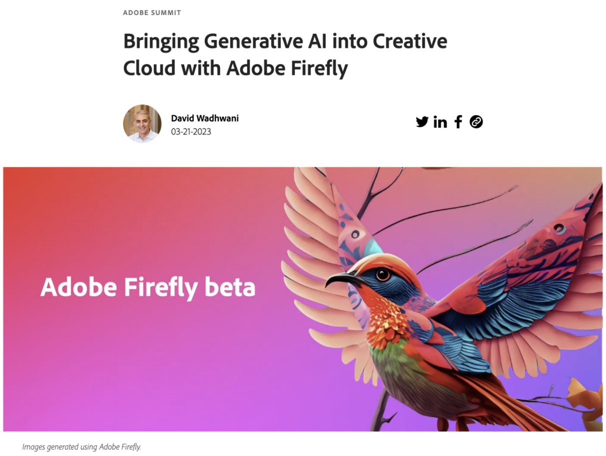


Figure 9: An example for the Read dimension of GPT-4-based instruction-following evaluation.

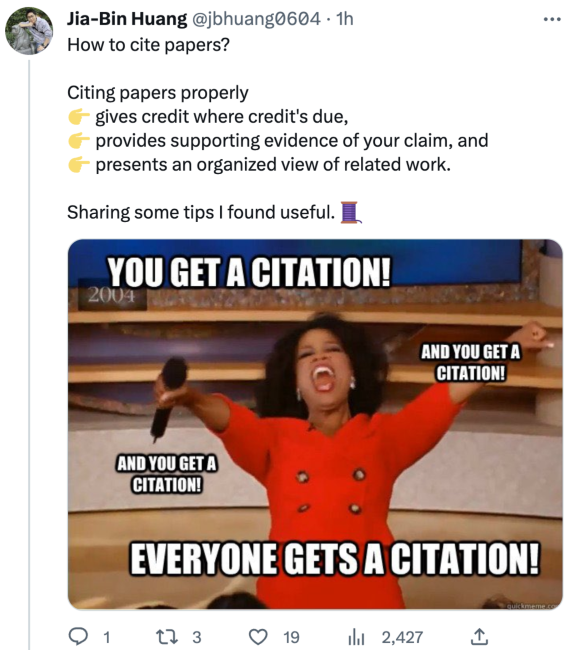
If I want to study optimization, is Georgia Tech a good place for me?

Yes, Georgia Tech is a good place for you if you want to study optimization. The image mentions that Georgia Tech's H. Milton Stewart School of Industrial and Systems Engineering is ranked as the best in the nation according to the U.S. News & World Report for 33 consecutive years. This indicates that the school has a strong reputation and offers quality education in the field of optimization and industrial engineering.



Pretend you are the author, write the next paragraph of this article.

Adobe Firefly is an exciting new tool that allows users to bring generative AI into their creative workflow. By integrating Firefly with Adobe's popular Creative Cloud, users can unlock new levels of creativity and automation, streamlining their design processes and enabling them to explore more innovative ideas. With Firefly, designers can explore data-driven design, real-time collaborative editing, and seamless integration with other Adobe products, making the entire creative process more efficient and intuitive.



What is this tweet about?

This tweet is about sharing tips on how to cite papers properly, giving credit where credit is due, providing support for evidence of your claim, and presenting an organized view of related work. It also humorously emphasizes the importance of citations by mentioning that "everyone gets a citation" in a playful manner.

By the way, who is the woman in this meme? Any background information?

The woman in the meme is Oprah Winfrey, a famous television host, author, and philanthropist. She is known for her influential book club and her ability to make people feel good about themselves. In this meme, she is seen enthusiastically endorsing the concept of citing sources in academic work.

Figure 10: Transferred instruction-following capability of LLaVA_R.

