Steerable Visual Intelligence

by

Haotian Liu

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

(Computer Sciences)

at the

UNIVERSITY OF WISCONSIN–MADISON

2024

Date of final oral examination: 04/22/2024

The dissertation is approved by the following members of the Final Oral Committee:
Yong Jae Lee, Associate Professor, Computer Sciences
Frederic Sala, Assistant Professor, Computer Sciences
Kangwook Lee, Assistant Professor, Electrical & Computer Engineering
Junjie Hu, Assistant Professor, Biostatistics & Medical Informatics

© Copyright by Haotian Liu 2024
All Rights Reserved

To my family.

ACKNOWLEDGMENTS

I want to sincerely thank my advisor, Dr. Yong Jae Lee, for his constant support during my PhD journey. His knowledge, valuable suggestions, and dedication to high academic standards have greatly influenced my work and inspired a deep interest in learning.

I also want to express my gratitude to the other members of my dissertation committee: Dr. Frederic Sala, Dr. Kangwook Lee, and Dr. Junjie Hu. Their insightful feedback has significantly enhanced the quality and depth of my study.

I'm truly grateful for the opportunity to work alongside my lab mates at UC-Davis, where I began my PhD journey: Dr. Krishna Kumar Singh, Dr. Fanyi Xiao, Dr. Maheen Rashid, Rafael A. Rivera Soto, Xueyan Zou, Utkarsh Ojha, Dr. Yuheng Li, Chong Zhou, and Yangming Wen. The collaboration and shared moments with each one of them have deeply enriched both my personal and academic life. Additionally, I'm thankful for meeting new lab mates at UW-Madison who introduced me to a new chapter of my PhD life: Zhuoran Yu, Mu Cai, Zeyi Huang, Thao Nguyen, and Anirudh Sundara Rajan. Your companionship has provided me with memorable experiences and has greatly enriched my life.

I'm also thankful for the opportunity to collaborate with other talented individuals not mentioned earlier: Dr. Sheng Shen, Bo Li, Yuanhan Zhang, Qingyang Wu, Dr. Fangzhou Mu, Dr. Jianwei Yang, Dr. Jianfeng Gao, and Dr. Chunyuan Li. Their insightful discussions and support have continually inspired my research.

Special appreciation goes to my roommates: Yuheng Li, Youzhi Tian, Haitian Chen, as well as to my many friends from the gym and daily life. I'm grateful for the time and effort you've invested in helping me tackle challenges. Your support has been a crucial pillar of strength and happiness for me.

Lastly, I extend my deepest gratitude to my wife, my parents, and all family members for their unmatched support, understanding, and encouragement throughout my PhD studies. Their love and reassurance have given me the strength to face challenges and pursue my academic ambitions with determination.

CONTENTS

Contents iv

List of Tables vi

List of Figures x

Abstract xii

1 Introduction 1

2 Learning Customized Visual Models with Retrieval-Augmented Knowledge 5
  2.1 Introduction 5
  2.2 Related Work 9
  2.3 Retrieval-Augmented Customization 11
  2.4 Experiments 17
  2.5 Conclusion 27

3 Visual Instruction Tuning (LLaVA) 28
  3.1 Introduction 28
  3.2 Related Work 30
  3.3 GPT-assisted Visual Instruction Data Generation 32
  3.4 Visual Instruction Tuning 35
  3.5 Experiments 38
  3.6 Conclusion 46

4 Improved Baselines with Visual Instruction Tuning 48
  4.1 Introduction 48
  4.2 Related Work 52
  4.3 Approach 53

4.4 Empirical Evaluation 60
4.5 Open Problems in LMMs 67
4.6 Conclusion 69
4.7 LLaVA-NeXT: Improved reasoning, OCR, and world knowledge 70

5 Conclusion and Discussion 73

References 75

LIST OF TABLES

2.1 Comparison of zero-shot task transfer with public checkpoints from CLIP [112] and OpenCLIP [55]. LAION [119, 118] is abbreviated as "L" in the table. Web-800M†: a privately collected web database with 800M image-text pairs. By continue pretraining on only ~10M retrieved data, REACT outperforms *all* public CLIP/OpenCLIP checkpoints. 20
2.2 The average scores of image classification performance on 20 datasets in ELEVATER. REACT consistently outperforms CLIP in both data-limited and data-rich regimes. 21
2.3 Image-text retrieval results on Flickr30K [110] and MSCOCO [83] datasets. CLIP†, Bletchley†: our evaluation. 24
2.4 Zero-shot and open-vocabulary object detection results on MSCOCO [83] dataset using RegionCLIP [170] pipeline. 24
2.5 Zero-shot and annotation-free semantic segmentation results on COCO Stuff [83] using MaskCLIP [171] (ViT-B/16). 26
2.6 Ablation: tuning strategy. For our model customization purpose, we advocate locked-text (gated-image) tuning methods in gray rows. ✓: trainable, ✗: locked. 27

3.1 One example to illustrate the instruction-following data. The top block shows the contexts such as captions and boxes used to prompt GPT, and the bottom block shows the three types of responses. Note that the visual image is not used to prompt GPT, we only show it here as a reference. 33

3.2 The input sequence used to train the model. Only two conversation turns are illustrated here; in practice, the number of turns varies based on the instruction-following data. In our current implementation, we follow Vicuna-v0 [25] to set the system message $\mathbf{X}_{\texttt{system-message}}$ and we set `<STOP>` = `###`. The model is trained to predict the assistant answers and where to stop, and thus only green sequence/tokens are used to compute the loss in the auto-regressive model. 37
3.3 Example prompt from GPT-4 paper [104] to compare visual reasoning and chat capabilities. Compared to BLIP-2 [77] and OpenFlamingo [6], LLaVA accurately follows the user's instructions, instead of simply describing the scene. LLaVA offers a more comprehensive response than GPT-4. Even when merely asked to describe the image, LLaVA identifies atypical aspects of the image. 39
3.4 Ablation on LLaVA-Bench (COCO) with different training data. We report relative scores *w.r.t.* a text-only GPT-4 model that uses ground truth image captions and bounding boxes as visual input. We prompt GPT-4 with the answers from our model outputs and the answers by GPT-4 (text-only), and let it compare between both responses and give a rating with an explanation. 41
3.5 Instruction-following capability comparison using relative scores on LLaVA-Bench (In-the-Wild). The results are reported in the format of *mean* ± *std*. For the first three rows, we report three inference runs. LLaVA performs significantly better than others. † For a given set of LLaVA decoding sequences, we evaluate by querying GPT-4 three times; GPT-4 gives a consistent evaluation. 41

3.6 Challenging examples from LLaVA-Bench (In-the-Wild), we provide extremely-detailed annotation for each image for an accurate evaluation. Some questions require the model to extract details from high resolution image and to have a broad knowledge coverage. 43
3.7 Accuracy (%) on Science QA dataset. Question categories: NAT = natural science, SOC = social science, LAN = language science, TXT = text context, IMG = image context, NO = no context, G1-6 = grades 1-6, G7-12 = grades 7-12. †Text-only GPT-4, our eval. Our novel model ensembling with the text-only GPT-4 consistently improves the model's performance under all categories, setting the new SoTA performance. . . . 45
3.8 Design choice ablations (%). The difference with the best variant is reported in red text. 46

4.1 Visual input example to illustrate the challenge of (a) multitask balancing and (b) different format prompts. The same image input is used. 55
4.2 **Scaling results** on 🟦 data, 🟥 model, and 🟧 resolution. We choose to conduct experiments on GQA [53], MME [36], and MM-Vet [157] to examine the representative capabilities of VQA with short answers, VQA with output formatting, and natural visual conversations, respectively. *Training images of GQA were observed during training. 56
4.3 **Comparison with SoTA methods on academic-task-oriented datasets.** LLaVA-1.5 achieves the best performance on 4/5 benchmarks, and ranks the second on the other. *The training images/annotations of the datasets are observed during training. †Includes in-house data that is not publicly accessible. 58

4.4 **Comparison with SoTA methods on benchmarks for instruction-following LMMs.** LLaVA-1.5 achieves the best overall performance. 59
4.5 LLaVA-1.5 can detect and answer tricky questions when prompted to verify the question. 61
4.6 LLaVA-1.5 can extract information from the image and answer following the required format, despite a few errors compared with GPT-4V. GPT-4V results are obtained from [147]. 63

LIST OF FIGURES

2.1 React achieves the best zero-shot ImageNet performance among public checkpoints (Top Left), achieves new SoTA on semi-supervised ImageNet classification in the 1% labeled data setting (Top Right), and consistently transfer better than CLIP on across a variety of tasks, including ImageNet classification, zero/few/full-shot classification on 20 datasets in ELEVATER benchmark, image-text retrieval, object detection and segmentation (Bottom). Please see the detailed numbers and settings in the experimental section. For the left figure, circle size indicates model size. 7

2.2 Illustration of the proposed React framework. 12

2.3 Illustrative comparisons across different model tuning methods. (a) and (b) are existing baseline tuning methods. For model customization in a target domain, we found that (c) and (d) work better. One layer of the proposed modularized image encoder in locked-text gated-image tuning is illustrated in right side. 13

2.4 Zero-shot comparison on Elevater ICinW 20 datasets. React (B32) improves over the base checkpoints on most datasets. . 18

2.5 Success and failure cases in Elevater benchmark. We show class name and the caption of the first retrieved image-text pairs, others are similar and omitted due to limited space. . . . 22

3.1 LLaVA network architecture. 35

4.1 **LLaVA-1.5** achieves SoTA on a broad range of 11 tasks (Top), with high training sample efficiency (Left) and simple modifications to LLaVA (Right): an MLP connector and including academic-task-oriented data with response formatting prompts. 49

4.2 **LLaVA-1.5-HD.** Scaling LLaVA-1.5 to higher resolutions by splitting the image into grids and encoding them independently. This allows the model to scale to any resolution, without performing positional embedding interpolation for ViTs. We additionally concatenate the feature of a downsampled image to provide the LLM with a global context. 56
4.3 **Ablation on LLM choices**. Data points represent the relative performance of the best performing variant for each dataset. . 65
4.4 **Ablation on data efficiency.** Data points represent the relative performance of the best performing variant for each dataset. . 66

ABSTRACT

Understanding and reasoning about the visual world based on human instructions has long been a challenging problem. With the advancements in deep learning and computer vision, machine learning models have learned to tackle challenging vision problems like classification, detection, segmentation, *etc*. However, the previous paradigm, which involved training supervised models on many sub-tasks and unifying them into a large system, was not streamlined and offered limited steerability for real-world applications. Addressing this issue, this thesis, this thesis introduces advancements in the realm of visual perception. It focuses on enhancing the steerability of visual intelligence systems. This is achieved through a series of innovative approaches that offer a promising path for building customizable, large multimodal models that follow human intent at an affordable cost.

The introductory chapters lay the foundation by highlighting the importance of visual perception models in various applications and outlining the limitations of early models in terms of steerability. The thesis then presents main contributions: First, it introduces a method for enhancing the customizability of vision-language models using retrieval. Second, it explores to bring the steerability to visual intelligence systems with natural language instructions. The second part is distributed into two chapters: (1) instruction-following large multimodal models, (2) a systematic study for unsolved problems in multimodal models and a cost-effective scaling of multimodal models that marks as the first model from the academia to surpass industry SoTA models like Google Gemini.

1 INTRODUCTION

Vision models, especially those empowered by deep learning [51, 50], have profoundly transformed our ability to interpret and interact with the visual world. These models have achieved remarkable success across a range of applications, mastering tasks such as image classification, object detection, and semantic segmentation, *etc*. Their capacity to analyze and understand visual data has surpassed traditional methods, leading to innovative applications in fields ranging from autonomous driving to medical imaging.

Despite these advancements, the application of vision technologies in real-world scenarios often faces significant challenges. One major limitation is the inflexibility of these systems when faced with tasks that require a broader understanding of context or user-specific needs. For instance, while a model might excel in identifying objects within an image, it may struggle to understand the narrative or emotional context that a human might infer. This is largely because traditional vision systems are designed as highly specialized tools, each fine-tuned for specific tasks without a unified framework to handle diverse or unexpected scenarios that mimic human visual understanding.

The integration of language with vision, *i.e*. vision-language models, proposes a solution to these limitations. These models aim not just to see but to interpret and generate visual content that aligns with complex human instructions, providing a more intuitive interface for users. However, the current generation of vision-language models, while capable, still often requires extensive customization and specialist knowledge to adapt to new tasks or domains. This gap highlights a crucial need for models that are not only powerful but also versatile and user-friendly, enabling non-specialist users to leverage advanced vision capabilities in their everyday applications without the need for deep technical expertise.

The potential benefits of steerable vision-language systems are substantial, particularly when highlighting their responsiveness to natural language instructions, their customizability, and their reduced dependence on extensive human-labeled data. These attributes represent a crucial advancement in our approach to developing and deploying machine learning models within visual domains, enabling a more intuitive and flexible interaction between humans and AI systems:

Efficient Customization with Minimal Labeling Effort: Traditional computer vision models often require vast amounts of labeled data to achieve high performance, especially when customized to specialized domains. This process is not only labor-intensive but also costly. With the advent of large foundational models in vision-language tasks, the need for extensive manual labeling is drastically reduced. These large models, pretrained on diverse datasets, possess a broad understanding of the world, enabling them to adapt to new domains with minimal additional input. For example, a retail company could fine-tune such a model to recognize and categorize its unique product inventory using only a small set of example images and descriptions. This capability significantly lowers the barrier to deploying advanced vision systems across various industries, from healthcare, where it could rapidly adapt to recognize rare medical imaging findings, to wildlife conservation efforts where quick identification of species from camera trap images is essential.

Unified Model Steerability through Natural Language Instructions: One of the most transformative aspects of steerable vision-language models is their ability to operate under a unified framework that can be directed using natural language instructions. This shift away from the traditional model, where specialized knowledge and extensive coding were prerequisites, democratizes access to advanced AI technologies. For instance, educational technologists can use such models to create interactive learning aids where students explore historical or scientific concepts through dynamically generated visual content that responds to their queries. Similarly, in the film and gaming industries, directors and game designers can articulate complex scenes or environments that the model then visually interprets, significantly speeding up the creative process and reducing the dependency on large teams of animators or programmers. Moreover, these advancements offer significant benefits for accessibility, particularly for visually impaired individuals. For example, a vision-language system can be steered via voice commands to describe physical environments or to identify and articulate the contents of images and videos in real-time. This capability could transform everyday interactions for visually impaired users, providing them with a more independent and enriched experience as they navigate both physical and digital spaces. This steerability not only makes the technology more accessible but also enhances its flexibility, allowing for rapid prototyping and iterative design in creative and technical fields alike, ultimately making powerful AI tools accessible and useful across a broader range of applications and industries.

My PhD thesis focuses on the steerability and customizability of vision-language models. Specifically, it contributes to the field by refining capabilities such as efficient customization of large models with minimal human effort, and pioneering a steerable, unified visual intelligence system through natural language instructions. By focusing on these critical areas, my research has established new standards for intuitive and versatile interactions across both general and specialized uses of visual intelligence systems and multimodal research.

Learning Customized Visual Models with Retrieval-Augmented Knowledge: CLIP [112], through large-scale contrastive pretraining, enables zero-shot adaptation for image classification. To better adapt CLIP to the downstream, common approaches often involves finetuning on the downstream data, which again requires human labeling. Chapter 2 of my thesis introduces REACT [87], a framework that enhances and specializes the capabilities of vision-language models like CLIP with retrieval. By strategically retrieves the most pertinent image-text pairs from a web-scale database for the downstream, and the proposed locked-text gated-image tuning for adaptation, REACT effectively and efficiently enhances its adaptability to specialized tasks.

Steerable Vision Models via Natural Language Instructions: Despite the great adaptation performance REACT brings about, creating a streamlined and steerable visual system still requires much effort and remains under-explored. Chapter 3 introduces **LLaVA** [86], the pioneering effort to build instruction-following large multimodal models. It introduces a text-only data engine that facilitates the automatic synthesis of instruction-following multimodal data, and presents the first open-source end-to-end trained large multimodal model that efficiently combines a vision encoder with an LLM for a steerable and robust visual and language understanding model. **LLaVA** has also inspired a series of follow-up works of various modalities and downstream domains.

Progressing further, Chapter 4 details the evolution of this framework through **LLaVA-1.5**, which systematically investigates and refines the design choices within the LLaVA architecture. LLaVA-1.5 advances the state-of-the-art across 11 benchmarks through strategic modifications with great efficiency. This chapter also addresses open problems in large multimodal models (LMMs), paving the way for broader adoption and deeper exploration of large multimodal models in the research community.

2 LEARNING CUSTOMIZED VISUAL MODELS WITH RETRIEVAL-AUGMENTED KNOWLEDGE

Image-text contrastive learning models such as CLIP have demonstrated strong task transfer ability. The high generality and usability of these visual models is achieved via a web-scale data collection process to ensure broad concept coverage, followed by expensive pre-training to feed all the knowledge into model weights. Alternatively, we propose **React**, **RE**trieval-**A**ugmented **C**us**T**omization, a framework to acquire the relevant web knowledge to build customized visual models for target domains. We retrieve the most relevant image-text pairs (~3% of CLIP pre-training data) from the web-scale database as external knowledge and propose to customize the model by only training new modularized blocks while freezing all the original weights. The effectiveness of React is demonstrated via extensive experiments on classification, retrieval, detection and segmentation tasks, including zero, few, and full-shot settings. Particularly, on the zero-shot classification task, compared with CLIP, it achieves up to 5.4% improvement on ImageNet and 3.7% on the Elevater benchmark (20 datasets).

2.1 Introduction

It has been a fundamental research problem in computer vision (CV) to build a transferable visual system that can easily adapt to a wide range of downstream tasks. With remarkable advances in deep learning, a de facto solution to achieve this is to train deep neural networks on a large amount of data to pursue the so-called *generic* visual representations. This dates back to the standard supervised training on ImageNet [30], whose superb representation power is further demonstrated in BiT [64]/ViT [32] by scaling up the training to JFT300M [127]. Along the way, recent efforts have been applied to the popular image self-supervised learning [49, 19, 48] to reduce the demand for labeled data. The third approach is image-text contrastive learning trained on billion-scale web-crawled image-text pairs. Such models, like CLIP [112] and ALIGN [58], are able to achieve great performance on different downstream domains, without the need of any human labels.

Excellent empirical performance has been achieved with the above three pre-training methods, by following the well established two-stage *pre-training then adaptation* pipeline: model pre-training from scratch on large data, then model adaptation directly on downstream tasks. Specifically, the pre-trained models are adapted to downstream tasks by considering the available task-specific samples only: either evaluated in a zero-shot task transfer manner, or updated using linear probing (LP) [112], finetuning (FT) [75], or prompt tuning [173, 117]. Following this two-stage pipeline, most research has reverted to the faith that building transferable visual systems is equivalent to developing more generic visual models by feeding all knowledge in the model pre-training stage. Therefore, the community has been witnessing a trend in exploring scaling success of pre-training model and data size with less care on the target domain, hoping that the model can adapt to any downstream scenario.

In this chapter, we argue that the conventional two-stage pipeline above is over-simplified and less efficient, in achieving the goal of building a transferable visual system in real-world settings. Instead, we propose a *customization* stage in between the pre-training and adaptation, where customization is implemented by systematically leveraging retrieved external knowledge. The inspiration comes from how humans are specialized in society for better generalization: instead of trying to memorize all concepts, humans are trained/prepared in a relevant subject to master a certain skill, while maintaining the basic skills in pre-training.

To this end, we explore a systematic approach to acquire and learn

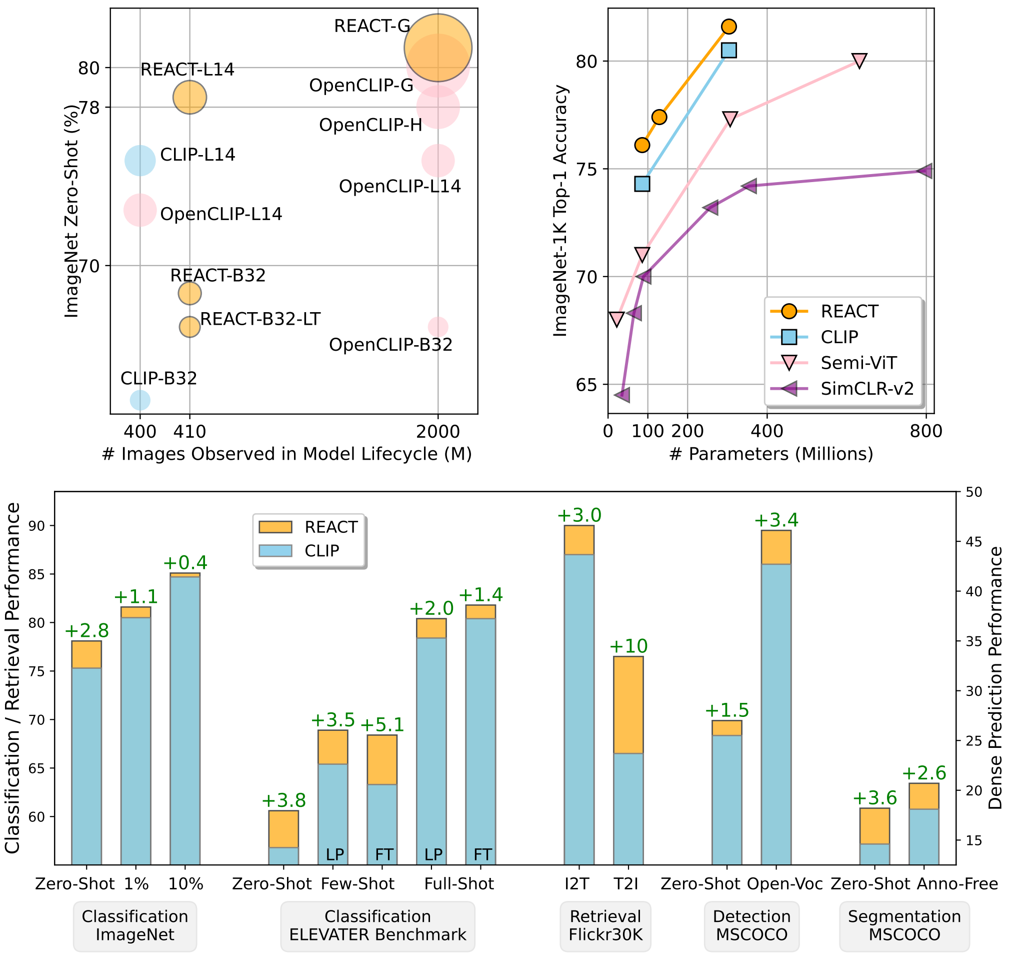


Figure 2.1: REACT achieves the best zero-shot ImageNet performance among public checkpoints (Top Left), achieves new SoTA on semi-supervised ImageNet classification in the 1% labeled data setting (Top Right), and consistently transfer better than CLIP on across a variety of tasks, including ImageNet classification, zero/few/full-shot classification on 20 datasets in ELEVATER benchmark, image-text retrieval, object detection and segmentation (Bottom). Please see the detailed numbers and settings in the experimental section. For the left figure, circle size indicates model size.

with external knowledge sources from a large image-text corpus for model customization. The process of collecting external image-text knowledge is fully automatic without extra human annotation. The acquired knowledge typically contains richer information about the concept: relevant images that never appear in the downstream training and evaluation set, and richer text descriptions about concept semantics. Such multi-modal knowledge sources are generally available on the web, and further open-sourced like LAION [119, 118]. They cover a variety of domains, making it possible to develop customized visual models for task-level transfer. Similar retrieval-augmented intuitions have been exploited in computer vision for class-level transfer [91], but not yet for task-level transfer (similar to that of CLIP). Our main findings/contributions can be summarized as follows.

