# TG-LLaVA: Text Guided LLaVA via Learnable Latent Embeddings

**Dawei Yan[1,2], Pengcheng Li[2], Yang Li[2], Hao Chen[3], Qingguo Chen[2], Weihua Luo[2], Wei Dong[4], Qingsen Yan[5], Haokui Zhang[1*], Chunhua Shen[3]**

[1]School of Cybersecurity, Northwestern Polytechnical University
[2]AI Business, Alibaba Group
[3] College of Computer Science and Technology, Zhejiang University
[4] College of Information and Control Engineering, Xi'an University of Architecture and Technology
[5]School of Computer Science, Northwestern Polytechnical University

## Abstract

Currently, inspired by the success of vision-language models (VLMs), an increasing number of researchers are focusing on improving VLMs and have achieved promising results. However, most existing methods concentrate on optimizing the connector and enhancing the language model component, while neglecting improvements to the vision encoder itself. In contrast, we propose Text Guided LLaVA (TG-LLaVA) in this paper, which optimizes VLMs by guiding the vision encoder with text, offering a new and orthogonal optimization direction. Specifically, inspired by the purpose-driven logic inherent in human behavior, we use learnable latent embeddings as a bridge to analyze textual instruction and add the analysis results to the vision encoder as guidance, refining it. Subsequently, another set of latent embeddings extracts additional detailed text-guided information from high-resolution local patches as auxiliary information. Finally, with the guidance of text, the vision encoder can extract text-related features, similar to how humans focus on the most relevant parts of an image when considering a question. This results in generating better answers. Experiments on various datasets validate the effectiveness of the proposed method. Remarkably, without the need for additional training data, our proposed method can bring more benefits to the baseline (LLaVA-1.5) compared with other concurrent methods. Furthermore, the proposed method consistently brings improvement in different settings.

**Code** — https://github.com/AIDC-AI

## Introduction

By incorporating visual information into large language models (LLMs), visual language models (VLMs) build on the success of LLMs like ChatGPT (OpenAI 2023a) and Llama (Touvron et al. 2023), taking their capabilities a step further. VLMs are not limited to language-based dialogue with humans, they can also discuss the image content, answer questions related to the visual inputs, etc. Recently, centered around VLMs, researchers have conducted extensive works (Wu et al. 2023; Zhang et al. 2024; Awadalla et al. 2023; Reid et al. 2024).

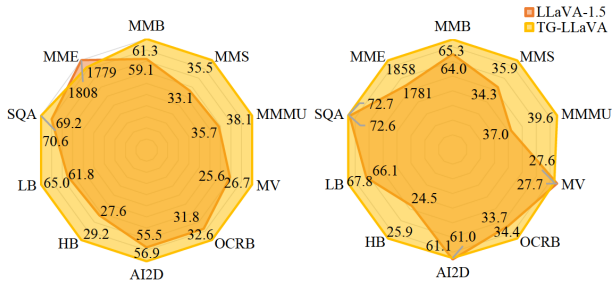


Figure 1: Percentage performance improvements of TG-LLaVA over the baseline LLaVA-1.5 (Liu et al. 2024a) across ten benchmarks, using Vicuna-7B (left) and 13B (right), respectively.

Current adopted VLMs typically consist of three main components: vision encoder, large language model, and connector. The vision encoder, trained on vast amounts of image-text pairs using contrastive learning, encodes images into a shared space with text. Widely used examples include CLIP (Radford et al. 2021) and SigLIP (Zhai et al. 2023). LLMs such as Llama (Touvron et al. 2023), Vicuna (Chiang et al. 2023), Qwen (Bai et al. 2023a), and Yi (Young et al. 2024) have made significant strides in natural language processing tasks, paving the way for integrating vision with text in VLMs. Connector focuses on aligning visual and language features, serving as bridges between modalities.

Corresponding to the main architecture of VLMs, current improvement methods primarily focus on optimizing the connector and enhancing the language model component among the three major components. For instance, BLIP2 (Li et al. 2023b) carefully designs multiple loss functions for both contrastive and generative learning, which allows it to achieve precise cross-modal alignment through a multi-stage training process. MoE-LLaVA (Lin et al. 2024) incorporates a mixture of experts into the second feature forward network layer to enhance the connector component. DenseConnector (Yao et al. 2024) uses dense connections to merge features from various levels, providing more visual information to the LLM. ImageBind-LLM (Han et al. 2023) transforms image features using a binding network and then integrates these transformed features with the word tokens of the LLM. Besides improving the model structure, increasing

*Corresponding author
Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

the amount of data is a commonly employed strategy. This method usually yields more noticeable results, but it also involves a significantly greater workload.