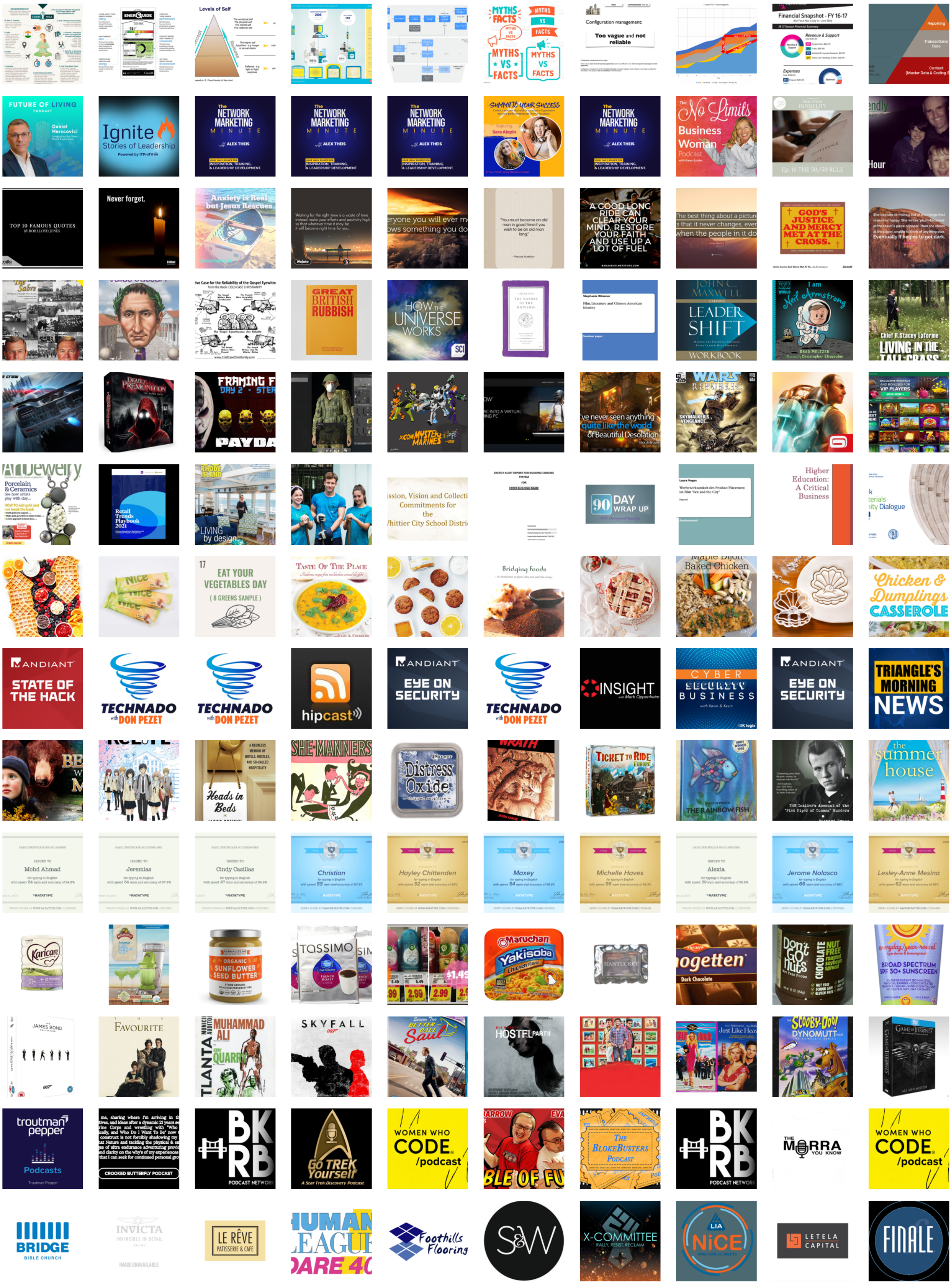


Figure 11: All 14 clusters we selected as text-rich images. Each row corresponds to one cluster, where we show ten randomly sampled examples before de-duplication.