We propose to explore the potential of the web-scale image-text corpus as external knowledge to significantly improve task-level transfer performance on the target domain at an affordable cost. A simple and effective strategy is proposed. To begin with, we build a large-scale multi-modal indexing system to retrieve the relevant image-text pairs using CLIP features and approximate nearest neighbor search. For a CV problem, the task instruction is often sufficiently specified with text such as class names, which allows us to utilize them as queries to retrieve the relevant image-text pair knowledge from the indexing system. *No images from the CV problem are needed.* To efficiently build the customized visual model, we propose a novel modularized learning strategy: only updating the additional trainable weights on the retrieved knowledge, and freezing the original model weights. Hence, the model masters the new skill without forgetting basic skills.

The generality and effectiveness of the proposed customization strategy is demonstrated on four CV problems. We instantiate it with CLIP, and develop the customized visual models for image classification on ImageNet and 20 datasets in ELEVATER [75], image-text retrieval on COCO [83]/Flickr [110], as well as object detection and semantic segmentation on COCO [83]. The knowledge bases are considered as LAION [119] and larger web-crawled multi-modal data. The retrieval-augmented knowledge (~3% image-text pairs compared with the original training data) significantly improves the model's zero-shot performance without the need of accessing any images on downstream tasks. See Figure 2.1 for highlighted results. For example, our ViT-L/14 checkpoint achieves 78.5% zero-shot accuracy on ImageNet [30], surpassing all public checkpoints from CLIP [112] and OpenCLIP [55], including those with larger model size and trained on a much larger LAION-2B [118]. The new customized models demonstrate higher few/full-shot performance than the generic model counterparts.

Our retrieval system, codebase, and pre-trained models are publicly available. To make this line of research more accessible, our retrieved subsets for both ELEVATER and ImageNet will also be made available, with an easy-to-use toolkit to download the subsets without storing the whole dataset locally. It poses a feasible direction for leveraging the ever-increasing data from the Internet for customized visual recognition, especially for the low-resource regimes.

2.2 Related Work

Vision-Language Pretraining. Learning transferable visual representations from natural language supervision is an emerging research area. The pioneering works of CLIP [112] and ALIGN [58] make use of contrastive learning to pretrain models on billion-scale web-crawled image-text pairs. There are an increasing number of studies to improve their generality from various modeling perspectives, including training objectives [40, 31, 155, 102, 160, 39], scaling techniques [23, 155, 109], data efficiency [80, 68], and leveraging multilingual correlations [57, 23]. In academia, several works demonstrate techniques to improve the learned semantic representations on datasets at a smaller scale (*e.g*. CC3M [122], CC12M [15], YFCC15M [112, 131]), by exploring pretraining on a unified image-text-label space [144], token-level contrastive loss [149], and auxiliary within-modality contrastive loss [101, 82, 145, 153]. Complementary to the above works, we build on top of existing pre-trained generic models, and aim to improve the model's performance by customizing them using retrieved *relevant* image-text pairs.

Retrieval-Augmented Models. In natural language processing, several works augment large language models with external data encoded with structured language and relation representations [108, 46, 69, 89, 158, 11, 62]. Motivated by retrieval-augmented models in NLP, several recent works leverage visual and / or textual knowledge to improve classification [91], question answering [142, 97, 146, 20], image generation [10, 124, 21, 174], and multi-modal tasks simultaneously [150]. RAC [91] improves long-tail classification by retrieving from a non-parametric memory consisting of pre-encoded images and text. K-LITE [123] enhances the text prompts with the retrieved external knowledge that is encoded in natural language. This chapter leverages the paired knowledge of image-text and aims to improve task transfer performance for core vision problems such as classification, retrieval, detection and segmentation.

Adaptation of Vision-Language models. CLIP demonstrates impressive zero-shot and linear probing performance on different downstream domains. Several works explore improving the domain adaptation performance on CLIP models. ELEVATER [75] leverages the text encoder outputs to initialize the task-specific linear head to improve the linear probe and finetuning performance of CLIP. Inspired by prompting techniques in NLP, recent works [173, 117] make use of learnable prompts that are trained on a few samples on downstream tasks. Similar to these works, this chapter aims to improve CLIP's performance on downstream tasks, while making use of relevant image-text pairs data to improve the model's performance, without access to the downstream images. Furthermore, when downstream samples are available, they are complimentary to our method.

2.3 Retrieval-Augmented Customization

Preliminaries

Computer vision models have achieved strong transfer performance, when learning with large-scale image data only [49], image-label data [64] and/or image-caption data [112, 144, 159]. Without loss of generality, we follow [144] and define a unified triplet-wise format $(\boldsymbol{x}, \boldsymbol{t}, y)$ for image-text-label data, where $\boldsymbol{x} \in \mathcal{X}$ is an image, $\boldsymbol{t} \in \mathcal{T}$ is its language description, and $y \in \mathcal{Y}$ is a label indicating the index of the unique language description in the dataset. In a general form, the language description is a text sequence $\boldsymbol{t} = [t_1, \cdots, t_L]$. It ranges from simple category names representing visual concepts when L is small, to more free-form and semantic-rich sentences such as captions when L is relatively large.

A typical transfer learning pipeline follows the procedure of *pre-training then adaptation*: (i) With large-scale pre-training, an image encoder foundation model f_θ parameterized by θ is first trained to represent image $\boldsymbol{x}$ as a visual feature vector $\tilde{\boldsymbol{v}} \in \mathbb{R}^{P \times 1}$: $\tilde{\boldsymbol{v}} = f_\theta(\boldsymbol{x})$. For recent language-image models [112], a dual-encoder architecture is often employed, where an additional text encoder $f_{\boldsymbol{\phi}}(\boldsymbol{t})$ parameterized by $\boldsymbol{\phi}$ represents the sentence $\tilde{\boldsymbol{u}} \in \mathbb{R}^{P \times 1} : \tilde{\boldsymbol{u}} = f_{\boldsymbol{\phi}}(\boldsymbol{t})$. (ii) Given a downstream task, model adaptation is typically performed using the available task-specific information, or *task instruction* $\mathcal{I}$. For example, the task-level transfer of a language-image model is described as:

- *Zero-shot.* In a customized setting, the simplest task definition can be provided as a set of category names for visual recognition, leading to the task instruction $\mathcal{I}_0 = \{\boldsymbol{t}\}$. No training image $\boldsymbol{x}$ is available, not to mention the corresponding label y.

- *Few/Full-shot.* The users may spend annotation cost to curate N image-label pairs as the training instances, making the task instruc-

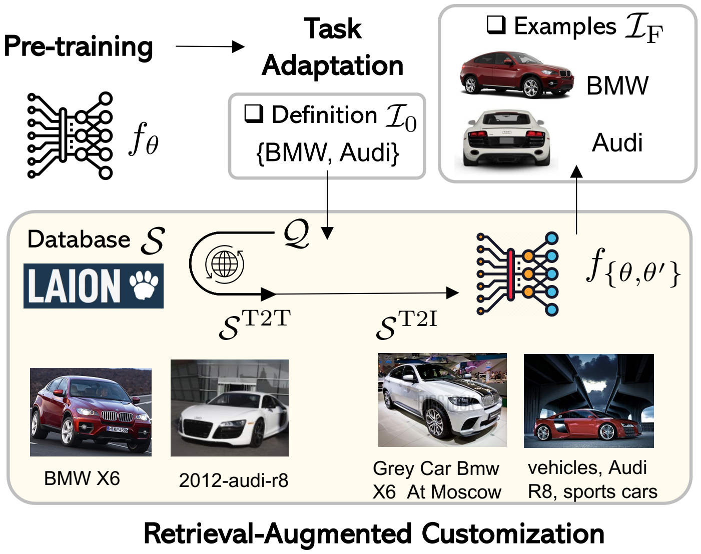

Pre-training
f_θ
Task Adaptation
❑ Definition $\mathcal{I}_0$
{BMW, Audi}
❑ Examples $\mathcal{I}_F$
BMW
Audi
Database $\mathcal{S}$
LAION
$\mathcal{Q}$
$\mathcal{S}^{\mathrm{T2T}}$
$\mathcal{S}^{\mathrm{T2I}}$
$f_{\{\theta,\theta'\}}$
BMW X6
2012-audi-r8
Grey Car Bmw X6 At Moscow
vehicles, Audi R8, sports cars
Retrieval-Augmented Customization

Figure 2.2: Illustration of the proposed REACT framework.

tion more specific, $\mathcal{I}_F = \{(\mathbf{x}_n, \mathbf{t}_n, y_n)\}_{n=1}^{N}$, which allows updating the image encoder model f_θ for better adaptation performance.

In this chapter, we assume there exists a web-scale image-text corpus as the external knowledge source $\mathcal{S} = \{(\mathbf{x}_m, \mathbf{t}_m)\}_{m=1}^{M}$, where M is the database size, *e.g*. 400M for LAION [119]. One may use the task instruction $\mathcal{I}$ as a query to seek additional relevant knowledge to build a more transferable visual system. Given the downstream task instruction $\mathcal{I}$ and an external knowledge source $\mathcal{S}$, our goal is to learn customized visual-semantic representations, which are readily transferable to the downstream task of interest, whose training and evaluation images are not observed during the customization process. To this end, we propose REACT. We illustrate the high-level idea in Figure 2.2, and describe the process as follows.

Multi-modal External Knowledge

Knowledge Base Construction. We explore web-scale image-text data as the multi-modal knowledge base $\mathcal{S}$ in this chapter. Ideally, one may consider the entire web as the knowledge base, and use Google or Bing search to retrieve the relevant knowledge. We consider two large static datasets with image-text pairs. To control the experiment complexity and ensure reproducibility, we use LAION-400M [119], a publicly available database with 400M pairs, for most of the experiments. To further study the scaling influence of the retrieval base, we conduct comparisons on Web-800M, a privately collected web database with 800M pairs.

To facilitate an efficient knowledge acquisition process, we use pre-trained contrastive models (*e.g*. CLIP) as the feature extractor, and build a cross-modal retrieval system using FAISS [60]. We use its Hierarchical Navigable Small World (HNSW) approximate k-NN lookup [95] to balance performance and efficiency. After the retrieval system is built on the designated retrieval pool, it can be efficiently used for retrieving relevant image-text pairs for *various* downstream domains.

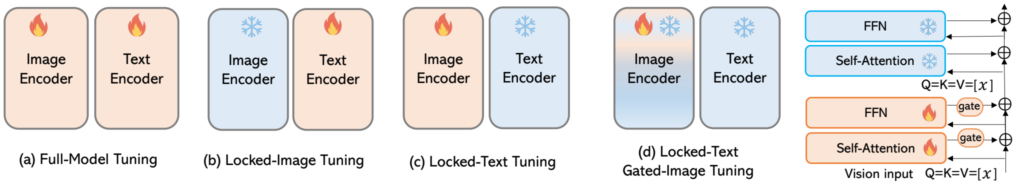

(a) Full-Model Tuning
(b) Locked-Image Tuning
(c) Locked-Text Tuning
(d) Locked-Text Gated-Image Tuning

Figure 2.3: Illustrative comparisons across different model tuning methods. (a) and (b) are existing baseline tuning methods. For model customization in a target domain, we found that (c) and (d) work better. One layer of the proposed modularized image encoder in locked-text gated-image tuning is illustrated in right side.

Retrieval-Augmented Task Instruction. To facilitate the same interface for various customized visual tasks in the wild, it is desirable to have the same uniform task instruction schema. In NLP, all task instructions can follow the same uniform schema, composed of *task definition* and *positive/negative examples* [100, 139]. Here, the task definition defines a given task in natural language, completely specifying how an input is expected to be mapped to an output text. We note a coherence connection between this NLP task schema and the customized zero/few/full-shot CV settings in Section 2.3. Following a similar schema, the minimum requirement to specify a visual task is the task definition $\mathcal{T}_0$, where category names illustrate the target visual concepts in natural language. Though adding human-annotated examples is a natural way to clarify the task and yield the complete schema $\mathcal{T}_F$, extra cost is introduced.

It is of high interest to clarify the task using relevant examples, without human curating cost. Therefore, we propose to augment the task instruction with the retrieved examples from the external multi-modal knowledge base $\mathcal{S}$. For each concept $\mathbf{t} \in \mathcal{T}_0$ in a given task, we first represent it in natural language $\mathbf{q} = g_{\mathtt{prompt}}(\mathbf{t})$ using the language prompt as in [112], through inserting the concept into a set of task-specific templates $\mathcal{P}$. The task definition is expanded in its natural language form:

$$\mathcal{Q} = \{\mathbf{q} \mid \mathbf{q} = g_{\mathtt{prompt}}(\mathbf{t}), \forall \mathbf{t} \in \mathcal{T}_0, \mathtt{prompt} \in \mathcal{P}\}. \tag{2.1}$$

Next, we perform our knowledge retrieval process to acquire the relevant image-text pair $\mathbf{s} = g_{\mathtt{retrieve}}(\mathbf{q})$ from the source $\mathcal{S}$. Two types of retrieval processes are considered to acquire the top-K pairs:

- Text-to-Text (T2T) retrieval allows us to retrieve more relevant examples as they have a better match with our target concept. The T2T-retrieved set for $\mathcal{T}_0$ is:

$$\mathcal{S}^{\mathrm{T2T}} = \{(\mathbf{x}, \mathbf{t}) \in \mathcal{S} : \operatorname*{argmax}_{\mathbf{t} \in \mathbb{T}, |\mathbb{T}| = K} f_{\boldsymbol{\phi}}(\mathbf{t})^{\top} f_{\boldsymbol{\phi}}(\mathbf{q}), \forall \mathbf{q} \in \mathcal{Q}\} \tag{2.2}$$

- Text-to-image (T2I) retrieval allows us to have more diversity in the text descriptions in our retrieved examples. The T2I-retrieved set for $\mathcal{I}_0$ is:

$$\mathcal{S}^{\mathrm{T2I}} = \{(\mathbf{x}, \mathbf{t}) \in \mathcal{S} : \underset{\mathbf{x} \in \mathbb{X}, |\mathbb{X}| = K}{\operatorname{argmax}} \, f_{\boldsymbol{\theta}}(\mathbf{x})^{\top} f_{\boldsymbol{\phi}}(\mathbf{q}), \forall \mathbf{q} \in \mathcal{Q}\} \tag{2.3}$$

Both $\mathcal{S}^{\mathrm{T2T}}$ and $\mathcal{S}^{\mathrm{T2I}}$ are retrieved examples to augment the task definition $\mathcal{I}_0$, without accessing the images in the training or validation set of the task. Compared to $\mathcal{I}_F$, they are "free" external knowledge to clarify the task and can be used to build a more transferable system.

Model Customization

After retrieving the relevant multi-modal examples, one may employ the naive customization solution by fine-tuning the full-model initialized from pre-trained weights, as in Figure 2.3(a). Alternatively, we propose an affordable solution to endow pre-trained models with a new capability to leverage this external knowledge. The pre-trained generic visual models have gained strong transfer abilities and access to a large amount of internal knowledge stored in the model weights. We freeze the weights of these models so that their initial capacity remains unchanged. To bridge these pretrained models harmoniously to the customized domain, we consider *locked-text gated-image tuning* with the following two techniques, illustrated in Figure 2.3(d).

Modularized Image Encoder. In order to provide sufficient expressivity to the model and make it able to adapt well on retrieved knowledge, we insert gated self-attention dense blocks in between the original layers of the image encoder, and train the new blocks from scratch. Those blocks are made of a self-attention layer, that attends the early layer inputs, followed by an extra dense feed-forward layer. Please see a visual illustration of this gated block in the rightmost of Figure 2.3(d). We denote the parameters of all new modules as θ'. This design is inspired by the gated cross-attention-dense blocks in Flamingo [3] and frozen multi-modal model [133]. The difference is that the trainable module is introduced in Flamingo to enable cross-modal conditioning, while we adapt it for model growing in new customized domains.

Frozen Text Encoder. The text encoder in language-image contrast models represents the task semantic space. To maintain it, we propose *locked-text tuning*, which freezes the text model weights so that the generic task encoding knowledge remains locked; see Figure 2.3(c). This is in contrast with *locked-image tuning* (LiT) [160] in Figure 2.3(b), where the image encoder is frozen and the text encoder is fine-tuned, which teaches a text model to read out good representations from a pre-trained image model for new tasks.

We extract the normalized feature vectors in a hyper-sphere using $\boldsymbol{u}_i = \frac{f_{\{\theta,\theta'\}}(\boldsymbol{x}_i)}{\|f_{\{\theta,\theta'\}}(\boldsymbol{x}_i)\|}$ and $\boldsymbol{v}_j = \frac{f_{\phi}(\boldsymbol{t}_j)}{\|f_{\phi}(\boldsymbol{t}_j)\|}$. To customize the model wrt task definition $\mathcal{T}_0$, we update θ' using a bidirectional learning objective between images and language on the retrieved knowledge pool $\mathcal{S}^{\mathrm{T2T}}$and/or $\mathcal{S}^{\mathrm{T2I}}$:

$$\min_{\{\theta'\}} \; \mathcal{L}_{C} = \mathcal{L}_{i2t} + \mathcal{L}_{t2i}, \text{ with } \mathcal{B} \sim \mathcal{S}^{\mathrm{T2T}} \text{or } \mathcal{S}^{\mathrm{T2I}} \tag{2.4}$$

$$\mathcal{L}_{i2t} = -\sum_{i \in \mathcal{B}} \frac{1}{|\mathcal{P}(i)|} \sum_{k \in \mathcal{P}(i)} \log \frac{\exp(\tau \boldsymbol{u}_i^{\top} \boldsymbol{v}_k)}{\sum_{j \in \mathcal{B}} \exp(\tau \boldsymbol{u}_i^{\top} \boldsymbol{v}_j)} \quad \text{and} \tag{2.5}$$

$$\mathcal{L}_{t2i} = -\sum_{j \in \mathcal{B}} \frac{1}{|\mathcal{Q}(j)|} \sum_{k \in \mathcal{Q}(j)} \log \frac{\exp(\tau \boldsymbol{u}_k^{\top} \boldsymbol{v}_j)}{\sum_{i \in \mathcal{B}} \exp(\tau \boldsymbol{u}_i^{\top} \boldsymbol{v}_j)} \tag{2.6}$$

where τ is a temperature hyper-parameter controlling the strength of penalties on hard negative samples, and $\mathcal{P}(i) = \{k | k \in \mathcal{B}, \boldsymbol{v}_k^{\top} \boldsymbol{v}_i \geqslant \gamma\}$, $\mathcal{Q}(j) = \{k | k \in \mathcal{B}, \boldsymbol{v}_k^{\top} \boldsymbol{v}_j \geqslant \gamma\}$. We set $\gamma = 0.9$ for classification tasks to force image-text pairs sharing the similar text to be positive. Note (2.4) is a general form; it reduces to UniCL [144] when $\gamma = 1.0$; it further reduces to the training objective of CLIP [112] when there is a one-to-one mapping between an image and its paired caption in a batch, *i.e.* $\mathcal{P}(i) = \{i\}$ and $\mathcal{Q}(j) = \{j\}$.

In our empirical study we find that locked pre-trained image and text encoders with trainable gated modules in image encoder work best. Once the customized visual models are trained with the retrieved knowledge, we transfer it to the downstream domain for zero/few/full-shot evaluation.

2.4 Experiments

In this section, we conduct experiments to answer three research questions: (1) What are the unique advantages of retrieval-augmented image-text knowledge for task transfer? (2) How does our design choice of locked-text gated-image tuning compare to existing methods for model customization? (3) Is customization still beneficial in settings where the training data in downstream tasks are observed, *i.e.*, in few-shot or full-shot settings? (4) Does customization scales well to dense prediction tasks like detection/segmentation?