In this paper, we propose Text Guided LLaVA (TG-LLaVA), which optimizes the Visual Language Model from a different even contrasting perspective. Unlike previous work that focuses on enhancing the connector or LLM components, our approach concentrates on improving the visual encoder itself. In contrast to the main strategy that integrating image features into the LLM, we integrate text-guided information into the image features.

The basic idea of our TG-LLaVA is motivated by two key insights: 1) When humans solve visual question answering tasks, they use the question as a prior, selectively focusing on local regions or specific targets to observe and respond. 2) Numerous studies have demonstrated that improved visual representations are crucial for enhancing VLM performance. The proposed TG-LLaVA aims at guiding the visual encoding process of current VLMs using textual instructions, thereby optimizing the visual branch of VLMs. Specifically, the proposed TG-LLaVA contains two text guided modules, text-guided feature optimization mask (TG-FOM) module and text-guided detail perceiver (TG-DP) module. In TG-FOM module, a set of learnable latent embeddings is used to analyze the input text from the global view, then the analyzed language information is added to image feature via a zero-initialized linear layer as guidance. In TG-DP module, a very small number of learnable latent embeddings are used to parse the input text in detail, then the parsed tokens are used as guidance to fuse information from focused image perspective. As shown in Figure 1, extensive experiments have demonstrated the effectiveness of the proposed design, showing significant improvements over the baseline across multiple datasets and different framework without the need for any additional data augmentation or complex enhancements. Main contributions are summarized as follows:

- We propose TG-LLaVA, a text-guided architecture based on learnable latent embeddings which is different even opposite to most of existing VLM optimization approaches, which open up a new and worthwhile research avenue for consideration.
- The proposed TG-FOM module and TG-DP module can be universally applied as a modular plug-in to mainstream VLM frameworks, consistently brings improvement.
- Through extensive experiments on various settings of VLM variations and numerous multimodal tasks, we show that our proposed TG-LLaVA not only delivers substantial benefits but also provides valuable insights and methodologies for the existing VLM research field.

# Related Work

## Vision Language Models

VLMs primarily consist of a visual encoder and a LLM, representing prominent architectures in the multimodal domain. Researchers have proposed numerous architectures (Li et al. 2023a; Zhu et al. 2024; Chen et al. 2023b) for integrating visual features into advanced LLM inference pipelines. Llama-Adapter (Zhang et al. 2023) proposes to generate language answer with taking the image input as condition. Flamingo (Alayrac et al. 2022) and LLaVA (Liu et al. 2024c) blend visual tokens with text as inputs to LLM, differing in that Flamingo employs gating mechanisms to inject encoded visual features into LLMs, while LLaVA directly concatenates visual and textual features at input. Complementarily, the availability of high-quality image-text pairs for VLM training is crucial. Several methods use Chat-GPT (OpenAI 2023a) and GPT-4 (OpenAI 2023b) to construct large-scale, high-quality datasets (Zhu et al. 2023; Liu et al. 2024c; Zhao et al. 2023).

Inspired by the compact structure and outstanding performance of LLaVA-1.5 (Liu et al. 2024a), we use LLaVA-1.5 as our baseline and incorporate a text-guided approach, similar to other LLaVA-based methods. Unlike most of these methods, which create additional datasets to enhance performance, our improvements focus entirely on the model architecture itself. This approach can further enhance the performance of methods that rely on extra datasets. The results in fifth and sixth lines of Table 1 verified this point.

## Image-Text Alignment

Align the visual and text information in high semantic level is the base for building VLMs. Centered around this problem, researchers have done extensive work (Chen et al. 2024b). Previous researchers have typically employed contrastive learning across modalities and autoregressive learning for text. CLIP (Radford et al. 2021) and SigLIP (Zhai et al. 2023) trained encoders on massive datasets, laying foundational work for aligning visual and textual modalities and significantly advancing subsequent VLM developments. BLIP (Li et al. 2022) meticulously design multiple loss functions for contrastive and generative learning, achieving refined cross-modal alignment through multi-stage training. BLIP-2 (Li et al. 2023b) adopts a Q-former structure, interacting with the visual modality using learnable query vectors before merging with the text modality. Many LLaVA-like approaches use simple MLPs for modal alignment, with subsequent works like MobileVLM V2 (Chu et al. 2024).

Both image-text alignment methods and our proposed TG-LLaVA recognize the importance of integrating textual and visual information. However, while these methods focus on bridging different modalities, our approach leverages the textual modality to guide and optimize the visual modality. This alignment makes the operation of VLMs more consistent with the purpose-driven logic of human behavior in real-world scenarios.