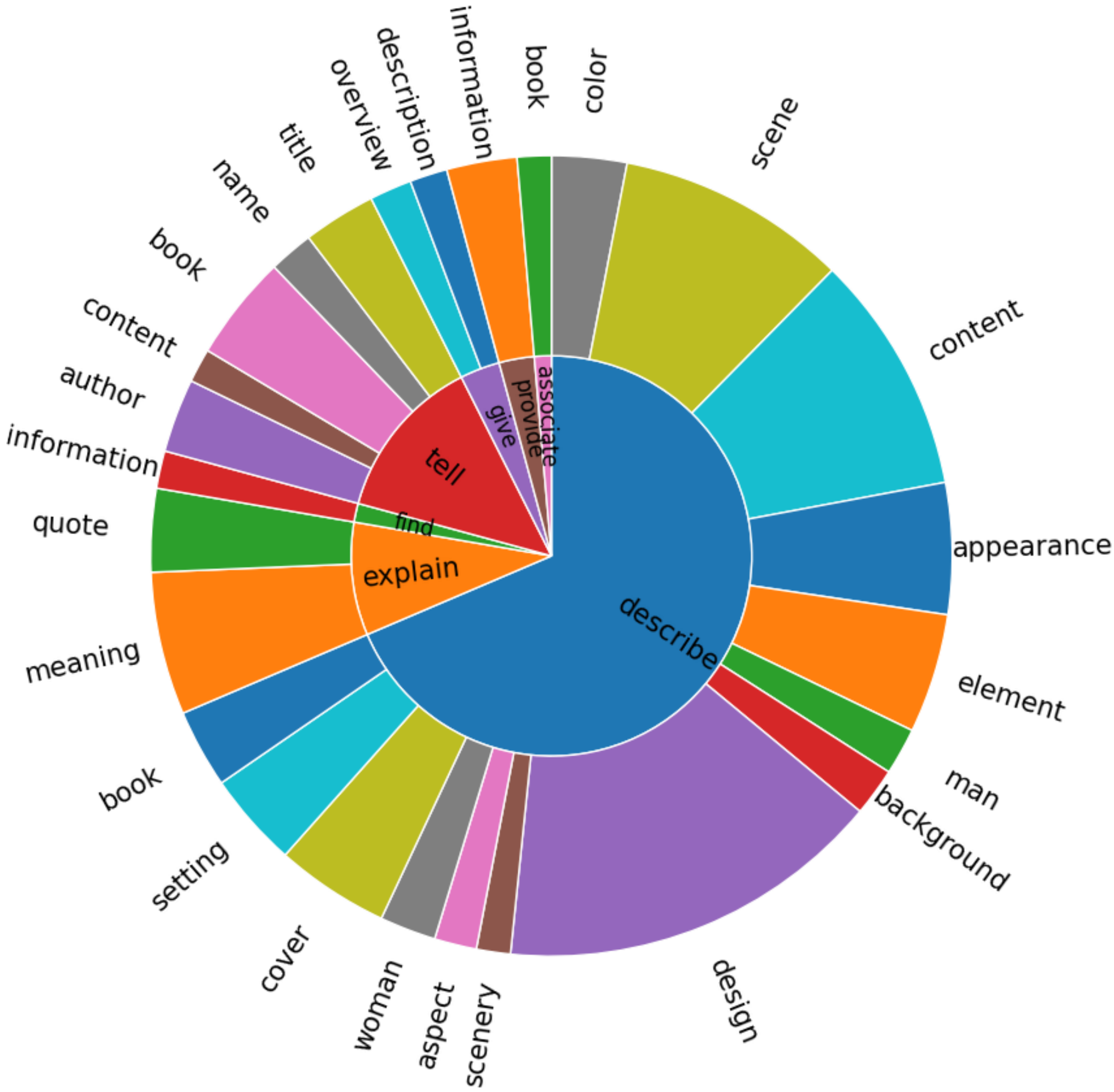


Figure 12: Visualization of collected instructions.