We evaluate our models on four CV problems: image classification, image-text retrieval, object detection, and semantic segmentation. We first consider ImageNet [30] for zero-shot task transfer. We then further evaluate our model on ELEVATER [75], which is an open-set image classification benchmark that contains 20 datasets. We also conduct experiments on image-text retrieval with MSCOCO [83] and Flickr [154] dataset. Finally, we evaluate on object detection and semantic segmentation with MSCOCO [83] dataset.

One of the most intriguing benefits of REACT is that it does not need access to any images from the downstream task. Therefore, we first evaluate on task-level zero-shot transfer, which requires no images in the target to be observed [112, 123, 75]. This setting is different from traditional class-level zero-shot [143], where both the category and images in evaluation

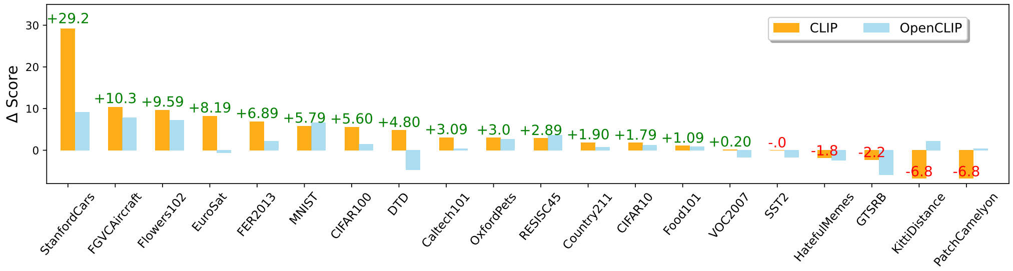


Figure 2.4: Zero-shot comparison on Elevater ICinW 20 datasets. React (B32) improves over the base checkpoints on most datasets.

should not be observed in training. We argue that ImageNet concepts have been observed in CLIP (Sec. 2.2 of [112]) and other web-scale trained models [80], as WordNet synsets and common words in English Wikipedia are explicitly added in the query list when searching for (image, text) pairs in their training data construction process.

Image Classification

Zero-Shot on ImageNet-1K

As shown in Table 2.1, by customizing the generic model CLIP/OpenCLIP on 10M retrieved image-text pairs from LAION-400M, React achieves a significant and consistent gain (up to 5.4%) on zero-shot image classification on ImageNet-1K, with different backbones and original pretraining datasets. There are three interesting findings.

F1: React can benefit from model's own pre-training data. Compared to OpenCLIP [55] (ViT-B/32) trained on LAION-400M, by training on 10M relevant pairs from the *same* LAION-400M dataset, React improves over OpenCLIP by 3.5%. Note that the model purely uses the image-text pairs that it has seen during its pre-training, and does *not* see any extra data. This shows that React can more adequately adapt to the target domain during the model customization stage, suggesting a favorable property that no new data is required for customization.

F2: REACT efficiently explores new image-text sources, even for large models. We customize CLIP [112] ViT-L/14 on 10M retrieved relevant image-text pairs, and the model achieves a 2.8% improvement to 78.1%. This surpasses the checkpoint with a much larger ViT-H/14 backbone and trained on a much larger LAION-2B dataset. This suggests that REACT is a more sample-efficient approach to improve the model performance on the domain-of-interest.

F3: Scaling up the retrieval pool increases performance. We perform REACT in a privately collected dataset with over 800M pairs, and train a customized model on 6M retrieved pairs. The performance is increased to 78.5%, yielding 0.9% gain compared with 6M pairs retrieved from LAION-400M. This suggests that REACT scales well with the larger retrieval pool. It showcases REACT as a cost-efficient approach to leveraging the ever-increasing web image-text corpus.

Zero-, Few-, and Full-Shot on ELEVATER

As a proxy for performing vision tasks for many customized scenarios in the wild, we consider the *image classification in the wild* (ICinW) benchmark in ELEVATER [75]. It consists of 20 datasets from a diverse selection of domains and covers a wide range of concepts, totaling 1151 classes.

We perform multi-modal knowledge retrieval for 20 datasets together – the retrieved samples are around 10M image-text pairs in total, on which one single customized visual model is trained. After the process, we feed the customized model to different downstream tasks separately. For each downstream dataset, we use the official ELEVATER toolkit to obtain the train/val/test splits, and perform zero-shot, few-shot, and full-shot evaluation.

We report the average scores in Table 2.2. It achieves 3.8% improvement in the zero-shot setting, even when we do not perform a separate

f_θ	Pretrain Data	Retrieved Data		Method	ImageNet-1K Zero-Shot
		Dataset	Size		
B/32	WIT-400M	–	–	CLIP	63.2
		L-400M	10M	REACT	**68.6** (+5.4)
	LAION-400M	–	–	OpenCLIP	62.9
		L-400M	10M	REACT	**66.4** (+3.5)
L/14	WIT-400M	–	–	CLIP	75.3
		L-400M	6M	REACT	**77.6** (+2.3)
		L-400M	10M	REACT	**78.1** (+2.8)
		W-800M†	6M	REACT	**78.5** (+3.2)
	LAION-400M	–	–	OpenCLIP	72.8
	LAION-2B	–	–	OpenCLIP	75.3
H/14	LAION-2B	–	–	OpenCLIP	78.0
G/14	LAION-2B	–	–	OpenCLIP	80.1
		L-2B	12M	REACT	**81.0** (+0.9)

Table 2.1: Comparison of zero-shot task transfer with public checkpoints from CLIP [112] and OpenCLIP [55]. LAION [119, 118] is abbreviated as "L" in the table. Web-800M†: a privately collected web database with 800M image-text pairs. By continue pretraining on only ~10M retrieved data, REACT outperforms *all* public CLIP/OpenCLIP checkpoints.

customization for different datasets. This demonstrates the robustness of our customization process. Further, we see the consistent improvement in few-shot and full-shot settings, including both linear probe (LP) and fine-tuning (FT). This result is encouraging, as it demonstrates that when we have access to some or all data from the downstream task, the proposed model customization stage remains beneficial. Therefore, *we advocate model customization process in both data-limited and data-rich settings*.

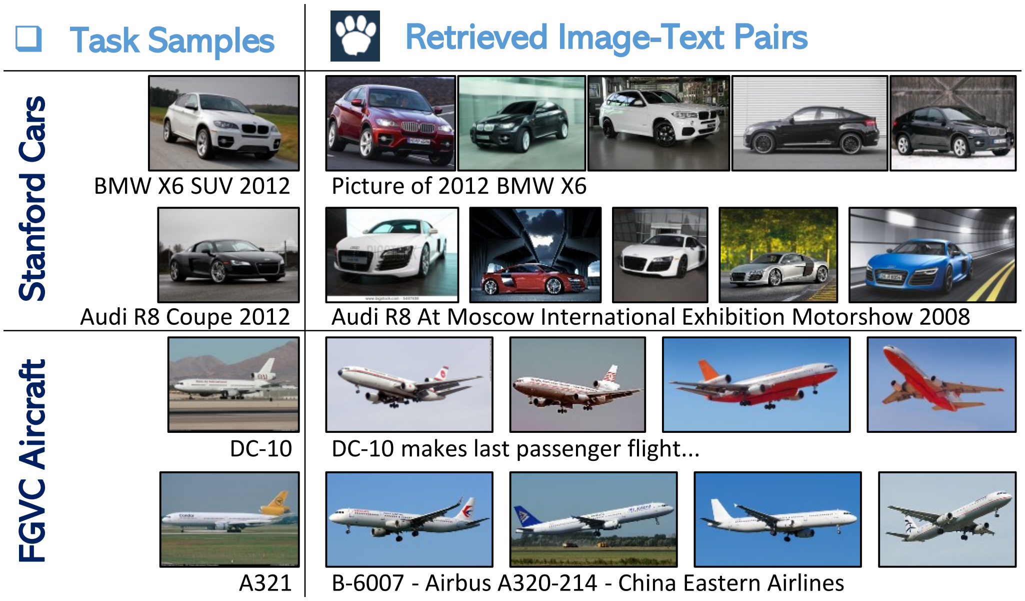
Breakdown Analysis. Next, we ask why does the retrieved image-text knowledge improve the zero-shot task transfer performance on a broad range of datasets? We compare the breakdown performance on all 20

Method	Zero-Shot	Few-shot LP	Few-shot FT	Full-shot LP	Full-shot FT
CLIP	56.8	65.4	63.3	78.4	80.4
REACT	**60.6**	**68.9**	**68.4**	**80.4**	**81.8**
Gains	(+3.8)	(+3.5)	(+5.1)	(+2.0)	(+1.4)

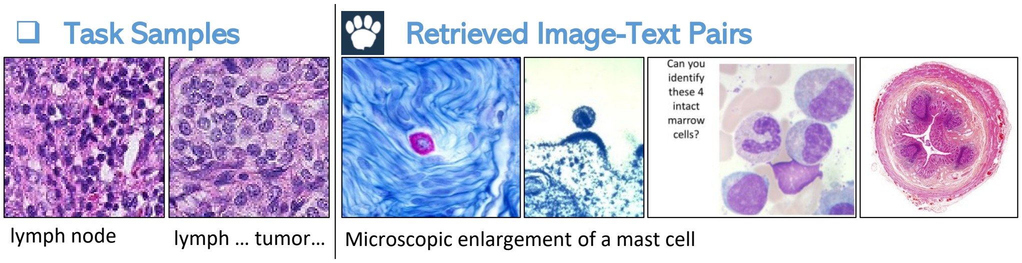
Table 2.2: The average scores of image classification performance on 20 datasets in ELEVATER. REACT consistently outperforms CLIP in both data-limited and data-rich regimes.

datasets in Figure 2.4 for the zero-shot settings. Out of 20 datasets, the retrieval-augmented knowledge shows superior/comparable/inferior performance to the baseline on 15/1/4 datasets for CLIP and 14/0/6 datasets for OpenCLIP, respectively. Most of the improved and failure datasets are consistent for both checkpoints. For the top two datasets that gains the most, *i.e*. StanfordCars and FGVC Aircraft, relevant image-text knowledge is retrieved from the web-crawled data LAION-400M to describe the concepts; see Fig. 2.5a. Interestingly, this observation is complementary to K-LITE [123], which failed on these two datasets, because no knowledge was extracted from Wiktionary for them, as it often requires domain-specific knowledge and even visual knowledge to best define a car brand (*e.g*. BMW X6 SUV or Audi R8) or an aircraft model type (*e.g*. DC-10 or A321).

Limitations. As shown in Fig. 2.4, REACT struggles on the PatchCamelyon dataset, a cancer cell recognition benchmark. We visualize the retrieved samples and the samples from the original training set in Fig. 2.5b. The retrieved images are either instruction photos and from another sensing method, which exhibits a different visual distribution from PatchCamelyon. This suggests the importance of ensuring the retrieval quality for the domain-of-interest.


Task Samples
Retrieved Image-Text Pairs
Stanford Cars
BMW X6 SUV 2012
Picture of 2012 BMW X6
Audi R8 Coupe 2012
Audi R8 At Moscow International Exhibition Motorshow 2008
FGVC Aircraft
DC-10
DC-10 makes last passenger flight...
A321
B-6007 - Airbus A320-214 - China Eastern Airlines

(a) Success examples. The two datasets with largest improvement in Fig. 2.4: Stanford-Cars [65] and FGVC-Aircraft [94]. There is a high concept converage for these datasets in LAION, resulting in a relevant and diverse retrieved set.


Task Samples
Retrieved Image-Text Pairs
lymph node
lymph ... tumor...
Microscopic enlargement of a mast cell
Can you identify these 4 intact marrow cells?

(b) Failure case. The dataset with the largest degradation in Fig. 2.4: PatchCamelyon [135]. LAION-400M has a low concept coverage on this domain, and the retrieved samples are in a different distribution from the target set.

Figure 2.5: Success and failure cases in Elevater benchmark. We show class name and the caption of the first retrieved image-text pairs, others are similar and omitted due to limited space.

Image-Text Retrieval

To demonstrate the generality of React, we consider Flickr30K [154] and MSCOCO [83] image-text retrieval tasks, in both zero-shot and full-shot settings. We use the standard image-text contrastive objective [112]. For image-text retrieval task, following [112, 58], we use the CLIP-L/14 with 336x336 input resolution in both zero-shot, customization, and fine-tuning stage. We use the captions from MSCOCO as queries to retrieve 6M image-text pairs and perform customization. Note that *none* of the caption queries are used in the model training stage.

As shown in Table 2.3, REACT improves the generic CLIP counterparts on both zero-shot and full-shot retrieval for Flickr30K and MSCOCO datasets. The gain on zero-shot task transfer is large. On Flickr30K, it achieves 3.4%/10.0% recall improvement for I2T and T2I retrieval, respectively. Afer fine-tuning on full training data, REACT still improves over the baseline slightly. It provides another piece of evidence for REACT in data-rich settings. Furthermore, we conduct the same customization procedure of REACT on a large checkpoint Bletchley [134] with 864M parameters, and observe consistent gains over both datasets. It demonstrates that REACT scales well with model size on retrieval tasks.

Dense Prediction Tasks

Although REACT is optimized with the image-level contrastive loss during the customization stage, we find it beneficial for dense prediction tasks as well. We showcase its application to dense prediction tasks on object detection and semantic segmentation.

Object Detection. For object detection, we choose the state-of-the-art RegionCLIP [170] as our framework. We conduct experiments in two settings: zero-shot inference and open-vocabulary object detection (OVD) on MSCOCO dataset. We perform the model customization following the same setting as Sec. 2.4. Following RegionCLIP, we conduct experiments on ResNet50 backbone. The results are shown in Table 2.4. REACT consistently improves over CLIP checkpoint under all settings.

	Method	Flickr30K				MSCOCO			
		Img → Text		Text → Img		Img → Text		Text → Img	
		R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
Zero-Shot	ImgBert [111]	70.7	90.2	54.3	79.6	44.0	71.2	32.3	59.0
	ALIGN [58]	88.6	98.7	75.7	**93.8**	58.6	83.0	45.6	69.8
	CLIP [112]	88.0	98.7	68.7	90.6	58.4	81.5	37.8	62.4
	CLIP†	87.0	98.3	66.5	88.0	59.2	80.7	37.8	62.4
	React	**90.4**	**99.1**	**76.5**	**93.7**	**63.3**	**85.1**	**47.5**	**72.0**
	Bletchley†	90.8	98.2	78.0	94.0	66.7	85.6	48.9	72.7
	React	**92.1**	**98.7**	**79.2**	**94.7**	**67.7**	**85.9**	**50.5**	**74.4**
Fine-tuned	GPO [17]	88.7	98.9	76.1	94.5	68.1	90.2	52.7	80.2
	ALIGN [58]	95.3	**99.8**	84.9	97.4	77.0	93.5	59.9	83.3
	CLIP†	96.4	**99.8**	86.5	**97.9**	78.3	93.8	60.9	83.8
	React	**96.6**	**99.9**	**86.8**	**98.0**	**78.7**	**94.0**	**61.1**	**84.1**

Table 2.3: Image-text retrieval results on Flickr30K [110] and MSCOCO [83] datasets. CLIP†, Bletchley†: our evaluation.

	Pretrain Method	Backbone	Region Proposals	MSCOCO AP_{50}		
				Novel	Base	All
Zero-Shot	CLIP	ResNet-50	GT	58.6	58.2	58.3
	React	ResNet-50	GT	**58.9** (+0.3)	**59.4** (+1.2)	**59.3** (+1.0)
	CLIP	ResNet-50	RPN	29.7	24.0	25.5
	React	ResNet-50	RPN	**31.6** (+1.9)	**25.4** (+1.4)	**27.0** (+1.5)
OVD	CLIP	ResNet-50	–	14.2	52.8	42.7
	React	ResNet-50	–	**20.6** (+6.4)	**55.1** (+2.3)	**46.1** (+3.4)

Table 2.4: Zero-shot and open-vocabulary object detection results on MSCOCO [83] dataset using RegionCLIP [170] pipeline.

For zero-shot inference, when ground-truth region proposal is used, React improves over CLIP by +1.0 on overall AP50; when the pretrained RPN is used, React demonstrates +1.5/+1.4/+1.9 AP50 improvements on novel, base, and all classes, respectively.

For OVD, we can see that with the React, the detector yields improved performance on base with +2.3 AP50, and importantly, it significantly improves novel categories with +6.4 AP50. This suggests that the injected knowledge during the model customization stage improves the learned fine-grained visual feature that is beneficial to both seen and unseen categories for object detection, when the downstream coarse-grained data is available. This is favored, because (1) the weakly-supervised data such as the coarse-grained image-text pairs requires much less human annotation cost than fine-grained bounding box annotation, (2) the paired data in REACT is free, as it is retrieved from the web, where COCO image-text pairs are not used in customized training.

Semantic Segmentation. For semantic segmentation, we choose the state-of-the-art MaskCLIP [171] as the framework. It investigates three evaluation settings for segmentation. First, it makes use of the pretrained CLIP checkpoint to discover the alignment between grid visual features and the text prompt features, so as to perform zero-shot semantic segmentation. Second, to further improve the performance, MaskCLIP [171] proposes two techniques for refining its zero-shot predictions: key smoothing and prompt denoising. Lastly, when the training images are available, without the need to access the training labels, it further proposes MaskCLIP+ [171] to perform full-shot finetuning on the target training set using pseudo-labels. Following MaskCLIP [171], we use ViT-B/16 checkpoints, and use their official code base to train and evaluate models. We report results in Table 2.5.

On all of the three settings, REACT demonstrates improvements over the MaskCLIP. Notably, when refinement techniques are used, REACT with locked-text tuning demonstrates a significant 3.6% gain in mIoU. Surprisingly, without seeing the downstream COCO images, the zero-shot evaluation of REACT (18.2) even slightly outperforms MaskCLIP+ (18.1), which is finetuned on the downstream training COCO images with self-training.

Summary. These results are encouraging, as it shows that the customized

Method	mIoU zero-shot	 w/ refine	 w/ finetune
MaskCLIP [171]	12.5	14.6	18.1
REACT (Locked-Text)	**14.4** (+1.9)	**18.2** (+3.6)	**20.7** (+2.6)
REACT (Locked-Text Gated-Image)	**14.5** (+2.0)	**16.3** (+1.7)	**19.2** (+1.1)

Table 2.5: Zero-shot and annotation-free semantic segmentation results on COCO Stuff [83] using MaskCLIP [171] (ViT-B/16).

knowledge from REACT transfers well to dense prediction tasks like detection and segmentation.

Ablation Studies

We ablate REACT on ImageNet with CLIP ViT-B/32 checkpoint, with 10M retrieved image-text pairs from LAION-400M. See more ablations in supplementary.

Tuning strategy. We ablate the design choices in the model customization stage: (1) direct tuning the pre-trained weights *vs*. training gated blocks from scratch; (2) updating visual *vs*. text encoder. We report results in Table 2.6. First, a frozen text encoder consistently outperforms a frozen visual encoder. We conjecture the phenomenon is due to that the retrieved texts have a much more limited space, comparing to text space in the original pre-training set (*e.g*. LAION-400M), as the concepts are limited to the query classes from the target domain. Therefore, fine-tuning the text encoder may tend to collapse the pre-trained semantic space.

We advocate two tuning methods for model customization. Locked-text gated-image tuning has a strong adaptation power, and is efficient in the model customization stage, with fewer trainable parameters. Locked-text tuning is also an effective way of customizing the models to downstream tasks, without the need of adding extra parameters. By default, we use gated blocks for its superior performance and efficiency.

					IN1K	COCO R@1	
	Method	Visual	Text	#Train	Acc.	T2I	I2T
Direct	CLIP [112]	✗	✗	–	63.2	48.8	29.9
	Locked-Image [160]	✗	✓	63.4M	63.7	50.5	34.2
	Locked-Text	✓	✗	88.1M	66.9	51.1	36.1
	Full-model	✓	✓	151.3M	64.3	54.3	37.9
Gated	Locked-Text Gated-Image	✗	✓	18.9M	63.0	49.5	33.5
		✓	✗	42.5M	**68.6**	53.4	38.1
		✓	✓	89.8M	**68.7**	54.3	39.9

Table 2.6: Ablation: tuning strategy. For our model customization purpose, we advocate locked-text (gated-image) tuning methods in gray rows. ✓: trainable, ✗: locked.

Retrieval size. We observe that training with a small retrieval size is more likely to overfit. We find that a larger retrieval size generally yields better performance, and saturates at around 6-10M image-text pairs.

Retrieval Size	0	1M	3M	6M	10M
ImageNet-1K Accuracy	63.2	64.8	66.9	**68.6**	**68.6**

2.5 Conclusion

We presented React, a plug-and-play framework for leveraging large-scale image-text corpus as external knowledge to efficiently customize models on downstream tasks. Extensive experiments demonstrate its generality and effectiveness in image classification, image-text retrieval, object detection, and semantic segmentation, on more than 20 different downstream datasets. We highly advocate the model customization stage for building more transferable visual system for different downstream tasks.

3 VISUAL INSTRUCTION TUNING (LLAVA)

Despite the great adaptation performance REACT [87] brings about, creating a streamlined and steerable visual system still requires much effort and remains under-explored. On the other hand, natural language is a versatile media to allow human to steer and control the deep models on a variety of tasks. Instruction tuning large language models (LLMs) using machine-generated instruction-following data has been shown to improve zero-shot capabilities on new tasks, but the idea is less explored in the multimodal field. We present the first attempt to use language-only GPT-4 to generate multimodal language-image instruction-following data. By instruction tuning on such generated data, we introduce LLaVA: **L**arge **L**anguage **a**nd **V**ision **A**ssistant, an end-to-end trained large multimodal model that connects a vision encoder and an LLM for general-purpose visual and language understanding. To facilitate future research on visual instruction following, we construct two evaluation benchmarks with diverse and challenging application-oriented tasks. Our experiments show that LLaVA demonstrates impressive multimodal chat abilities, sometimes exhibiting the behaviors of multimodal GPT-4 on unseen images/instructions, and yields a 85.1% relative score compared with GPT-4 on a synthetic multimodal instruction-following dataset. When fine-tuned on Science QA, the synergy of LLaVA and GPT-4 achieves a new state-of-the-art accuracy of 92.53%. We make GPT-4 generated visual instruction tuning data, our model, and code publicly available.

3.1 Introduction

Humans interact with the world through many channels such as vision and language, as each individual channel has a unique advantage in representing and communicating certain concepts, and thus facilitates a better understanding of the world. One of the core aspirations in artificial intelligence is to develop a general-purpose assistant that can effectively follow multi-modal vision-and-language instructions, aligned with human intent to complete various real-world tasks in the wild [5, 75, 74].

To this end, the community has witnessed an emergent interest in developing language-augmented foundation vision models [75, 38], with strong capabilities in open-world visual understanding such as classification [112, 55, 159, 144, 109], detection [78, 169, 88], segmentation [73, 176, 161] and captioning [136, 77], as well as visual generation and editing [114, 115, 156, 37, 116, 82]. We refer readers to the *Computer Vision in the Wild* reading list for a more up-to-date literature compilation [28]. In this line of work, each task is solved independently by one single large vision model, with the task instruction implicitly considered in the model design. Further, language is only utilized to describe the image content. While this allows language to play an important role in mapping visual signals to language semantics—a common channel for human communication, it leads to models that usually have a fixed interface with limited interactivity and adaptability to the user's instructions.

Large language models (LLM), on the other hand, have shown that language can play a wider role: a universal interface for a general-purpose assistant, where various task instructions can be explicitly represented in language and guide the end-to-end trained neural assistant to switch to the task of interest to solve it. For example, the recent success of ChatGPT [103] and GPT-4 [104] have demonstrated the power of aligned LLMs in following human instructions, and have stimulated tremendous interest in developing open-source LLMs. Among them, LLaMA [132] is an open-source LLM that matches the performance of GPT-3. Alpaca [130], Vicuna [25], GPT-4-LLM [107] utilize various machine-generated high-quality instruction-following samples to improve the LLM's alignment ability, reporting impressive performance compared with proprietary LLMs.

Importantly, this line of work is *text-only*.

In this chapter, we present *visual instruction-tuning*, the first attempt to extend instruction-tuning to the language-image multimodal space, to pave the way towards building a general-purpose visual assistant. In particular, this chapter makes the following contributions:

- *Multimodal instruction-following data*. One key challenge is the lack of vision-language instruction-following data. We present a data reformation perspective and pipeline to convert image-text pairs into an appropriate instruction-following format, using ChatGPT/GPT-4.
- *Large multimodal models*. We develop a large multimodal model (LMM), by connecting the open-set visual encoder of CLIP [112] with the language decoder Vicuna [25], and fine-tuning end-to-end on our generated instructional vision-language data. Our empirical study validates the effectiveness of using generated data for LMM instruction-tuning, and suggests practical tips for building a general-purpose instruction-following visual agent. When ensembled with GPT-4, our approach achieves SoTA on the Science QA [92] multimodal reasoning dataset.
- *Multimodal instruction-following benchmark*. We present LLaVA-Bench with two challenging benchmarks, with a diverse selection of paired images, instructions and detailed annotations.
- *Open-source*. We release the following assets to the public: the generated multimodal instruction data, the codebase, the model checkpoints, and a visual chat demo.

3.2 Related Work

Multimodal Instruction-following Agents. In computer vision, existing works that build instruction-following agents can be broadly categorized into two classes: (i) End-to-end trained models, which are separately explored for each specific research topic. For example, the vision-language navigation task [4, 47] and Habitat [129] require the embodied AI agent to follow natural language instructions and take a sequence of actions to complete goals in visual environments. In the image editing domain, given an input image and a written instruction that tells the agent what to do, InstructPix2Pix [12] edits images by following the human instructions. (ii) A system that coordinates various models via LangChain [1] / LLMs [103], such as Visual ChatGPT [141], X-GPT [176], MM-REACT [148], VisProg [44], and ViperGPT [128]. While sharing the same goal in building instruction-following agents, we focus on developing an end-to-end trained language-vision multimodal model for *multiple* tasks.

Instruction Tuning. In the natural language processing (NLP) community, to enable LLMs such as GPT-3 [13], T5 [113], PaLM [26], and OPT [164] to follow natural language instructions and complete real-world tasks, researchers have explored methods for LLM instruction-tuning [106, 139, 138], leading to instruction-tuned counterparts such as InstructGPT [106]/ChatGPT [103], FLAN-T5 [27], FLAN-PaLM [27], and OPT-IML [56], respectively. It turns out that this simple approach can effectively improve the zero- and few-shot generalization abilities of LLMs. It is thus natural to borrow the idea from NLP to computer vision. More broadly, the teacher-student distillation ideas with foundation models have been studied in other topics such as image classification [34]. Flamingo [3] can be viewed as the GPT-3 moment in the multimodal domain, due to its strong performance on zero-shot task transfer and in-context-learning. Other LMMs trained on image-text pairs include BLIP-2 [77], FROMAGe [63], and KOSMOS-1 [52]. PaLM-E [33] is an LMM for embodied AI. Based on the recent "best" open-source LLM LLaMA, OpenFlamingo [6] and LLaMA-Adapter [162] are open-source efforts that enable LLaMA to use image inputs, paving the way to build open-source multimodal LLMs. While these models present promising task transfer generalization performance, they are not explicitly tuned with vision-language instruction data, and their performance in multimodal tasks usually falls short compared to language-only tasks. In this chapter, we aim to fill this gap and study its effectiveness. Finally, note that visual instruction tuning is different from visual prompt tuning [59]: the former aims to improve the model's instruction-following abilities, while the latter aims to improve the parameter-efficiency in model adaptation.

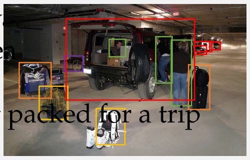
3.3 GPT-assisted Visual Instruction Data Generation

The community has witnessed a surge in the amount of public multimodal data such as image-text pairs, ranging from CC [15] to LAION [118]. However, when it comes to multimodal instruction-following data, the available amount is limited, partially because the process for creating such data is time-consuming and less well-defined when human crowd-scouring is considered. Inspired by the success of recent GPT models in text-annotation tasks [41], we propose to leverage ChatGPT/GPT-4 for multimodal instruction-following data collection, based on the widely existing image-pair data.

For an image $\mathbf{X}_{\mathrm{v}}$ and its associated caption $\mathbf{X}_{\mathrm{c}}$, it is natural to create a set of questions $\mathbf{X}_{\mathrm{q}}$ with the intent to instruct the assistant to describe the image content. We prompt GPT-4 to curate such a list of questions. Therefore, a simple way to expand an image-text pair to its instruction-following version is $\texttt{Human}: \mathbf{X}_{\mathrm{q}}\ \mathbf{X}_{\mathrm{v}}\texttt{<STOP>}\ \texttt{Assistant}: \mathbf{X}_{\mathrm{c}}\texttt{<STOP>}$. Though cheap to construct, this simple expanded version lacks diversity and in-depth reasoning in both the instructions and responses.

To mitigate this issue, we leverage language-only GPT-4 or ChatGPT

`Context type 1: Captions`
A group of people standing outside of a black vehicle with v
Luggage surrounds a vehicle in an underground parking are
People try to fit all of their luggage in an SUV.
The sport utility vehicle is parked in the public garage, being packed for a trip
Some people with luggage near a van that is transporting it.



`Context type 2: Boxes`
person: [0.681, 0.242, 0.774, 0.694], backpack: [0.384, 0.696, 0.485, 0.914], suitcase: ...<omitted>

`Response type 1: conversation`
Question: What type of vehicle is featured in the image?
Answer: The image features a black sport utility vehicle (SUV) ...<omitted>

`Response type 2: detailed description`
The image is an underground parking area with a black sport utility vehicle (SUV) parked. There are three people in the scene, with one person standing closer to the left side of the vehicle, another person in the middle, and the third person on the right side. They are all working together to pack their luggage into the SUV for a trip. ...<omitted>

`Response type 3: complex reasoning`
Question: What challenges do these people face?
Answer: In the image, a group of people is standing outside a black SUV in a parking area, surrounded by various pieces of luggage, including suitcases and backpacks. They are facing the challenge of fitting all their luggage into the black SUV. There are multiple suitcases and backpacks to be packed, which suggests that the group has a significant amount of belongings ...<omitted>

Table 3.1: One example to illustrate the instruction-following data. The top block shows the contexts such as captions and boxes used to prompt GPT, and the bottom block shows the three types of responses. Note that the visual image is not used to prompt GPT, we only show it here as a reference.

as the strong teacher (both accept only text as input), to create instruction-following data involving visual content. Specifically, in order to encode an image into its visual features to prompt a text-only GPT, we use two types of symbolic representations: (i) *Captions* typically describe the visual scene from various perspectives; (ii) *Bounding boxes* usually localize the objects in the scene, and each box encodes the object concept and its spatial location. One example is shown in the top block of Table 3.1.

This symbolic representation allows us to encode the image as an LLM-recognizable sequence. We use COCO images [83] and generate three types of instruction-following data. One example per type is shown in the bottom block of Table 3.1. For each type, we first manually design a few examples. They are the only human annotations we have during data collection, and are used as seed examples in in-context-learning to query GPT-4.

- *Conversation*. We design a conversation between the assistant and a person asking questions about this photo. The answers are in a tone as if the assistant is seeing the image and answering the question. A diverse set of questions are asked about the visual content of the image, including the object types, counting the objects, object actions, object locations, relative positions between objects. Only questions that have definite answers are considered.
- *Detailed description*. To include a rich and comprehensive description for an image, we create a list of questions with such an intent. We prompt GPT-4 then curate the list. For each image, we randomly sample one question from the list to ask GPT-4 to generate the detailed description.
- *Complex reasoning*. The above two types focus on the visual content itself, based on which we further create in-depth reasoning questions. The answers typically require a step-by-step reasoning process by following rigorous logic.

We collect 158K unique language-image instruction-following samples in total, including 58K in conversations, 23K in detailed description, and 77k in complex reasoning, respectively. We ablated the use of ChatGPT and GPT-4 in our early experiments, and found that GPT-4 consistently provides higher quality instruction-following data, such as spatial reasoning.

3.4 Visual Instruction Tuning

Architecture

The primary goal is to effectively leverage the capabilities of both the pre-trained LLM and visual model. The network archtecture is illustrated in Figure 3.1. We choose Vicuna [25] as our LLM $f_{\phi}(\cdot)$ parameterized by ϕ, as it has the best instruction following capabilities in language tasks among publicly available checkpoints [130, 25, 107].

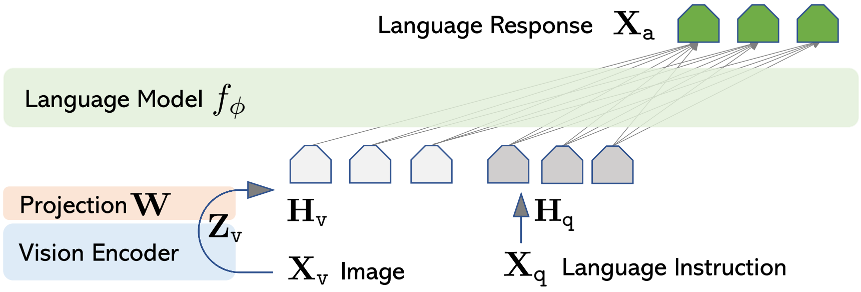


Figure 3.1: LLaVA network architecture.

For an input image $\mathbf{X}_v$, we consider the pre-trained CLIP visual encoder ViT-L/14 [112], which provides the visual feature $\mathbf{Z}_v = g(\mathbf{X}_v)$. The grid features before and after the last Transformer layer are considered in our experiments. We consider a simple linear layer to connect image features into the word embedding space. Specifically, we apply a trainable projection matrix $\mathbf{W}$ to convert $\mathbf{Z}_v$ into language embedding tokens $\mathbf{H}_v$, which have the same dimensionality as the word embedding space in the language model:

$$\mathbf{H}_v = \mathbf{W} \cdot \mathbf{Z}_v, \text{ with } \mathbf{Z}_v = g(\mathbf{X}_v) \tag{3.1}$$

Thus, we have a sequence of visual tokens $\mathbf{H}_v$. Note that our simple projection scheme is lightweight, which allows us to iterate data centric experiments quickly. More sophisticated schemes to connect the image and language representations can also be considered, such as gated cross-attention in Flamingo [3] and Q-former in BLIP-2 [77]. We leave exploring possibly more effective and sophisticated architecture designs for LLaVA as future work.

Training

For each image $\mathbf{X}_v$, we generate multi-turn conversation data $(\mathbf{X}_q^1, \mathbf{X}_a^1, \cdots, \mathbf{X}_q^T, \mathbf{X}_a^T)$, where T is the total number of turns. We organize them as a sequence, by treating all answers as the assistant's response, and the instruction $\mathbf{X}_{\texttt{instruct}}^t$ at the t-th turn as:

$$\mathbf{X}_{\texttt{instruct}}^t = \begin{cases} \text{Randomly choose } [\mathbf{X}_q^1, \mathbf{X}_v] \text{ or } [\mathbf{X}_v, \mathbf{X}_q^1], & \text{the first turn } t = 1 \\ \mathbf{X}_q^t, & \text{the remaining turns } t > 1 \end{cases} \tag{3.2}$$

This leads to the unified format for the multimodal instruction-following sequence illustrated in Table 3.2. We perform instruction-tuning of the LLM on the prediction tokens, using its original auto-regressive training objective.

Specifically, for a sequence of length L, we compute the probability of the target answers $\mathbf{X}_a$ by:

$$p(\mathbf{X}_a | \mathbf{X}_v, \mathbf{X}_{\texttt{instruct}}) = \prod_{i=1}^{L} p_\theta(x_i | \mathbf{X}_v, \mathbf{X}_{\texttt{instruct},<i}, \mathbf{X}_{a,<i}), \tag{3.3}$$

where θ is the trainable parameters, $\mathbf{X}_{\texttt{instruct},<i}$ and $\mathbf{X}_{a,<i}$ are the instruction and answer tokens in all turns before the current prediction token x_i, respectively. Please see Table 3.2 for an illustration of the prediction

$\mathbf{X}_{\texttt{system-message}}$ <STOP>
Human : $\mathbf{X}^1_{\texttt{instruct}}$ <STOP> Assistant: $\mathtt{X}^1_{\mathtt{a}}$ <STOP>
Human : $\mathbf{X}^2_{\texttt{instruct}}$ <STOP> Assistant: $\mathtt{X}^2_{\mathtt{a}}$ <STOP> $\cdots$

Table 3.2: The input sequence used to train the model. Only two conversation turns are illustrated here; in practice, the number of turns varies based on the instruction-following data. In our current implementation, we follow Vicuna-v0 [25] to set the system message $\mathbf{X}_{\texttt{system-message}}$ and we set `<STOP>` = `###`. The model is trained to predict the assistant answers and where to stop, and thus only green sequence/tokens are used to compute the loss in the auto-regressive model.

tokens. For the conditionals in (3.3), we explicitly add $\mathbf{X}_{\mathtt{v}}$ to emphasize the fact that the image is grounded for all answers, and we omit $\mathbf{X}_{\texttt{system-message}}$ and all previous `<STOP>` for better readability. For LLaVA model training, we consider a two-stage instruction-tuning procedure.

Stage 1: Pre-training for Feature Alignment. To strike a balance between concept coverage and training efficiency, we filter CC3M to 595K image-text pairs. These pairs are converted to the instruction-following data using the naive expansion method describe in Section 3.3. Each sample can be treated as a single-turn conversation. To construct the input $\mathbf{X}_{\texttt{instruct}}$ in (3.2), for an image $\mathbf{X}_{\mathtt{v}}$, a question $\mathbf{X}_{\mathtt{q}}$ is randomly sampled, which is a language instruction to request the assistant to describe the image briefly. The ground-truth prediction answer $\mathbf{X}_{\mathtt{a}}$ is the original caption. In training, we keep both the visual encoder and LLM weights frozen, and maximize the likelihood of (3.3) with trainable parameters $\theta = \mathbf{W}$ (the projection matrix) only. In this way, the image features $\mathbf{H}_{\mathtt{v}}$ can be aligned with the pre-trained LLM word embedding. This stage can be understood as training a compatible visual tokenizer for the frozen LLM.

Stage 2: Fine-tuning End-to-End. We always keep the visual encoder weights frozen, and continue to update both the pre-trained weights of the projection layer and LLM in LLaVA; i.e., the trainable parameters are $\theta = \{\mathbf{W}, \boldsymbol{\phi}\}$ in (3.3). We consider two specific use case scenarios:

- *Multimodal Chatbot*. We develop a Chatbot by fine-tuning on the 158K language-image instruction-following data in Section 3.3. Among the three types of responses, conversation is multi-turn while the other two are single-turn. They are uniformly sampled in training.
- *Science QA*. We study our method on the ScienceQA benchmark [92], the first large-scale multimodal science question dataset that annotates the answers with detailed lectures and explanations. Each question is provided a context in the form of natural language or an image. The assistant provides the reasoning process in natural language and selects the answer among multiple choices. For training in (3.2), we organize the data as a single turn conversation, the question & context as $\mathbf{X}_{\texttt{instruct}}$, and reasoning & answer as $\mathbf{X}_{\texttt{a}}$.

3.5 Experiments

We assess the performance of LLaVA in instruction-following and visual reasoning capabilities with two primary experimental settings: multimodal chatbot and the ScienceQA dataset, respectively. We train all models with 8× A100s, following Vicuna's hyperparameters [25]. We pre-train our model on the filtered CC-595K subset for 1 epoch with a learning rate of 2e-3 and a batch size of 128, and fine-tune on the proposed LLaVA-Instruct-158K dataset for 3 epochs, with a learning rate of 2e-5 and a batch size of 32.

Multimodal Chatbot

We developed a chatbot demo to show the image understanding and conversation abilities of LLaVA, and to study how well LLaVA is able to

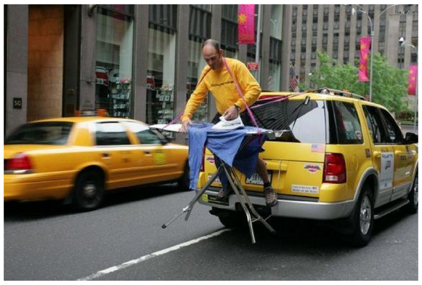
	Visual input example, Extreme Ironing:
	 Src: `https://www.barnorama.com/wp-content/uploads/2016/12/03-Confusing-Pictures.jpg`
User	What is unusual about this image?
LLaVA	The unusual aspect of this image is a man ironing clothes on the back of a minivan or van. This is not a typical place to perform this activity, as one would usually iron clothes in a more stationary and safe location, such as a home, using a regular ironing board. The scene depicted in the image is peculiar as it involves a makeshift ironing setup on a vehicle, which can be both unsafe and unconventional. Additionally, it is not clear how the man is able to maintain balance and stability while ironing clothes in such an unstable environment.
	[*Start a new conversation, and clear the history*]
User	What's happening in the scene?
LLaVA	The image depicts a man standing on top of a yellow SUV in a busy city street. He is holding a portable ladder, seemingly ironing clothes while standing on the vehicle. Around the scene, there are other cars, a traffic light, and a fire hydrant. Additionally, there are two other people in the vicinity, one close to the man on the car and another further away. The man on the ladder appears to be attracting attention due to his unconventional choice of ironing his clothes on top of a moving car. The city street around him is bustling with activity, adding to the unique nature of the scene.
User	What is unusual about this image?
GPT-4 [104]	The unusual thing about this image is that a man is ironing clothes on an ironing board attached to the roof of a moving taxi.
User	What is unusual about this image?
BLIP-2	a man is sitting on the back of a yellow cab
User	What is unusual about this image?
OpenFlamingo	The man is drying his clothes on the hood of his car.

Table 3.3: Example prompt from GPT-4 paper [104] to compare visual reasoning and chat capabilities. Compared to BLIP-2 [77] and Open-Flamingo [6], LLaVA accurately follows the user's instructions, instead of simply describing the scene. LLaVA offers a more comprehensive response than GPT-4. Even when merely asked to describe the image, LLaVA identifies atypical aspects of the image.

digest visual inputs and exhibit instruction-following capabilities. We first use the examples in the original GPT-4 paper [104], shown in Table 3.3, that require in-depth image understanding. For comparisons, we quote the prompt and response of the multimodal GPT-4 from their paper, and query BLIP-2 and OpenFlamingo model checkpoints to get their response.

Surprisingly, although LLaVA is trained with a small multimodal instruction-following dataset (~80K unique images), it demonstrates quite similar reasoning results with multimodal GPT-4 on these examples. Note that while these images are out-of-domain for LLaVA, LLaVA is still able to understand the scenes and follow the question instruction to provide a reasonable response. In contrast, BLIP-2 and OpenFlamingo focus on describing the image, instead of following the user instruction to answer in an appropriate manner.