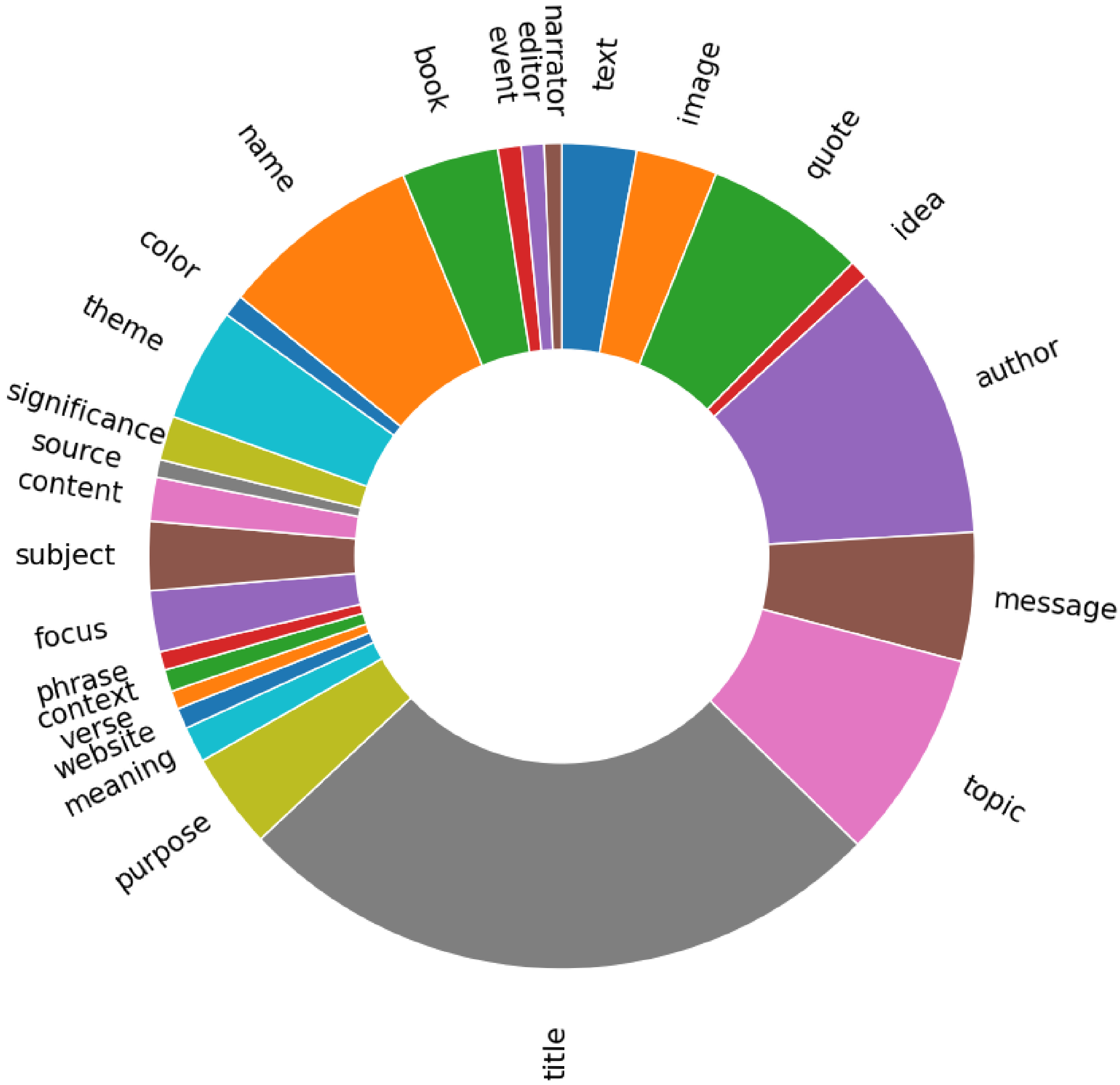


Figure 13: Visualization of collected instructions.