Quantitative Evaluation. To gain a systematic understanding of the performance of LLaVA, we propose a quantitative metric to measure the model's instruction-following capability on multimodal data. Inspired by [25], we leverage GPT-4 to measure the quality of generated responses. Specifically, we create triplets consisting of image, ground-truth textual descriptions, and question. The candidate models (*e.g.* LLaVA) predict the answers based on the question and the image. To provide an *approximate theoretical upper bound*, we create a reference prediction based on the question and the *ground-truth* textual descriptions, using the text-only GPT-4. After obtaining the responses from both models, we feed the question, visual information (in the format of textual descriptions), and the generated responses from both assistants, to the judge (*i.e.* text-only GPT-4). It evaluates the helpfulness, relevance, accuracy, and level of detail of the responses from the assistants, and gives an overall score on a scale of 1 to 10, where a higher score indicates better overall performance. It is also asked to provide a comprehensive explanation for the evaluation, for us to better understand the models. We report relative scores *w.r.t.* the text-only

	Conversation	Detail description	Complex reasoning	**All**
Full data	83.1	75.3	96.5	85.1
Detail + Complex	81.5 (-1.6)	73.3 (-2.0)	90.8 (-5.7)	81.9 (-3.2)
Conv + 5% Detail + 10% Complex	81.0 (-2.1)	68.4 (-7.1)	91.5 (-5.0)	80.5 (-4.4)
Conversation	76.5 (-6.6)	59.8 (-16.2)	84.9 (-12.4)	73.8 (-11.3)
No Instruction Tuning	22.0 (-61.1)	24.0 (-51.3)	18.5 (-78.0)	21.5 (-63.6)

Table 3.4: Ablation on LLaVA-Bench (COCO) with different training data. We report relative scores *w.r.t.* a text-only GPT-4 model that uses ground truth image captions and bounding boxes as visual input. We prompt GPT-4 with the answers from our model outputs and the answers by GPT-4 (text-only), and let it compare between both responses and give a rating with an explanation.

	Conversation	Detail description	Complex reasoning	All
OpenFlamingo [6]	19.3 ± 0.5	19.0 ± 0.5	19.1 ± 0.7	19.1 ± 0.4
BLIP-2 [77]	54.6 ± 1.4	29.1 ± 1.2	32.9 ± 0.7	38.1 ± 1.0
LLaVA	57.3 ± 1.9	52.5 ± 6.3	81.7 ± 1.8	67.3 ± 2.0
LLaVA†	58.8 ± 0.6	49.2 ± 0.8	81.4 ± 0.3	66.7 ± 0.3

Table 3.5: Instruction-following capability comparison using relative scores on LLaVA-Bench (In-the-Wild). The results are reported in the format of *mean* ± *std*. For the first three rows, we report three inference runs. LLaVA performs significantly better than others. † For a given set of LLaVA decoding sequences, we evaluate by querying GPT-4 three times; GPT-4 gives a consistent evaluation.

GPT-4 model that uses the textural ground truth description as visual input. We create two benchmarks to evaluate the model's performance.

LLaVA-Bench (COCO). We randomly select 30 images from COCO-Val-2014, and for each image, we generate three types of questions (conversation, detailed description, complex reasoning) using the proposed data generation pipeline in Sec. 3.3, totaling 90 questions. This benchmark studies the model's alignment behavior and capabilities with consistent visual inputs. We vary the training datasets to study the effectiveness of different types of instruction-following data, and show the results in Table 4.4. First, with instruction tuning, the model's ability of following user instructions improves significantly by over 50 points. Second, adding a small amount of detailed description and complex reasoning questions contributes to a considerable improvement of the model's overall capability by 7 points. Furthermore, it also improves the model's performance on conversational questions, suggesting that improvements in reasoning capabilities complement conversational abilities. Finally, we show that having all three types of data yields the best performance at 85.1%.

LLaVA-Bench (In-the-Wild). To evaluate the model's capability in more challenging tasks and generalizability to novel domains, we collect a diverse set of 24 images with 60 questions in total, including indoor and outdoor scenes, memes, paintings, sketches, *etc*., and associate each image with a highly-detailed and manually-curated description and a proper selection of questions. We compare LLaVA, BLIP, and OpenFlamingo in Table 3.5. Thanks to visual instruction tuning, LLaVA achieves significantly better performance compared with BLIP-2 (+29%) and OpenFlamingo (+48%). Compared to the text-only GPT-4 that has access to ground-truth labels, LLaVA achieves an impressive 81.7% performance on complex reasoning questions, with an overall score of 67.3%.

Limitations. This LLaVA-Bench (In-the-Wild) is designed to be challenging and to reveal a model's weaknesses. We provide two examples with associated captions and questions in Table 3.6. For the ramen example (left), to correctly answer the name of the restaurant, it requires the model to have a large knowledge coverage and multilingual understanding capability; to correctly describe the side dishes, the model may need to retrieve relevant multimodal information from Internet. For the fridge example (right), perceiving the correct brand of the yogurt requires the model to process high resolution images and possess extensive knowledge coverage. We also observed an interesting failure of LLaVA, as it responds with *yes* when asked if strawberry-flavored yogurt is present, even though the fridge contains only yogurt *and* strawberries. This indicates that, at times,

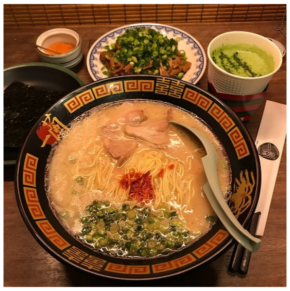
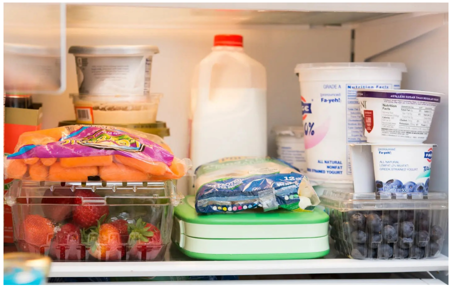
	Challenging examples from LLaVA-Bench (In-the-Wild):	
	 ICHIRAN Ramen [source]	 Filled fridge [source]
Annotation	A close-up photo of a meal at ICHIRAN. The chashu ramen bowl with a spoon is placed in the center. The ramen is seasoned with chili sauce, chopped scallions, and served with two pieces of chashu. Chopsticks are placed to the right of the bowl, still in their paper wrap, not yet opened. The ramen is also served with nori on the left. On top, from left to right, the following sides are served: a bowl of orange spice (possibly garlic sauce), a plate of smoke-flavored stewed pork with chopped scallions, and a cup of matcha green tea.	An open refrigerator filled with a variety of food items. In the left part of the compartment, towards the front, there is a plastic box of strawberries with a small bag of baby carrots on top. Towards the back, there is a stack of sauce containers. In the middle part of the compartment, towards the front, there is a green plastic box, and there is an unidentified plastic bag placed on it. Towards the back, there is a carton of milk. In the right part of the compartment, towards the front, there is a box of blueberries with three yogurts stacked on top. The large bottle of yogurt is Fage non-fat yogurt, and one of the smaller cups is Fage blueberry yogurt. The brand and flavor of the other smaller cup are unknown. Towards the back, there is a container with an unknown content.
Question 1	What's the name of the restaurant?	What is the brand of the blueberry-flavored yogurt?
Question 2	Describe this photo in detail.	Is there strawberry-flavored yogurt in the fridge?

Table 3.6: Challenging examples from LLaVA-Bench (In-the-Wild), we provide extremely-detailed annotation for each image for an accurate evaluation. Some questions require the model to extract details from high resolution image and to have a broad knowledge coverage.

LLaVA perceives the image as a "bag of patches", failing to grasp the complex semantics within the image. We hope LLaVA serves as a solid baseline on the benchmarks, on which our findings can inspire future work in developing more capable LMMs.

ScienceQA

ScienceQA [92] contains 21k multimodal multiple choice questions with rich domain diversity across 3 subjects, 26 topics, 127 categories, and 379 skills. The benchmark dataset is split into training, validation, and test splits with 12726, 4241, and 4241 examples, respectively. We consider two representative methods, including GPT-3.5 model (`text-davinci-002`) with and without chain-of-thought (CoT), LLaMA-Adapter [162], as well as multimodal chain-of-thought (MM-CoT) [166], which is the current SoTA method on this dataset. For more baseline numbers, please see [92].

The results are reported in Table 3.7. For LLaVA, we use the visual features before the last layer, ask the model to first predict reasons and then the answer, and train it for 12 epochs. It yields 90.92% accuracy, which is quite close to the SoTA 91.68%. To explore the limit of LLMs, we also prompt GPT-4 using 2-shot in-context-learning and achieve 82.69% accuracy, which is a 7.52% absolute gain compared with 75.17% from GPT-3.5. For a substantial number of questions, we note that GPT-4 fails simply because it reports that there is insufficient context such as images or plots. We consider two schemes to combine the outcomes from our model and GPT-4. (i) *A GPT-4 complement*. Whenever GPT-4 fails to provide answers, we use the prediction from our method. This schemes yields 90.97% accuracy, which is almost the same as applying our method alone. (ii) *GPT-4 as the judge*. Whenever GPT-4 and LLaVA produce different answers, we prompt GPT-4 again, asking it to provide its own final answer based on the question and two outcomes. The spirit is similar with CoT, but with the external knowledge from the other model. Surprisingly, this

Method	Subject			Context Modality			Grade		Average
	NAT	SOC	LAN	TXT	IMG	NO	G1-6	G7-12	
Representative & SoTA methods with numbers reported in the literature									
Human [92]	90.23	84.97	87.48	89.60	87.50	88.10	91.59	82.42	88.40
GPT-3.5 [92]	74.64	69.74	76.00	74.44	67.28	77.42	76.80	68.89	73.97
GPT-3.5 w/ CoT [92]	75.44	70.87	78.09	74.68	67.43	79.93	78.23	69.68	75.17
LLaMA-Adapter [162]	84.37	88.30	84.36	83.72	80.32	86.90	85.83	84.05	85.19
MM-CoT$_{Base}$ [166]	87.52	77.17	85.82	87.88	82.90	86.83	84.65	85.37	84.91
MM-CoT$_{Large}$ [166]	95.91	82.00	90.82	95.26	88.80	92.89	92.44	90.31	91.68
Results with our own experiment runs									
GPT-4[†]	84.06	73.45	87.36	81.87	70.75	90.73	84.69	79.10	82.69
LLaVA	90.36	95.95	88.00	89.49	88.00	90.66	90.93	90.90	90.92
LLaVA+GPT-4[†] (complement)	90.36	95.50	88.55	89.05	87.80	91.08	92.22	88.73	90.97
LLaVA+GPT-4[†] (judge)	91.56	96.74	91.09	90.62	88.99	93.52	92.73	92.16	**92.53**

Table 3.7: Accuracy (%) on Science QA dataset. Question categories: NAT = natural science, SOC = social science, LAN = language science, TXT = text context, IMG = image context, NO = no context, G1-6 = grades 1-6, G7-12 = grades 7-12. [†]Text-only GPT-4, our eval. Our novel model ensembling with the text-only GPT-4 consistently improves the model's performance under all categories, setting the new SoTA performance.

scheme is able to provide consistent improvement over all question classes, and achieves a new SoTA accuracy of 92.53%. Interestingly, the text-only GPT-4, which cannot process images, improves the overall performance of the model on questions that have an image as context. This is because some of these questions do not actually require the image context for a correct answer. The GPT-4 judge can identify such cases and correct some of the errors that LLaVA makes. To the best of our knowledge, this is the first time that GPT-4 is used for model ensembling. We hope this finding can encourage future research to explore more effective methods to leverage LLMs for model ensembling.

Ablations. We ablate several design choices on ScienceQA in Table 3.8. (i) *Visual features*. We tried using the last layer feature from CLIP vision encoder, which yields 89.96% and is 0.96% lower than the feature before the last layer. We hypothesize that this is because CLIP's last layer features may focus more on global and abstract image properties compared to the

Visual features	Before	Last
Best variant	90.92	89.96 (-0.96)
Predict answer first	-	89.77 (-1.15)
Training from scratch	85.81 (-5.11)	-
7B model size	89.84 (-1.08)	-

Table 3.8: Design choice ablations (%). The difference with the best variant is reported in red text.

layer before it, which can focus more on localized properties that are useful for understanding specific image details. (ii) *Chain-of-thought*. To decide the order between the answer and reasoning process in the model prediction, we run both variants and observe that answer-first reports the best number 89.77% accuracy in 12 epochs, while reasoning-first can quickly reach 89.77% accuracy in 6 epochs, but no further improvement with more training. Training the model for 24 epochs does not improve the performance. We conclude that CoT-like reasoning-first strategy can largely improve convergence, but contributes relatively little to the final performance. (iii) *Pre-training*. We skip pre-training and directly train on Science QA from scratch – performance drops to 85.81% accuracy. The 5.11% absolute degradation indicates the importance of our pre-training stage, in aligning multimodal features while preserving the vast pre-trained knowledge. (iv) *Model size*. We keep all configurations the same as our best 13B model, and train a 7B model. This yields 89.84% accuracy, which is 1.08% lower than 90.92%, demonstrating the importance of model scale.

3.6 Conclusion

This chapter demonstrated the effectiveness of visual instruction tuning. We presented an automatic pipeline to create language-image instruction-following data, based on which we train LLaVA, a multimodal model to follow human intent to complete visual tasks. It achieves the new SoTA accuracy when fine-tuned on ScienceQA, and excellent visual chat capabilities when fine-tuned on multimodal chat data. Besides, we present the first benchmark to study multimodal instruction-following capability. This chapter is an initial step in visual instruction tuning, and mainly focuses on real-life tasks. For more quantitative results of LLaVA on academic benchmarks, please refer to the improved baselines with visual instruction tuning [85]. We hope our work can inspire future research on building more capable multimodal models.

4 IMPROVED BASELINES WITH VISUAL INSTRUCTION TUNING

Instruction-following large multimodal models (LMM) like LLaVA have shown encouraging progress with visual instruction tuning. In this chapter, we present the first systematic study to investigate the design choices of LMMs in a controlled setting under the LLaVA framework. We show that the fully-connected vision-language connector in LLaVA is surprisingly powerful and data-efficient. With simple modifications to LLaVA, namely, using CLIP-ViT-L-336px with an MLP projection and adding academic-task-oriented VQA data with response formatting prompts, we establish stronger baselines that achieve state-of-the-art across 11 benchmarks. Our final 13B checkpoint uses merely 1.2M publicly available data, and finishes full training in ~1 day on a single 8-A100 node. Furthermore, we present some early exploration of open problems in LMMs, including scaling to higher resolution inputs, compositional capabilities, and model hallucination, etc. We hope this makes state-of-the-art LMM research more accessible. Code and model are publicly available.

4.1 Introduction

Large multimodal models (LMMs) have become increasingly popular in the research community, as they are the key building blocks towards general-purpose assistants [74, 105, 3]. Recent studies on LMMs are converging on a central concept known as visual instruction tuning [86]. The results are promising, *e.g*. LLaVA [86] and MiniGPT-4 [175] demonstrate impressive results on natural instruction-following and visual reasoning capabilities. To better understand the capability of LMMs, multiple benchmarks [90, 157, 72, 81, 36] have been proposed. Recent works further demonstrate improved performance by scaling up the pretraining

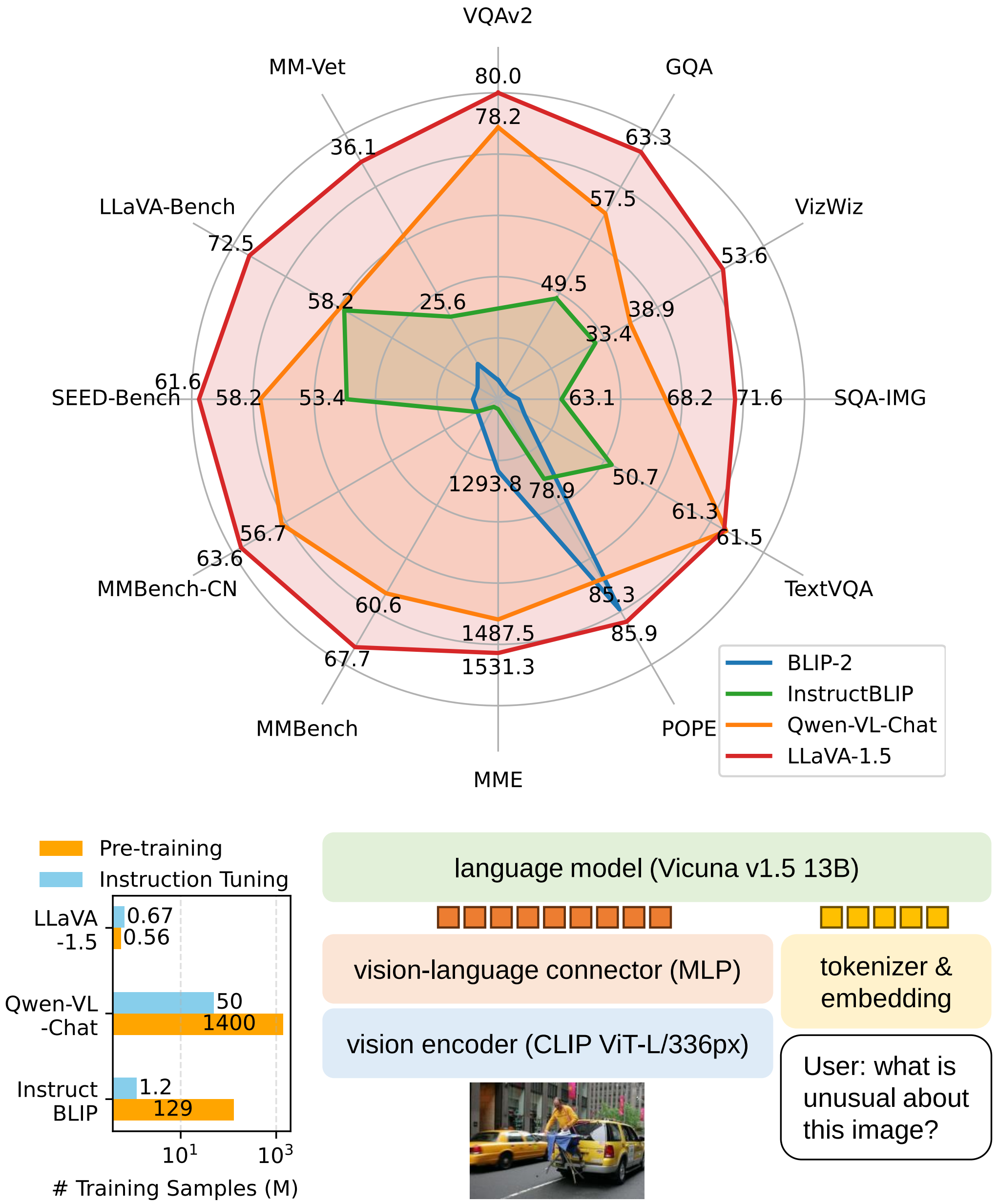


Figure 4.1: **LLaVA-1.5** achieves SoTA on a broad range of 11 tasks (Top), with high training sample efficiency (Left) and simple modifications to LLaVA (Right): an MLP connector and including academic-task-oriented data with response formatting prompts.

data [29, 7, 152], instruction-following data [167, 71, 29, 42], visual encoders [7], or language models [93], respectively. The LLaVA architecture is also leveraged in different downstream tasks and domains, including region-level [18, 163] and pixel-level [67, 137] understanding, biomedical assistants [76], image generation [9], adversarial studies [14, 168].

However, despite many benchmarks and developments, it still remains unclear what the best recipe is to train LMMs towards the goal of general-purpose assistants. For example, LLaVA [86] excels in conversational-style visual reasoning and even outperforms later approaches like InstructBLIP [29] on such benchmarks [157], while InstructBLIP excels in traditional VQA benchmarks that demands single-word or short answers. Given the significant differences in the model architecture and training data between them, the root cause of the disparity in their capabilities remains elusive, despite conjectures [157, 90]: the amount of training data, the usage of resamplers like Qformer [77], *etc*. To this end, we present the first systematic study to investigate the design choices of LMMs in a controlled setting. Our study originates from LLaVA and builds a road map by carefully making effective contributions from the perspectives of the input, model, and data.

First, we unveil that the fully-connected vision-language connector in LLaVA is surprisingly powerful and data-efficient, and we establish stronger and more feasible baselines built upon the LLaVA framework. We report that two simple improvements, namely, an MLP cross-modal connector and incorporating academic task related data such as VQA, are orthogonal to the framework of LLaVA, and when used with LLaVA, lead to better multimodal understanding capabilities. In contrast to InstructBLIP [29] or Qwen-VL [7], which trains specially designed visual resamplers on hundreds of millions or even billions of image-text paired data, LLaVA uses one of the simplest architecture design for LMMs and requires only training a simple fully-connected projection layer on merely

600K image-text pairs. Our final model can finish training in ~1 day on a single 8-A100 machine and achieves state-of-the-art results on a wide range of benchmarks. Moreover, unlike Qwen-VL [7] that includes in-house data in training, LLaVA utilizes only publicly available data.

Next, we delve into an early exploration of other open problems of large multimodal models. Our findings include: (1) Scaling to high-resolution image inputs. We show that LLaVA's architecture is versatile in scaling to higher resolutions by simply dividing images into grids and maintains its data efficiency; with the increased resolution, it improves the model's detailed perception capabilities and reduces hallucination. (2) Compositional capabilities. We find that large multimodal models are capable of generalizing to compositional capabilities. For example, training on long-form language reasoning together with shorter visual reasoning can improve the model's writing capability for multimodal questions. (3) Data efficiency. We show that randomly downsampling LLaVA's training data mixture by up to 75% does not significantly decrease the model's performance, suggesting that the possibility of a more sophisticated dataset compression strategy can further improve LLaVA's already efficient training pipeline. (4) Data scaling. We provide empirical evidence for the scaling of data granularity in conjunction with the model's capability is crucial for an improved capability without introducing artifacts like hallucination.

In sum, we perform a systematic study on the training of large multimodal models, and introduce a simple yet effective approach to balance the multitask learning and effective scaling for large multimodal models. Our improved baselines, LLaVA-1.5, uses only *public* data, achieves the state-of-the-art on a broad range of 11 tasks, and is significantly more data-efficient than previous approaches. By rethinking the conventional approaches and exploring the open problems in visual instruction tuning, we pave the way for more robust and capable systems for LMMs. We hope these improved and easily-reproducible baselines will provide a reference for future research in open-source LMMs.

4.2 Related Work

Instruction-following large multimodal models (LMMs). Common architectures include a pre-trained visual backbone to encode visual features, a pre-trained large language model (LLM) to comprehend the user instructions and produce responses, and a vision-language cross-modal connector to align the vision encoder outputs to the language models. As shown in Fig. 4.1, LLaVA [86] is perhaps the simplest architecture for LMMs. Optionally, visual resamplers (*e.g.* Qformer [77]) are used to reduce the number of visual patches [175, 29, 7]. Training an instruction-following LMM usually follows a two-stage protocol. First, the vision-language alignment pretraining stage leverages image-text pairs to align the visual features with the language model's word embedding space. Earlier works utilize relatively few image-text pairs (*e.g.* ~600K [86] or ~6M [175]), while some recent works pretrain the vision-language connector for a specific language model on a large amount of image-text pairs (*e.g.* 129M [29] and 1.4B [7]), to maximize the LMM's performance. Second, the visual instruction tuning stage tunes the model on visual instructions [86], to enable the model to follow users' diverse requests on instructions that involve the visual contents. Dealing with higher resolution with grids in LMM are studied in con-current works [70, 2, 151].

Multimodal instruction-following data. In NLP, studies show that the quality of instruction-following data largely affects the capability of the resulting instruction-following models [172]. For visual instruction tuning, LLaVA [86] is the pioneer to leverage text-only GPT-4 to expand the existing COCO [84] bounding box and caption dataset to a multimodal instruction-following dataset that contains three types of instruction-following data: conversational-style QA, detailed description, and complex reasoning. LLaVA's pipeline has been employed to expand to textual understanding [165], million-scales [167], and region-level conversations [18]. InstructBLIP [29] incorporates academic-task-oriented VQA datasets to further enhance the model's visual capabilities. Conversely, [16] identifies that such naive data merging can result in models that tend to overfit to VQA datasets and thus are unable to participate in natural conversations. The authors further propose to leverage the LLaVA pipeline to convert VQA datasets to a conversational style. While this proves effective for training, it introduces added complexities in data scaling. However, in NLP, the FLAN family [140, 27] shows that adding a large number of academic language tasks for instruction tuning can effectively improve the generalization ability. In light of this, we consider investigating the root cause of the inability to balance between natural conversations and academic tasks in multimodal models.

4.3 Approach

Preliminaries

As the seminal work of visual instruction tuning, LLaVA [86] showcases commendable proficiency in visual reasoning capabilities, surpassing even more recent models on diverse benchmarks [157, 8] for real-life visual instruction-following tasks. LLaVA uses a single linear layer to project the visual features to language space, and optimizes the whole LLM for visual instruction tuning. However, LLaVA falls short on academic benchmarks that typically require short-form answers (*e.g.* single-word), and tends to answer *yes* for yes/no questions due to the lack of such data in the training distribution.

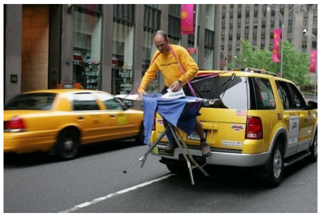
On the other hand, InstructBLIP [29] is the pioneer to incorporate academic-task-oriented datasets like VQA-v2 [43] along with LLaVA-

Instruct [86], and demonstrates improved performance on VQA benchmarks. It pretrains Qformer [77] on 129M image-text pairs and only finetunes the instruction-aware Qformer for visual instruction tuning. However, recent studies [16, 157] show that it does not perform as well as LLaVA on engaging in real-life visual conversation tasks. More specifically, as shown in Table 4.1a, it can overfit to VQA training sets with short-answers, even on requests that require detailed responses.

Response Format Prompting

We find that the inability [16] to balance between short- and long-form VQA for approaches like InstructBLIP [29], which leverages instruction following data that includes both natural responses and short-answers, is mainly due to the following reasons. First, *ambiguous prompts on the response format*. For example, *Q: {Question} A: {Answer}*. Such prompts do not clearly indicate the desired output format, and can overfit an LLM behaviorally to short-form answers even for natural visual conversations. Second, *not finetuning the LLM*. The first issue is worsened by InstructBLIP only finetuning the Qformer for instruction-tuning. It requires the Qformer's visual output tokens to control the length of the LLM's output to be either long-form or short-form, as in prefix tuning [79], but Qformer may lack the capability of properly doing so, due to its limited capacity compared with LLMs like LLaMA.

Thus, to enable LLaVA to better handle short-form answers while addressing the issues of InstructBLIP, we propose to use a single response formatting prompt that clearly indicates the output format. It is appended at the end of VQA questions when promoting short answers: *Answer the question using a single word or phrase.* We find that when the LLM is *finetuned* with such prompts, LLaVA is able to properly adjust the output format according to the user's instructions (see Table 4.1b), and does not require additional processing of the VQA answers using ChatGPT [16], which

	Visual input example, Multitask Balancing Problem:
	
User	Is this unusual? Please explain in detail.
InstructBLIP	yes

(a) Example of InstructBLIP [29] (Vicuna-13B) having difficulty balancing between short- and long-form answers.

	Visual input example, Different Format Prompts:
Normal prompt	What is the color of the shirt that the man is wearing?
Response	The man is wearing a yellow shirt.
Ambiguous prompt	Q: What is the color of the shirt that the man is wearing? A:
Response	The man is wearing a yellow shirt.
Formatting prompt	What is the color of the shirt that the man is wearing? **Answer the question using a single word or phrase.**
Response	Yellow.

(b) Comparison of how different prompts regularize the output format. The results are obtained zero-shot directly after LLaVA undergoes the first-stage vision-language alignment pretraining, without the second-stage visual instruction tuning.

Table 4.1: Visual input example to illustrate the challenge of (a) multitask balancing and (b) different format prompts. The same image input is used.

further enables scaling to various data sources. As shown in Table 4.2, by merely including VQAv2 [43] in training, LLaVA's performance on MME significantly improves (1323.8 *vs* 809.6) and outperforms InstructBLIP by 111 points.

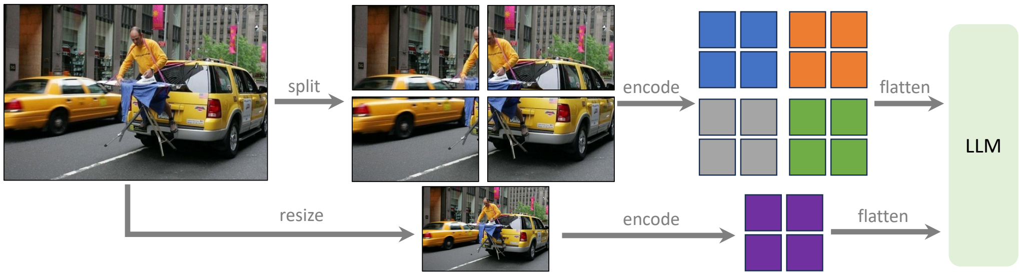


Figure 4.2: **LLaVA-1.5-HD.** Scaling LLaVA-1.5 to higher resolutions by splitting the image into grids and encoding them independently. This allows the model to scale to any resolution, without performing positional embedding interpolation for ViTs. We additionally concatenate the feature of a downsampled image to provide the LLM with a global context.

	Method	LLM	Res.	GQA	MME	MM-Vet
	InstructBLIP	14B	224	49.5	1212.8	25.6
	Only using a subset of InstructBLIP training data					
0	**LLaVA**	7B	224	–	809.6	25.5
1	+VQA-v2	7B	224	47.0	1197.0	27.7
2	+Format prompt	7B	224	46.8	1323.8	26.3
3	+MLP VL connector	7B	224	47.3	1355.2	27.8
4	+OKVQA/OCR	7B	224	50.0	1377.6	29.6
	Additional scaling					
5	+Region-level VQA	7B	224	50.3	1426.5	30.8
6	+Scale up resolution	7B	336	51.4	1450	30.3
7	+GQA	7B	336	62.0*	1469.2	30.7
8	+ShareGPT	7B	336	62.0*	1510.7	31.1
9	+Scale up LLM	13B	336	**63.3***	**1531.3**	**36.1**

Table 4.2: **Scaling results** on 🟦 data, 🟥 model, and 🟨 resolution. We choose to conduct experiments on GQA [53], MME [36], and MM-Vet [157] to examine the representative capabilities of VQA with short answers, VQA with output formatting, and natural visual conversations, respectively. *Training images of GQA were observed during training.

Scaling the Data and Model

MLP vision-language connector. Inspired by the improved performance in self-supervised learning by changing from a linear projection to an MLP [19, 24], we find that improving the vision-language connector's representation power with a two-layer MLP can improve LLaVA's multimodal capabilities, compared with the original linear projection.

Academic task oriented data. We further include additional academic-task-oriented VQA datasets for VQA, OCR, and region-level perception, to enhance the model's capabilities in various ways, as shown in Table 4.2. We first include four additional datasets that are used in InstructBLIP: open-knowledge VQA (OKVQA [98], A-OKVQA [120]) and OCR (OCRVQA [99], TextCaps [125]). A-OKVQA is converted to multiple choice questions and a specific response formatting prompt is used: *Answer with the option's letter from the given choices directly*. With only a subset of the datasets InstructBLIP uses, LLaVA already surpasses it on all three tasks in Table 4.2, suggesting LLaVA's effective design. Furthermore, we find further adding region-level VQA datasets (Visual Genome [66], RefCOCO [61, 96]) improves the model's capability of localizing fine-grained visual details.

Additional scaling. We further scale up the input image resolution to 336^2 to allow the LLM to clearly "see" the details of images, by swapping the vision encoder to CLIP-ViT-L-336px (the highest resolution available for CLIP). In addition, we add the GQA dataset as an additional visual knowledge source. We also incorporate ShareGPT [121] data and scale up the LLM to 13B as in [7, 93, 18]. Results on MM-Vet shows the most significant improvement when scaling the LLM to 13B, suggesting the importance of the base LLM's capability for visual conversations.

LLaVA-1.5. We denote this final model with all the modifications as LLaVA-1.5 (the last two rows in Table 4.2), which achieves an impressive

Method	LLM	Image Size	Sample Size		VQAv2 [43]	GQA [53]	VisWiz [45]	SciQA-IMG [92]	TextVQA [126]
			Pretrain	Finetune					
BLIP-2 [77]	Vicuna-13B	224^2	129M	-	65.0	41	19.6	61	42.5
InstructBLIP [29]	Vicuna-7B	224^2	129M	1.2M	–	49.2	34.5	60.5	50.1
InstructBLIP [29]	Vicuna-13B	224^2	129M	1.2M	–	49.5	33.4	63.1	50.7
Shikra [18]	Vicuna-13B	224^2	600K	5.5M	77.4^*	–	–	–	–
IDEFICS-9B [54]	LLaMA-7B	224^2	353M	1M	50.9	38.4	35.5	–	25.9
IDEFICS-80B [54]	LLaMA-65B	224^2	353M	1M	60.0	45.2	36.0	–	30.9
Qwen-VL [7]	Qwen-7B	448^2	$1.4B^\dagger$	$50M^\dagger$	78.8^*	59.3^*	35.2	67.1	**63.8***
Qwen-VL-Chat [7]	Qwen-7B	448^2	$1.4B^*$	$50M^\dagger$	78.2^*	57.5^*	38.9	68.2	61.5^*
LLaVA-1.5	Vicuna-7B	336^2	**558K**	**665K**	78.5^*	62.0^*	50.0	66.8	58.2
LLaVA-1.5	Vicuna-13B	336^2	**558K**	**665K**	**80.0***	**63.3***	**53.6**	**71.6**	61.3
LLaVA-1.5-HD	Vicuna-13B	448^2	**558K**	**665K**	**81.8***	**64.7***	**57.5**	71.0	62.5
Specialist SOTA: PaLI-X-55B [22]					86.1^*	72.1^*	70.9^*	–	71.4^*

Table 4.3: **Comparison with SoTA methods on academic-task-oriented datasets.** LLaVA-1.5 achieves the best performance on 4/5 benchmarks, and ranks the second on the other. *The training images/annotations of the datasets are observed during training. †Includes in-house data that is not publicly accessible.

performance that significantly outperforms the original LLaVA [86].

Computational cost. For LLaVA-1.5, we use the same pretraining dataset, and keep the training iterations and batch size roughly the same for instruction tuning as LLaVA [86]. Due to the increased image input resolution to 336^2, the training of LLaVA-1.5 is ~2× as long as LLaVA: ~6 hours of pretraining and ~20 hours of visual instruction tuning, using 8× A100s.

Scaling to Higher Resolutions

In Sec. 4.3, we observe the advantage that scaling up the input image resolution improves the model's capabilities. However, the image resolution of the existing open source CLIP vision encoders is limited to 336^2, preventing the support of higher resolution images by simply replacing the vision encoder as we did in Sec. 4.3. In this section, we present an early exploration of scaling the LMM to higher resolutions, while maintaining the data efficiency of LLaVA-1.5.

When using ViT [32] as the vision encoder, to scale up the resolution,

Method	POPE [81]			MME	MMBench [90]		SEED-Bench [72]			LLaVA-Wild [86]	MM-Vet
	rand	pop	adv	[36]	en	cn	all	img	vid		[157]
BLIP2-14B [77]	**89.6**	85.5	80.9	1293.8	–	–	46.4	49.7	36.7	38.1	22.4
InstructBLIP-8B [29]	–	–	–	–	36	23.7	53.4	58.8	38.1	60.9	26.2
InstructBLIP-14B [29]	87.7	77	72	1212.8	–	–	–	–	–	58.2	25.6
Shikra-13B [18]	–	–	–	–	58.8	–	–	–	–	–	–
IDEFICS-9B [54]	–	–	–	–	48.2	25.2	–	44.5	–	–	–
IDEFICS-80B [54]	–	–	–	–	54.5	38.1	–	53.2	–	–	–
Qwen-VL [7]	–	–	–	–	38.2	7.4	56.3	62.3	39.1	–	–
Qwen-VL-Chat [7]	–	–	–	1487.5	60.6	56.7	58.2	65.4	37.8	–	–
LLaVA-7B [86]	76.3	72.2	70.1	809.6	38.7	36.4	33.5	37.0	23.8	62.8	25.5
LLaVA-1.5-7B	87.3	86.1	84.2	1510.7	64.3	58.3	58.6	66.1	37.3	65.4	31.1
LLaVA-1.5-13B	87.1	**86.2**	**84.5**	**1531.3**	**67.7**	**63.6**	**61.6**	**68.2**	**42.7**	**72.5**	**36.1**
LLaVA-1.5-13B-HD	87.5	**86.4**	**85.0**	1500.1	**68.8**	61.9	**62.6**	**70.1**	41.3	72.0	**39.4**

Table 4.4: **Comparison with SoTA methods on benchmarks for instruction-following LMMs.** LLaVA-1.5 achieves the best overall performance.

previous approaches mostly choose to perform positional embedding interpolation [7, 77] and adapt the ViT backbone to the new resolution during finetuning. However, this usually requires the model to be finetuned on a large-scale image-text paired dataset [7, 77], and limits the resolution of the image to a fixed size that the LMM can accept during inference.

Instead, as shown in Fig. 4.2, we overcome this by dividing the image into smaller image patches of the resolution that the vision encoder is originally trained for, and encode them independently. After obtaining the feature maps of individual patches, we then combine them into a single large feature map of the target resolution, and feed that into the LLM. To provide the LLM with the global context and to reduce the artifact of the split-encode-merge operation, we additionally concatenate the feature of a downsampled image to the merged feature map. This allows us to scale the input to any arbitrary resolution and maintain the data efficiency of LLaVA-1.5. We call this resulting model LLaVA-1.5-HD.

4.4 Empirical Evaluation

Benchmarks

We evaluate LLaVA-1.5 on a collection of both academic-task-oriented benchmarks and recent benchmarks specifically proposed for instruction-following LMMs, totalling 12 benchmarks. For academic-task-oriented benchmarks, VQA-v2 [43] and GQA [53] evaluate model's visual perception capabilities on open-ended short answers. VizWiz [45] contains 8,000 images to evaluate model's zero-shot generalization on visual questions asked by visually impaired people. Following InstructBLIP [29], the image subset of ScienceQA [92] with multiple choice are used to evaluate the zero-shot generalization on scientific question answering. TextVQA [126] contains text-rich visual question answering.

For recent benchmarks proposed for instruction-following LMMs, POPE [81] evaluates model's degree of hallucination on three sampled subsets of COCO [84]: random, common, and adversarial and we report the F1 score on all three splits. Other benchmarks evaluate the model's capabilities on a wide range of domains and applications, with different response formats. MME-Perception [36] evaluates model's visual perception with yes/no questions. MMBench [90] evaluates model's answer robustness with all-round shuffling on multiple choice answers. MMBench-CN [90] is the Chinese-translated version of MMBench. SEED-Bench [72] evaluates model's performance on both images and videos with multiple choice, and we sample the frame in the middle to evaluate the accuracy on videos. LLaVA-Bench-in-the-Wild [86] and MM-Vet [157] evaluate model's capabilities in engaging in visual conversations on a diverse range of tasks, and evaluates the correctness and the helpfulness of the response with GPT-4 evaluation.

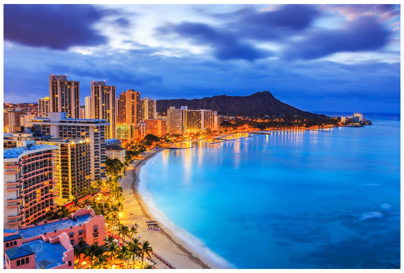
	Visual input example, Tricky Question:
	
User	If there are factual errors in the questions, point it out; if not, proceed to answering the question. What's happening in the desert?
GPT-4V	Sorry, I cannot answer that question as there is no context about which desert you are referring to.
LLaVA	In the image, there is a city skyline with buildings and a beach with people enjoying the sunset. The scene is set in a desert, which is unusual for a beach setting.
LLaVA-1.5	There are no deserts in the image. The image features a beach with palm trees, a city skyline, and a large body of water.

Table 4.5: LLaVA-1.5 can detect and answer tricky questions when prompted to verify the question.

Results

We show that LLaVA-1.5 achieves the best overall performance on 12 benchmarks, despite using magnitudes smaller pretraining and instruction tuning data compared with other methods [29, 7]. LLaVA-1.5 significantly outperforms LLaVA on all benchmarks for instruction-following LMMs. Note that it is challenging to evalute the original LLaVA on academic datasets like VQA-v2 [43] that demand open-ended short answers.

When we continue to scale up the image resolution to 448^2 with LLaVA-1.5-HD, it further improves the overall performance on all benchmarks, especially on tasks that require perception of details in the images (*e.g.*

OCR in MM-Vet, detailed description in LLaVA-Bench-in-the-Wild [86]). Moreover, we find that adding the global context effectively recovers the model from the split-and-merge artifacts and guides the model to more easily locate the relevant regions from the high-resolution features.

It is encouraging that *LLaVA-1.5 achieves the best performance with the simplest architecture, academic compute and public datasets, and yields a fully-reproducible and affordable baseline for future research*. The results also suggest that visual instruction tuning plays an important role in improving an LMM's capabilities, and raises questions upon the common belief that LMMs require significant amount of vision-language alignment pretraining [29, 77, 7], despite that the vision encoders (*e.g*. CLIP [112], OpenCLIP [55], EVA-CLIP [35], *etc*.) are already pretrained on web-scale image-text paired data. LLaVA-1.5 (even the 7B model) outperforms 80B IDEFICS [54], a Flamingo-like LMM with billions of trainable parameters for cross-modal connection. This also makes us rethink the benefits of the vision samplers and the necessity of the additional large-scale pretraining, in terms of multimodal instruction-following capabilities.

Global context. For higher resolution, we pad and resize the image to a single image of 224^2, and concatenate it with the high resolution features to provide a global context. Ablation on a 7B model shows that the global context effectively boosts performance on all three validation benchmarks.

Emerging Properties

Format instruction generalization. Although LLaVA-1.5 is only trained with a limited number of format instructions, it generalizes to others. First, VizWiz [45] requires the model to output "Unanswerable" when the provided content is insufficient to answer the question, and our response format prompt effectively instructs the model to do so (11.1% $\rightarrow$ 67.8% on unanswerable questions). We additionally present qualitative examples

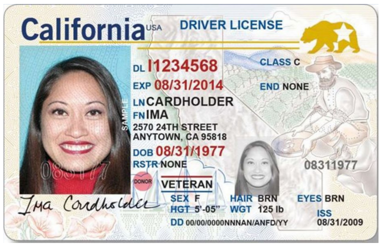
	Visual input example, Constrained JSON Output:
	 California USA DRIVER LICENSE DL I1234568 CLASS C EXP 08/31/2014 END NONE LN CARDHOLDER FN IMA 2570 24TH STREET ANYTOWN, CA 95818 DOB 08/31/1977 RSTR NONE 08311977 DONOR VETERAN SEX F HAIR BRN EYES BRN HGT 5'-05" WGT 125 lb ISS 08/31/2009 DD 00/00/0000NNNAN/ANFD/YY Ima Cardholder
User	Please read the text in this image and return the information in the following JSON format (note xxx is placeholder, if the information is not available in the image, put "N/A" instead). {"class": xxx, "DLN": xxx, "DOB": xxx, "Name": xxx, "Address": xxx, "EXP": xxx, "ISS": xxx, "SEX": xxx, "HGT": xxx, "WGT": xxx, "EYES": xxx, "HAIR": xxx, "DONOR": xxx}
GPT-4V	{"class": "C", "DLN": "1234568", "DOB": "08/31/1977", "Name": "Ima Cardholder", "Address": "2570 24th Street, Anytown, CA 95818", "EXP": "08/31/2014", "ISS": "08/31/2011", "SEX": "F", "HGT": "5-05", "WGT": "125 lbs", "EYES": "BRN", "HAIR": "BRN", "DONOR": "VETERAN"}
LLaVA	{"class": "Driver License", "DLN": "CA123 456 789", "DOB": "01/01/1990", "Name": "Amy", "Address": "Anytown, CA", "EXP": "08/31/2023", "ISS": "Veteran", "SEX": “F", "HGT": "5′5", "WGT": "130", "EYES": "Brown", "HAIR": "Brown", "DONOR": "Unknown"}
LLaVA-1.5	{"class": "C", "DLN": "1132456789", "DOB": "08/23/1971", "Name": "Ima Cardholder", "Address": "Anytown, State, Zip", "EXP": "08/23/2014", "ISS": "California", "SEX": "F", "HGT": "5′5", "WGT": "123", "EYES": "Brown", "HAIR": "Brown", "DONOR": "N/A"}

Table 4.6: LLaVA-1.5 can extract information from the image and answer following the required format, despite a few errors compared with GPT-4V. GPT-4V results are obtained from [147].

	GQA	MME	MM-Vet
high-res patch only	62.9	1425.8	31.9
+global context	63.8 (+0.9)	1497.5 (+71)	35.1 (+3.2)

on instructing LLaVA-1.5 to verify tricky questions (Fig. 4.5), respond in a constrained JSON format (Fig. 4.6).

Multilingual multimodal capability. Though LLaVA-1.5 is *not* finetuned for multilingual multimodal instruction following *at all* (all visual instructions including VQA are in English), we find that it is capable of following multilingual instructions. This is partly due to the multilingual language instructions in ShareGPT [121]. Although ShareGPT does not contain images in its instructions, the model learns from this dataset the behavior of adaptively responding with the language that corresponds to the user's request. We empirically show that this behavior is transferred to visual conversations. We also quantitatively evaluate the model's generalization capability to Chinese on MMBench-CN [90], where the questions of MMBench are converted to Chinese. Notably, LLaVA-1.5 outperforms Qwen-VL-Chat by +7.3% (63.6% vs 56.7%), despite Qwen being finetuned on Chinese multimodal instructions while LLaVA-1.5 is not.

Ablation on LLM Choices

In NLP, findings [132] suggest that the capability of the base LLM can affect its instruction-tuned successors. In this section, we explore two families of LLMs and study their contribution to the final model's multimodal capability: LLaMA-based (Vicuna-v1.1, Vicuna-v1.3) and LLaMA-2-based (Vicuna-v1.5, LLaMA-2-Chat). Vicuna-v1.3 and Vicuna-v1.5 use the same ~150K ShareGPT [121] data (2× that used in v1.1). Unlike Vicuna series that is only trained with supervised instruction finetuning (SFT), LLaMA-2-Chat is further optimized with reinforcement-learning from human-

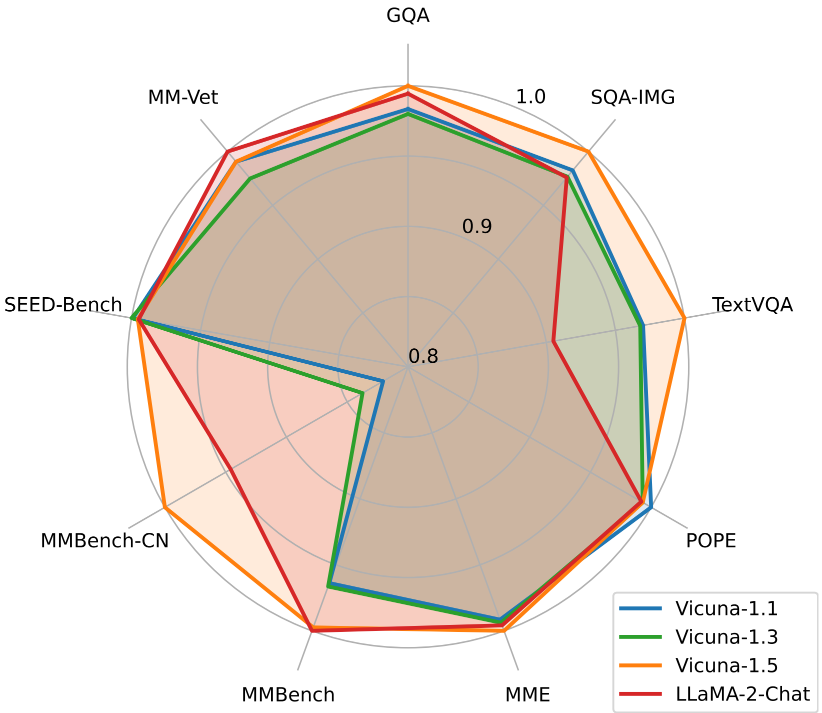


Figure 4.3: **Ablation on LLM choices**. Data points represent the relative performance of the best performing variant for each dataset.

feedback (RLHF). We visualize the relative performance of these variants in Fig. 4.3.

First, we find that Vicuna-v1.5 achieves the best overall performance, and LLaMA-2-based models generally perform better than LLaMA-1-based, suggesting the importance of the base language model. This is further evidenced by the results on MMBench-CN [90]: despite Vicuna-v1.3 and v1.5 using the same ShareGPT data for instruction tuning, the performance in generalization to Chinese of Vicuna-v1.3 is significantly worse than v1.5.

Second, language instruction-tuning matters on specific capabilities that are required by each dataset. For example, although LLaMA-2-Chat and Vicuna-v1.5 achieves almost the same performance on MMBench,

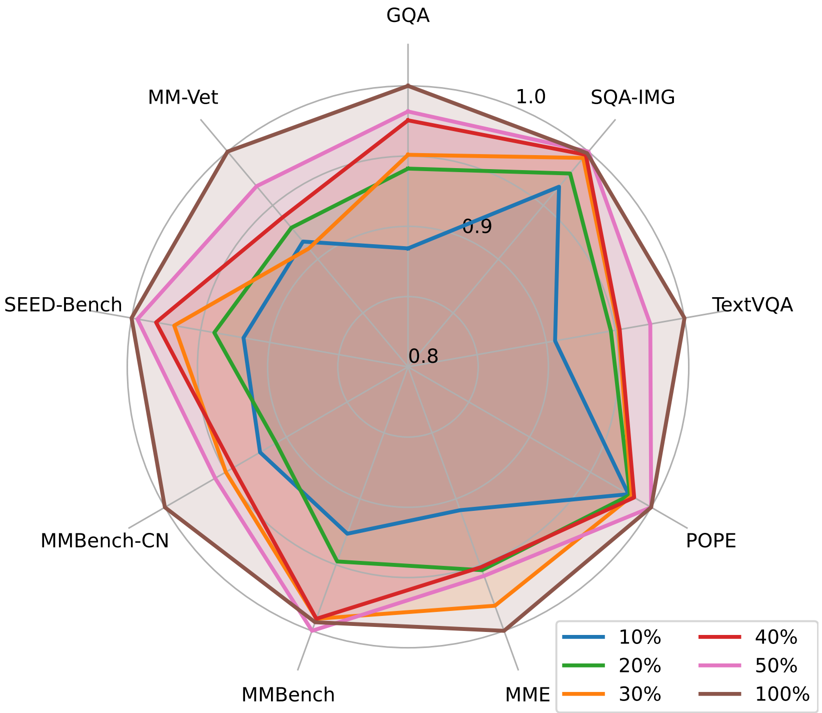


Figure 4.4: **Ablation on data efficiency.** Data points represent the relative performance of the best performing variant for each dataset.

the generalization to MMBench-CN [90] of LLaMA-2-Chat is worse than Vicuna-v1.5, which is partly due to that the most SFT/RLHF data of LLaMA-2-Chat is in English and does not contain as many multilingual data as in ShareGPT. Furthermore, TextVQA requires both the model's capability of identifying the text characters in the images, and also processing the noisy outputs from the OCR engine; such noise *may* be more commonly observed in the ShareGPT data, which is collected in-the-wild from daily usage of ChatGPT.

4.5 Open Problems in LMMs

Given the successful scaling of LLaVA-1.5, we conduct additional studies on open problems in LMMs using the model design and data mixture of LLaVA-1.5.

Data Efficiency

Despite the data efficiency of LLaVA-1.5 when compared with approaches like InstructBLIP [29], the training of LLaVA-1.5 still doubles when compared with LLaVA. In this section, we conduct experiments for further improving the data efficiency by randomly sub-sampling the training data mixture of LLaVA-1.5, with a sampling ratio ranging from 0.1 to 0.5. We visualize the relative performance of different sampling variants in Fig. 4.4.

First, the full data mixture provides the best knowledge coverage, and allows the model to achieve the best overall performance. To our surprise, with only 50% of the samples, the model still maintains more than 98% of the full dataset performance. This suggests that there is room for further improvements in data efficiency.

Second, when downsampling the dataset to 50%, the model's performance on MMBench, ScienceQA, and POPE does not decrease at all, and it even slightly improves on MMBench. Similarly, the model's performance remains steady when further downscaling the data from 50% to 30%. These results show promise of having the less-is-more [172] benefit for multimodal models as well.

Rethinking Hallucination in LMMs

Hallucination is an important issue to tackle for LLMs and LMMs. Often in LMMs, we attribute the model's hallucination to the errors or hallucinations in the training dataset. For example, the detailed descriptions in LLaVA-Instruct [86] may contain a small amount of hallucinated content, and it is believed that training on such data *may* have caused the model to hallucinate when asked to "describe the image in detail". However, we find that such hallucination is significantly reduced, when we scale the model's inputs to higher resolutions like 448^2.

This finding is interesting as it suggests that the LMMs may be robust to *a few* such errors in the training data. However, when the input resolution is not sufficient for the model to discern all details in the training data, and the amount of data that is at that granularity beyond the model's capability becomes large enough, the model *learns* to hallucinate. This further suggests that there needs to be a balance between improving the data annotation with more details and the model's capability to properly process the information at such granularities. We hope this finding provides a reference for future work in terms of dealing with hallucination and the scaling of the models and data.

Compositional Capabilities

We demonstrate interesting compositional capabilities in LLaVA-1.5: the model trained on a set of tasks independently generalizes to tasks that require a combination of these capabilities without explicit joint training. We note some of the findings below.

First, we observe an improved language capability in visual conversations after including the ShareGPT [121] data, including the multimodal multilingual capability as discussed in Sec. 4.4. Moreover, the model is more capable at providing longer and more detailed responses in visual conversations. Second, the additional visual knowledge from the academic-task-oriented datasets, improves the visual groundness of LLaVA-1.5's responses in visual conversations, as evidenced quantitatively by the improved results on MM-Vet [157] and LLaVA-Wild [86] in Table 4.4.

However, there is still difficulty in achieving ideal performance for some tasks that require a certain combination of capabilities. For example, being able to correctly answer the attribute of a certain object in VQA, does not guarantee an accurate depiction of that object attribute in a detailed description of the whole image. Furthermore, the capability of engaing in conversations with certain foreign languages (*e.g*. Korean) still falls behind.

These findings suggest that the compositional capabilities of LMMs can be leveraged to improve the model's performance without significantly increasing the data by exhaustively including all task combinations. Yet, it can be further investigated, and a deeper understanding of the mechanism behind the compositional capabilities of LMMs can further improve the capability and the data efficiency of LLaVA-1.5.

4.6 Conclusion

In this chapter, we take a step towards demystifying the design of large multimodal models, and propose a simple, effective, and data-efficient baseline, LLaVA-1.5, for large multimodal models. In addition, we explore the open problems in visual instruction tuning, scale LMMs to higher resolutions, and present some intriguing findings in terms of model hallucination and compositional capabilities for LMMs. We hope these improved and easily-reproducible baselines as well as the new findings will provide a reference for future research in open-source LMM.

Limitations. Despite the promising results demonstrated by LLaVA-1.5, several limitations must be acknowledged. First, LLaVA-1.5 utilizes full image patches, potentially prolonging each training iteration. While visual resamplers [77, 29, 7] reduce the number of visual patches in LLMs, they currently cannot achieve convergence as efficiently as LLaVA with a comparable amount of training data, probably due to more trainable parameters in the resamplers. The development of a sample-efficient visual resampler could pave the way for future scaling-up of instruction-following multimodal models. Second, LLaVA-1.5 is not yet capable of processing multiple images due to the lack of such instruction-following data, and the limit of the context length. Third, although LLaVA-1.5 exhibits proficiency in following complex instructions, its problem-solving capabilities can still be limited in certain domains, which could be improved with a more capable language model and with high-quality, targeted visual instruction tuning data. Finally, despite its significantly reduced propensity for hallucination, LLaVA-1.5 is not exempt from producing hallucinations and occasionally disseminating misinformation, and should be used with caution in critical applications (*e.g*. medical).

4.7 LLaVA-NeXT: Improved reasoning, OCR, and world knowledge

With a simple and efficient design, LLaVA-1.5 [85] achieves great performance on a benchmark suite of 12 datasets. It has since served as the foundation of many comprehensive studies of data, model, and capabilities of large multimodal models (LMM), and has enabled various new applications. However, there still exists a gap between open models from academia and proprietary models like GPT-4V and Google Gemini Pro. In this section, we conduct additional scaling based on LLaVA-1.5, and we present LLaVA-NeXT, with improved reasoning, OCR, and world knowledge. LLaVA-NeXT even exceeds Gemini Pro on several benchmarks.

Compared with LLaVA-1.5, LLaVA-NeXT has several improvements:

- Increasing the input image resolution to $4\times$ more pixels. This allows it to grasp more visual details. It supports three aspect ratios, up to 672×672, 336×1344, 1344×336 resolution.

- Better visual reasoning and OCR capability with an improved visual instruction tuning data mixture.
- Better visual conversation for more scenarios, covering different applications. Better world knowledge and logical reasoning.
- Efficient deployment and inference with SGLang.

Along with performance improvements, LLaVA-NeXT maintains the minimalist design and data efficiency of LLaVA-1.5. It re-uses the pretrained connector of LLaVA-1.5, and still uses less than 1M visual instruction tuning samples. The largest 34B variant finishes training in ~1 day with 32 A100s.

Dynamic High Resolution. We design our model at high resolution with an aim to preserve its data efficiency. When provided with high-resolution images and representations that preserve these details, the model's capacity to perceive intricate details in an image is significantly improved. It reduces the model hallucination that conjectures the imagined visual content when confronted with low-resolution images. Our 'AnyRes' technique is designed to accommodate images of various high resolutions. We employ a grid configuration of $\{2 \times 2, 1 \times \{2, 3, 4\}, \{2, 3, 4\} \times 1\}$, balancing performance efficiency with operational costs.

High-quality User Instruct Data. Our definition of high-quality visual instruction-following data hinges on two principal criteria: First, the diversity of task instructions, ensuring adequately represent a broad spectrum of user intents that are likely to be encountered in real-world scenarios, particularly during the model's deployment phase. Second, the superiority of responses is critical, with the objective of soliciting favorable user feedback. To achieve this, we consider two data sources: (1) Existing GPT-V data. LAION-GPT-V and ShareGPT-4V. (2) To further facilitate better visual conversation for more scenarios, we collect a small 15K visual instruction tuning dataset covering different applications. The instructions and images come from LLaVA demo, which are real-world users requests. We carefully filter samples that may have privacy concerns or are potentially harmful, and generate the response with GPT-4V.

Multimodal Document/Chart Data. (1) We remove TextCaps from our training data as we realize that TextCaps uses the same set of training images as TextVQA. This allows us to better understand the zero-shot OCR capability of our model when evaluating TextVQA during development. To maintain and further improve our model's OCR capability, we replace TextCaps with DocVQA and SynDog-EN. (2) Motivated by Qwen-VL-7B-Chat, we further add ChartQA, DVQA, and AI2D for better chart and diagram understanding.

Scaling LLM backbone. In addition to Vicuna-1.5 (7B and 13B), we consider more LLMs, including Mistral-7B and Nous-Hermes-2-Yi-34B. These LLMs possess nice properties, flexible commercial use terms, strong bilingual support and larger language model capacity. It allows LLaVA to support a wider spectrum of users and more scenarios in the community. The LLaVA recipe works well with various LLMs, and scales up smoothly with the LLM up to 34B.

Results

Data		Model	MMMU val	Math Vista	MMB ENG	MMB CN	MMVet	LLaVA Wild	SEED IMG
PT	IT								
N/A	N/A	GPT-4V	56.8	49.9	75.8	73.9	67.6	-	71.6
N/A	N/A	Gemini Ultra	59.4	53	-	-	-	-	-
N/A	N/A	Gemini Pro	47.9	45.2	73.6	74.3	64.3	-	70.7
1.4B	50M	Qwen-VL-Plus	45.2	43.3	-	-	55.7	-	65.7
1.5B	5.12M	CogVLM-30B	32.1	-	-	-	56.8	-	-
125M	~1M	Yi-VL-34B	45.9	-	-	-	-	-	-
558K	665K	LLaVA-1.5-13B	36.4	27.6	67.8	63.3	36.3	72.5	68.2
558K	760K	LLaVA-NeXT-13B	36.2	35.3	70	64.4	48.4	87.3	71.9
558K	760K	LLaVA-NeXT-34B	51.1	46.5	79.3	79	57.4	89.6	75.9

5 CONCLUSION AND DISCUSSION

In this thesis, we have explored the advancement of computer vision models, focusing on their steerability and customizability with vision language models. By introducing innovative methods and frameworks, this work has significantly pushed the boundaries of how these models understand and convey their understandings of the visual world, making them accessible and flexible for a wide range of tasks.

The contributions of this thesis to the field of vision-language models are manifold. We began with the development of REACT, a method that efficiently leverages the web-scale data with retrieval to customize vision models with minimal human labeling effort, effectively enables the adaptability of vision foundation models like CLIP to specialized tasks.

While being customizable, REACT models is not fully steerable without the access to the model weights. To further improve the steerability of vision-language models and advance the capability of vision-language models to follow natural language instructions, we introduced the LLaVA-series. These models represent pioneering efforts in building large multimodal systems that can be steered directly through user instructions. LLaVA utilized machine-generated multimodal data for instruction tuning, proving that large multimodal models can achieve impressive levels of understanding and response accuracy. LLaVA-1.5 and LLaVA-NeXT further refined these capabilities, optimizing design choices and demonstrating state-of-the-art performance with remarkable efficiency and scalability.

Each chapter of this thesis not only presented distinct approaches aimed at enhancing the interactivity and functionality of vision-language systems but also set new standards for intuitive and practical applications in the field. From educational technologies to assistive devices for the visually impaired, the models developed in this research offer significant improvements in user interaction, opening up new possibilities for how visual content is generated and manipulated in response to natural language inputs.

Future Directions

Looking ahead, the work presented in this thesis opens up several avenues for future research. One promising area is the further development of models that can seamlessly integrate even more diverse multimodal inputs, such as audio and tactile feedback, to create even more robust and immersive interaction systems. Additionally, enhancing the models' ability to understand and generate not just static images but dynamic visual content such as video in response to complex instructions poses a significant challenge that future research could address.

Another critical area involves improving the models' understanding of nuanced human language and its context, which could lead to more sophisticated and subtle interactions between AI systems and their users. This could involve deepening the models' grasp of cultural, emotional, and situational contexts, significantly enhancing the relevance and personalization of their responses.

Finally, as these models become more capable and widely used, it will be essential to consider and address ethical concerns related to privacy, bias, and the use of AI in sensitive applications. Ensuring that these powerful tools are used responsibly and beneficially should be a priority for the community moving forward.

In conclusion, this thesis has laid a robust foundation for the next generation of vision-language models, significantly advancing their capabilities and applications. The path forward is rich with opportunities for further innovation and impactful research, promising to revolutionize the ways we interact with and leverage AI in our daily lives and professional environments.

REFERENCES

[1] 2022. Langchain. `https://github.com/hwchase17/langchain`.

[2] Adept AI. 2024. Fuyu-8b: A multimodal architecture for ai agents. `https://www.adept.ai/blog/fuyu-8b`.

[3] Alayrac, Jean-Baptiste, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*.

[4] Anderson, Peter, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the ieee conference on computer vision and pattern recognition*.

[5] Askell, Amanda, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, et al. 2021. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*.

[6] Awadalla, Anas, 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.

[7] Bai, Jinze, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

[8] Bitton, Yonatan, Hritik Bansal, Jack Hessel, Rulin Shao, Wanrong Zhu, Anas Awadalla, Josh Gardner, Rohan Taori, and Ludwig Schimdt. 2023. Visit-bench: A benchmark for vision-language instruction following inspired by real-world use. `2308.06595`.

[9] Black, Kevin, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. 2023. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*.

[10] Blattmann, Andreas, Robin Rombach, Kaan Oktay, and Björn Ommer. 2022. Retrieval-augmented diffusion models. *arXiv preprint arXiv:2204.11824*.

[11] Borgeaud, Sebastian, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2021. Improving language models by retrieving from trillions of tokens. *arXiv preprint arXiv:2112.04426*.

[12] Brooks, Tim, Aleksander Holynski, and Alexei A Efros. 2022. Instruct pix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*.

[13] Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *NeurIPS*.

[14] Carlini, Nicholas, Milad Nasr, Christopher A Choquette-Choo, Matthew Jagielski, Irena Gao, Anas Awadalla, Pang Wei Koh, Daphne Ippolito, Katherine Lee, Florian Tramer, et al. 2023. Are aligned neural networks adversarially aligned? *arXiv preprint arXiv:2306.15447*.

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

[16] Chen, Delong, Jianfeng Liu, Wenliang Dai, and Baoyuan Wang. 2023. Visual instruction tuning with polite flamingo. *arXiv preprint arXiv:2307.01003*.

[17] Chen, Jiacheng, Hexiang Hu, Hao Wu, Yuning Jiang, and Changhu Wang. 2021. Learning the best pooling strategy for visual semantic embedding. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 15789–15798.

[18] Chen, Keqin, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. 2023. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*.

[19] Chen, Ting, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Icml*.

[20] Chen, Wenhu, Hexiang Hu, Xi Chen, Pat Verga, and William W Cohen. 2022. Murag: Multimodal retrieval-augmented generator for open question answering over images and text. *arXiv preprint arXiv:2210.02928*.

[21] Chen, Wenhu, Hexiang Hu, Chitwan Saharia, and William W Cohen. 2022. Re-imagen: Retrieval-augmented text-to-image generator. *arXiv preprint arXiv:2209.14491*.

[22] Chen, Xi, Josip Djolonga, Piotr Padlewski, Basil Mustafa, Soravit Changpinyo, Jialin Wu, Carlos Riquelme Ruiz, Sebastian Goodman, Xiao Wang, Yi Tay, et al. 2023. Pali-x: On scaling up a multilingual vision and language model. *arXiv preprint arXiv:2305.18565*.

[23] Chen, Xi, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. 2022. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*.

[24] Chen, Xinlei, Haoqi Fan, Ross Girshick, and Kaiming He. 2020. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*.

[25] Chiang, Wei-Lin, 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.

[26] Chowdhery, Aakanksha, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

[27] Chung, Hyung Won, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

[28] CVinW. 2022. Computer vision in the wild. `https://github.com/Computer-Vision-in-the-Wild/CVinW_Readings`.

[29] Dai, Wenliang, 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. *arXiv preprint arXiv:2305.06500*.

[30] Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Cvpr*.

[31] Dong, Xiaoyi, Yinglin Zheng, Jianmin Bao, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, et al. 2022. Maskclip: Masked self-distillation advances contrastive language-image pretraining. *arXiv preprint arXiv:2208.12262*.

[32] Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

[33] Driess, Danny, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. 2023. PaLM-E: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*.

[34] Faghri, Fartash, Hadi Pouransari, Sachin Mehta, Mehrdad Farajtabar, Ali Farhadi, Mohammad Rastegari, and Oncel Tuzel. 2023. Reinforce data, multiply impact: Improved model accuracy and robustness with dataset reinforcement. *arXiv preprint arXiv:2303.08983*.

[35] Fang, Yuxin, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. 2023. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 19358–19369.

[36] Fu, Chaoyou, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, et al. 2023. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*.

[37] Gafni, Oran, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. 2022. Make-a-scene: Scene-based text-to-image generation with human priors. *ArXiv* abs/2203.13131.

[38] Gan, Zhe, Linjie Li, Chunyuan Li, Lijuan Wang, Zicheng Liu, Jianfeng Gao, et al. 2022. Vision-language pre-training: Basics, recent advances, and future trends. *Foundations and Trends® in Computer Graphics and Vision*.

[39] Gao, Yuting, Jinfeng Liu, Zihan Xu, Jun Zhang, Ke Li, and Chunhua Shen. 2022. Pyramidclip: Hierarchical feature alignment for vision-language model pretraining. *arXiv preprint arXiv:2204.14095*.

[40] Geng, Xinyang, Hao Liu, Lisa Lee, Dale Schuurams, Sergey Levine, and Pieter Abbeel. 2022. Multimodal masked autoencoders learn transferable representations. *arXiv preprint arXiv:2205.14204*.

[41] Gilardi, Fabrizio, Meysam Alizadeh, and Maël Kubli. 2023. Chatgpt outperforms crowd-workers for text-annotation tasks. *arXiv preprint arXiv:2303.15056*.

[42] Gong, Tao, Chengqi Lyu, Shilong Zhang, Yudong Wang, Miao Zheng, Qian Zhao, Kuikun Liu, Wenwei Zhang, Ping Luo, and Kai Chen. 2023. Multimodal-gpt: A vision and language model for dialogue with humans. *arXiv preprint arXiv:2305.04790*.

[43] Goyal, Yash, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 6904–6913.

[44] Gupta, Tanmay, and Aniruddha Kembhavi. 2022. Visual programming: Compositional visual reasoning without training. *arXiv preprint arXiv:2211.11559*.

[45] Gurari, Danna, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 3608–3617.

[46] Guu, Kelvin, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *arXiv preprint arXiv:2002.08909*.

[47] Hao, Weituo, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. 2020. Towards learning a generic agent for vision-and-language navigation via pre-training. In *Cvpr*.

[48] He, Kaiming, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 16000–16009.

[49] He, Kaiming, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Cvpr*.

[50] He, Kaiming, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Iccv*.

[51] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Cvpr*.

[52] Huang, Shaohan, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. 2023. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*.

[53] Hudson, Drew A, and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Cvpr*.

[54] IDEFICS. 2023. Introducing idefics: An open reproduction of state-of-the-art visual language model. `https://huggingface.co/blog/idefics`.

[55] Ilharco, Gabriel, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. 2021. Openclip. *Zenodo*.

[56] Iyer, Srinivasan, Xi Victoria Lin, Ramakanth Pasunuru, Todor Mihaylov, Dániel Simig, Ping Yu, Kurt Shuster, Tianlu Wang, Qing Liu, Punit Singh Koura, et al. 2022. Opt-iml: Scaling language model instruction meta learning through the lens of generalization. *arXiv preprint arXiv:2212.12017*.

[57] Jain, Aashi, Mandy Guo, Krishna Srinivasan, Ting Chen, Sneha Kudugunta, Chao Jia, Yinfei Yang, and Jason Baldridge. 2021. Mural: multimodal, multitask retrieval across languages. *arXiv preprint arXiv:2109.05125*.

[58] Jia, Chao, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. *arXiv preprint arXiv:2102.05918*.

[59] Jia, Menglin, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. 2022. Visual prompt tuning. In *Eccv*.

[60] Johnson, Jeff, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data* 7(3): 535–547.

[61] Kazemzadeh, Sahar, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)*, 787–798.

[62] Khandelwal, Urvashi, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2019. Generalization through memorization: Nearest neighbor language models. *arXiv preprint arXiv:1911.00172*.

[63] Koh, Jing Yu, Ruslan Salakhutdinov, and Daniel Fried. 2023. Grounding language models to images for multimodal generation. *arXiv preprint arXiv:2301.13823*.

[64] Kolesnikov, Alexander, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. 2020. Big transfer (bit): General visual representation learning. In *Eccv*.

[65] Krause, Jonathan, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3d object representations for fine-grained categorization. In *4th international ieee workshop on 3d representation and recognition (3drr-13)*. Sydney, Australia.

[66] Krishna, Ranjay, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision* 123:32–73.

[67] Lai, Xin, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. 2023. Lisa: Reasoning segmentation via large language model. *arXiv preprint arXiv:2308.00692*.

[68] Lee, Janghyeon, Jongsuk Kim, Hyounguk Shon, Bumsoo Kim, Seung Hwan Kim, Honglak Lee, and Junmo Kim. 2022. Uniclip: Unified framework for contrastive language-image pre-training. *arXiv preprint arXiv:2209.13430*.

[69] Lewis, Patrick, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *NeurIPS*.

[70] Li, Bo, Peiyuan Zhang, Jingkang Yang, Yuanhan Zhang, Fanyi Pu, and Ziwei Liu. 2023. Otterhd: A high-resolution multi-modality model. `2311.04219`.

[71] Li, Bo, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Fanyi Pu, Jingkang Yang, Chunyuan Li, and Ziwei Liu. 2023. Mimic-it: Multimodal in-context instruction tuning. *arXiv preprint arXiv:2306.05425*.

[72] Li, Bohao, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. 2023. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*.

[73] Li, Boyi, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. 2022. Language-driven semantic segmentation. *ICLR*.

[74] Li, Chunyuan, Zhe Gan, Zhengyuan Yang, Jianwei Yang, Linjie Li, Lijuan Wang, and Jianfeng Gao. 2023. Multimodal foundation models: From specialists to general-purpose assistants. *arXiv preprint arXiv:2309.10020*.

[75] Li, Chunyuan, Haotian Liu, Liunian Harold Li, Pengchuan Zhang, Jyoti Aneja, Jianwei Yang, Ping Jin, Houdong Hu, Zicheng Liu, Yong Jae Lee, and Jianfeng Gao. 2022. ELEVATER: A benchmark and toolkit for evaluating language-augmented visual models. In *Neurips track on datasets and benchmarks*.

[76] Li, Chunyuan, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2023. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *arXiv preprint arXiv:2306.00890*.

[77] Li, Junnan, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

[78] Li, Liunian Harold, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. 2022. Grounded language-image pre-training. In *Cvpr*.

[79] Li, Xiang Lisa, and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

[80] Li, Yangguang, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. 2021. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*.

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

[82] Li, Yuheng, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. 2023. Gligen: Open-set grounded text-to-image generation. *CVPR*.

[83] Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, et al. 2014. Microsoft coco: Common objects in context. In *Eccv*.

[84] Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, et al. 2014. Microsoft COCO: Common objects in context. In *Eccv*.

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

[86] Liu, Haotian, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Neurips*.

[87] Liu, Haotian, Kilho Son, Jianwei Yang, Ce Liu, Jianfeng Gao, Yong Jae Lee, and Chunyuan Li. 2023. Learning customized visual models with retrieval-augmented knowledge.

[88] Liu, Shilong, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. 2023. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*.

[89] Liu, Weijie, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-BERT: Enabling language representation with knowledge graph. In *Aaai*.

[90] Liu, Yuan, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. 2023. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*.

[91] Long, Alexander, Wei Yin, Thalaiyasingam Ajanthan, Vu Nguyen, Pulak Purkait, Ravi Garg, Alan Blair, Chunhua Shen, and Anton van den Hengel. 2022. Retrieval augmented classification for long-tail visual recognition. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 6959–6969.

[92] Lu, Pan, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*.

[93] Lu, Yadong, Chunyuan Li, Haotian Liu, Jianwei Yang, Jianfeng Gao, and Yelong Shen. 2023. An empirical study of scaling instruct-tuned large multimodal models. *arXiv preprint arXiv:2309.09958*.

[94] Maji, S., J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. 2013. Fine-grained visual classification of aircraft. Tech. Rep., arXiv.

[95] Malkov, Yu A, and Dmitry A Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence* 42(4):824–836.

[96] Mao, Junhua, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. 2016. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 11–20.

[97] Marino, Kenneth, Xinlei Chen, Devi Parikh, Abhinav Gupta, and Marcus Rohrbach. 2021. Krisp: Integrating implicit and symbolic knowledge for open-domain knowledge-based VQA. In *Cvpr*.

[98] Marino, Kenneth, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Conference on computer vision and pattern recognition (cvpr)*.

[99] Mishra, Anand, 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)*, 947–952. IEEE.

[100] Mishra, Swaroop, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2021. Cross-task generalization via natural language crowdsourcing instructions. *arXiv preprint arXiv:2104.08773*.

[101] Mu, Norman, Alexander Kirillov, David Wagner, and Saining Xie. 2021. Slip: Self-supervision meets language-image pre-training. *arXiv preprint arXiv:2112.12750*.

[102] Mustafa, Basil, Carlos Riquelme, Joan Puigcerver, Rodolphe Jenatton, and Neil Houlsby. 2022. Multimodal contrastive learning with limoe: the language-image mixture of experts. *arXiv preprint arXiv:2206.02770*.

[103] OpenAI. 2023. ChatGPT. `https://openai.com/blog/chatgpt/`.

[104] ———. 2023. Gpt-4 technical report. `2303.08774`.

[105] ———. 2023. Gpt-4v(ision) system card. `https://cdn.openai.com/papers/GPTV_System_Card.pdf`.

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

[107] Peng, Baolin, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with GPT-4. *arXiv preprint arXiv:2304.03277*.

[108] Peters, Matthew E, Mark Neumann, Robert L Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A Smith. 2019. Knowledge enhanced contextual word representations. *arXiv preprint arXiv:1909.04164*.

[109] Pham, Hieu, Zihang Dai, Golnaz Ghiasi, Kenji Kawaguchi, Hanxiao Liu, Adams Wei Yu, Jiahui Yu, Yi-Ting Chen, Minh-Thang Luong, Yonghui Wu, et al. 2021. Combined scaling for open-vocabulary image classification. *arXiv preprint arXiv: 2111.10050*.

[110] Plummer, Bryan A, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Iccv*.

[111] Qi, Di, Lin Su, Jia Song, Edward Cui, Taroon Bharti, and Arun Sacheti. 2020. Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data. *arXiv preprint arXiv:2001.07966*.

[112] Radford, Alec, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*.

[113] Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*.

[114] Ramesh, Aditya, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *ArXiv* abs/2204.06125.

[115] Rombach, Robin, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. *CVPR* 10674–10685.

[116] Saharia, Chitwan, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, Seyedeh Sara Mahdavi, Raphael Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *ArXiv* abs/2205.11487.

[117] Saito, Kuniaki, Kihyuk Sohn, Xiang Zhang, Chun-Liang Li, Chen-Yu Lee, Kate Saenko, and Tomas Pfister. 2022. Prefix conditioning unifies language and label supervision. *arXiv preprint arXiv:2206.01125*.

[118] Schuhmann, Christoph, 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*.

[119] Schuhmann, Christoph, 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. *arXiv preprint arXiv:2111.02114*.

[120] Schwenk, Dustin, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. 2022. A-okvqa: A benchmark for

visual question answering using world knowledge. In *European conference on computer vision*, 146–162. Springer.

[121] ShareGPT. 2023. `https://sharegpt.com/`.

[122] Sharma, Piyush, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Acl*.

[123] Shen, Sheng, Chunyuan Li, Xiaowei Hu, Yujia Xie, Jianwei Yang, Pengchuan Zhang, Anna Rohrbach, Zhe Gan, Lijuan Wang, Lu Yuan, Ce Liu, Kurt Keutzer, Trevor Darrell, and Jianfeng Gao. 2022. K-LITE: Learning transferable visual models with external knowledge. In *Neurips*.

[124] Sheynin, Shelly, Oron Ashual, Adam Polyak, Uriel Singer, Oran Gafni, Eliya Nachmani, and Yaniv Taigman. 2022. Knn-diffusion: Image generation via large-scale retrieval. *arXiv preprint arXiv:2204.02849*.

[125] Sidorov, Oleksii, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. 2020. Textcaps: a dataset for image captioning with reading comprehension. In *Computer vision–eccv 2020: 16th european conference, glasgow, uk, august 23–28, 2020, proceedings, part ii 16*, 742–758. Springer.

[126] Singh, Amanpreet, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 8317–8326.

[127] Sun, Chen, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. 2017. Revisiting unreasonable effectiveness of data in deep learning era. In *Iccv*.

[128] Surís, Dídac, Sachit Menon, and Carl Vondrick. 2023. Vipergpt: Visual inference via python execution for reasoning. *arXiv preprint arXiv:2303.08128*.

[129] Szot, Andrew, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. 2021. Habitat 2.0: Training home assistants to rearrange their habitat. In *Advances in neural information processing systems (neurips)*.

[130] Taori, Rohan, 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`.

[131] Thomee, Bart, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. 2016. Yfcc100m: The new data in multimedia research. *Communications of the ACM*.

[132] Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

[133] Tsimpoukelli, Maria, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*.

[134] Turing Bletchley. Turing Bletchley. https://www.microsoft.com/en-us/research/blog/turing-bletchley-a-universal-image-language-representation-model-by-microsoft/.

[135] Veeling, Bastiaan S, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. 2018. Rotation equivariant cnns for digital pathology. In *International conference on medical image computing and computer-assisted intervention*, 210–218. Springer.

[136] Wang, Jianfeng, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. 2022. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*.

[137] Wang, Wenhai, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. 2023. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *arXiv preprint arXiv:2305.11175*.

[138] Wang, Yizhong, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*.

[139] Wang, Yizhong, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Benchmarking generalization via in-context instructions on 1,600+ language tasks. *arXiv preprint arXiv:2204.07705*.

[140] Wei, Jason, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

[141] Wu, Chenfei, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. 2023. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*.

[142] Wu, Jialin, Jiasen Lu, Ashish Sabharwal, and Roozbeh Mottaghi. 2021. Multi-modal answer validation for knowledge-based VQA. *arXiv preprint arXiv:2103.12248*.

[143] Xian, Yongqin, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. 2018. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *PAMI*.

[144] Yang, Jianwei, Chunyuan Li, Pengchuan Zhang, Bin Xiao, Lu Yuan, Ce Liu, and Jianfeng Gao. 2022. Unified contrastive learning in image-text-label space. *CVPR*.

[145] Yang, Jinyu, Jiali Duan, Son Tran, Yi Xu, Sampath Chanda, Liqun Chen, Belinda Zeng, Trishul Chilimbi, and Junzhou Huang. 2022. Vision-language pre-training with triple contrastive learning. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 15671–15680.

[146] Yang, Zhengyuan, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. 2021. An empirical study of GPT-3 for few-shot knowledge-based VQA. *arXiv preprint arXiv:2109.05014*.

[147] Yang, Zhengyuan, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. 2023. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv preprint arXiv:2309.17421*.

[148] Yang, Zhengyuan, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and

Lijuan Wang. 2023. Mm-react: Prompting chatgpt for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*.

[149] Yao, Lewei, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. 2021. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*.

[150] Yasunaga, Michihiro, Armen Aghajanyan, Weijia Shi, Rich James, Jure Leskovec, Percy Liang, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2022. Retrieval-augmented multimodal language modeling. *arXiv preprint arXiv:2211.12561*.

[151] Ye, Jiabo, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian, Qi Qian, Ji Zhang, Qin Jin, Liang He, Xin Alex Lin, and Fei Huang. 2023. Ureader: Universal ocr-free visually-situated language understanding with multimodal large language model. `2310.05126`.

[152] Ye, Qinghao, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. 2023. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

[153] You, Haoxuan, Luowei Zhou, Bin Xiao, Noel Codella, Yu Cheng, Ruochen Xu, Shih-Fu Chang, and Lu Yuan. 2022. Learning visual representation from modality-shared contrastive language-image pre-training. In *European conference on computer vision*, 69–87. Springer.

[154] Young, Peter, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics* 2:67–78.

[155] Yu, Jiahui, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. 2022. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*.

[156] Yu, Jiahui, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Benton C. Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. 2022. Scaling autoregressive models for content-rich text-to-image generation. *ArXiv* abs/2206.10789.

[157] Yu, Weihao, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. 2023. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*.

[158] Yu, Wenhao, Chenguang Zhu, Yuwei Fang, Donghan Yu, Shuohang Wang, Yichong Xu, Michael Zeng, and Meng Jiang. 2021. Dict-bert: Enhancing language model pre-training with dictionary. *arXiv preprint arXiv:2110.06490*.

[159] Yuan, Lu, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. 2021. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*.

[160] Zhai, Xiaohua, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. 2022. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 18123–18133.

[161] Zhang, Hao, Feng Li, Xueyan Zou, Shilong Liu, Chunyuan Li, Jianfeng Gao, Jianwei Yang, and Lei Zhang. 2023. A simple framework

for open-vocabulary segmentation and detection. *arXiv preprint arXiv:2303.08131*.

[162] Zhang, Renrui, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. 2023. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*.

[163] Zhang, Shilong, Peize Sun, Shoufa Chen, Min Xiao, Wenqi Shao, Wenwei Zhang, Kai Chen, and Ping Luo. 2023. Gpt4roi: Instruction tuning large language model on region-of-interest. *arXiv preprint arXiv:2307.03601*.

[164] Zhang, Susan, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. OPT: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

[165] Zhang, Yanzhe, Ruiyi Zhang, Jiuxiang Gu, Yufan Zhou, Nedim Lipka, Diyi Yang, and Tong Sun. 2023. Llavar: Enhanced visual instruction tuning for text-rich image understanding. *arXiv preprint arXiv:2306.17107*.

[166] Zhang, Zhuosheng, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. 2023. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*.

[167] Zhao, Bo, Boya Wu, and Tiejun Huang. 2023. Svit: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087*.

[168] Zhao, Yunqing, Tianyu Pang, Chao Du, Xiao Yang, Chongxuan Li, Ngai-Man Cheung, and Min Lin. 2023. On evaluating adversarial robustness of large vision-language models. *arXiv preprint arXiv:2305.16934*.

[169] Zhong, Yiwu, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. 2022. Regionclip: Region-based language-image pretraining. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 16793–16803.

[170] Zhong, Yiwu, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, et al. 2022. Regionclip: Region-based language-image pretraining. *CVPR*.

[171] Zhou, Chong, Chen Change Loy, and Bo Dai. 2022. Extract free dense labels from clip. In *Eccv*.

[172] Zhou, Chunting, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2023. Lima: Less is more for alignment. *arXiv preprint arXiv:2305.11206*.

[173] Zhou, Kaiyang, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Learning to prompt for vision-language models. *International Journal of Computer Vision* 130(9):2337–2348.

[174] Zhou, Yufan, Chunyuan Li, Changyou Chen, Jianfeng Gao, and Jinhui Xu. 2022. Lafite2: Few-shot text-to-image generation. *arXiv preprint arXiv:2210.14124*.

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

[176] Zou, Xueyan, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, et al. 2022.

Generalized decoding for pixel, image, and language. *arXiv preprint arXiv:2212.11270*.