## Visual Encoder in VLMs

To enable the LLM to extract more information from the input visual image, various strategies have been proposed for utilizing visual features. DenseConnector (Yao et al. 2024) employs dense connections to link visual features across different levels, feeding the combined features into a connector. TokenPacker (Li et al. 2024a) merges visual features from the high-resolution branch with those from the low-resolution branch to generate condensed visual tokens. Idefics2 (Laurençon et al. 2024) compresses visual

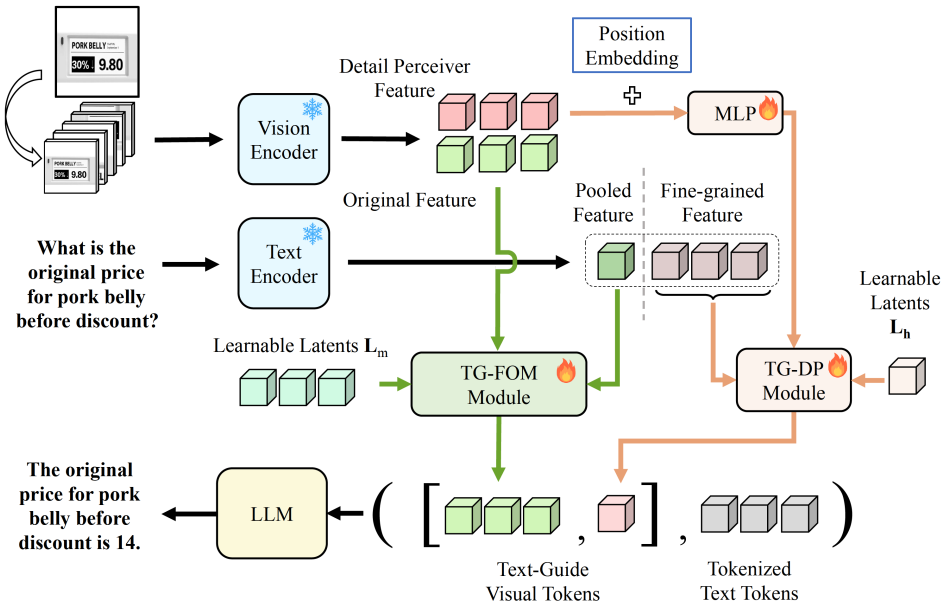


Figure 2: Overall framework of the proposed TG-LLaVA. Text-guided visual feature optimization mask (TG-FOM) module is designed to optimize the visual feature with the guidance of global text. Text-guided detail perceiver (TG-DP) module is proposed to capture instruction relevant details.

features using a perceiver structure, significantly reducing the number of visual tokens compared to other approaches. Approaches like Mini-Gemini (Li et al. 2024b), LLaVA-Next (Liu et al. 2024b), Qwen-VL (Bai et al. 2023b), and InternLM (Dong et al. 2024) leverage high-resolution images to capture finer visual feature details. ImageBind-LLM (Han et al. 2023) and Llama3.1 (Meta AI 2024b) explore injecting visual modality features into LLMs, with the former using trainable gating modules to add visual features to word tokens, and the latter introducing visual information across different layers of LLM through periodic cross-attention.

Unlike methods that focus on better utilizing existing visual features, our proposed TG-LLaVA aims to enhance the visual features themselves by using textual guidance. In contrast to ImageBind-LLM and Llama3.1, which incorporate image features into the LLM component, our approach integrates text into the visual encoder.

## Method

In this section, we first review the classic VLM architecture, using LLaVA (Liu et al. 2024c) as a representative example, to provide an overview of the VLM paradigm. Following this, we present a detailed explanation of the proposed TG-LLaVA, focusing on the implementation of text-guided visual feature optimization mask module and text-guided detail perceiver module.

### A Revisit of VLMs

Taking LLaVA (Liu et al. 2024c) as an example, the primary goal of VLMs is to effectively harness the capabilities of pre-trained LLM and visual model. The three key components of such framework can be defined as follows: 1) **Visual Encoder** $\mathrm{E_v}$, typically utilizing a pre-trained vision transformer like CLIP, is designed to partition the input image $\mathbf{I} \in \mathbb{R}^{H\times W\times C}$ into several patches with equal size and further encode them into visual features $\mathbf{F}_\mathrm{i} \in \mathbb{R}^{N\times D}$. Here, $H$ and $W$ represent the size of the input image, $C$ denotes the number of channels, $N$ corresponds to the number of patches in the output features, and $D$ represents the feature dimension of each encoded patch. When the patch size is $P$, $N = HW/P^2$. 2) **Connector** C (also referred as Projector) consists of two linear layers with a GELU activation function in between. Its purpose is to map visual features into the embedding space of the LLM, converting $\mathbf{F}_\mathrm{i}$ into visual tokens $\mathbf{T}_\mathrm{v}$. 3) **LLM** L employs a tokenizer and text embedding module to sequentially transform textual data into token IDs and their corresponding embedded tokens $\mathbf{T}_\mathrm{t}$, effectively converting the language into the feature space of its input. Within the VLM architecture, these textual tokens $\mathbf{T}_\mathrm{t}$ are concatenated with the aligned visual tokens $\mathbf{T}_\mathrm{v}$ processed by the connector, forming the input for the LLM to carry out subsequent predictions. For a sequence of length $L$, the probability of VLM predicting the target answer tokens $\mathbf{T}_\mathrm{a} = \{t_i\}_{i=1}^{L}$ can be formalized as:

$$p\left(\mathbf{T}_\mathrm{a} \mid \mathbf{T}_\mathrm{v}, \mathbf{T}_\mathrm{t}\right) = \prod_{i=1}^{L} p_\theta\left(t_i \mid \mathbf{T}_\mathrm{v}, \mathbf{T}_{\mathrm{t},<i}, \mathbf{T}_{\mathrm{a},<i}\right), \quad (1)$$

where $\theta$ represents all the trainable parameters in the VLM. In this VLM prediction paradigm, the visual features are directly obtained by encoding the raw input image through $\mathrm{E_v}$ without any interaction with the textual modality. This approach contrasts with the purpose-driven nature of human behavior. Optimizing the encoded features based on textual

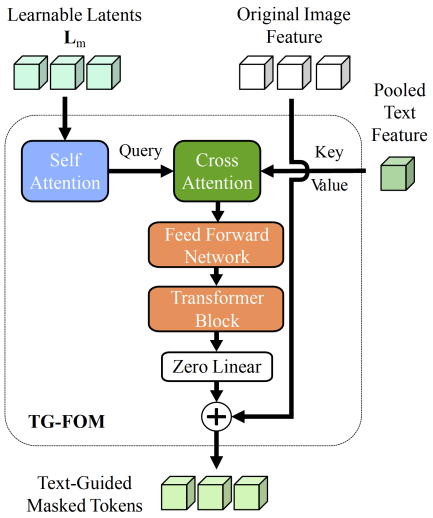


Figure 3: Illustration of Text-guided Visual Feature Optimization Mask module.

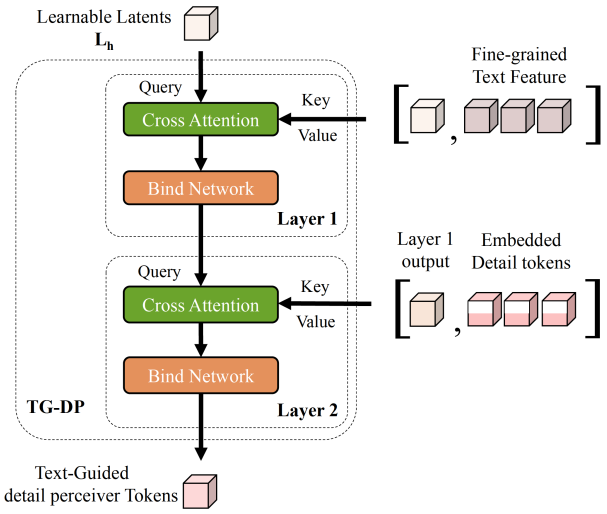


Figure 4: Illustration of Text-guided Detail Perceiver module.

instructions is more conducive to enabling the VLM to generate accurate responses.

## Text Guided LLaVA

Inspired by the reasoning logic humans use in visual question answering scenarios, we design TG-LLaVA, a novel approach that optimizes visual features to align the inference process of VLM more closely with purpose-driven human behavior, thereby further enhancing the capabilities of VLMs. As illustrated in Figure 2, TG-LLaVA primarily consists of two components: Text-Guided Visual Feature Optimization Mask (TG-FOM) and Text-Guided Detail Perceiver (TG-DP). The former uses learnable latents to parse the global information from textual instructions and attaches it as a mask to the output of visual encoder, optimizing features based on textual instructions. The latter employs another set of latents, first interacting with the detailed information from textual instructions, and then extracting fine-grained details from high-resolution patches of the input image based on these instructions. These details are concatenated with the original features, further refining the visual modality input of VLM. The specifics of this approach will be elaborated in the following sections.

**Text Guided Visual Feature Optimization Mask** In current VLMs, the visual representations typically originate solely from the final layer features of the visual encoder $\mathrm{E_v}$. Features obtained through this pipeline encompass the global information of the input image $\mathbf{I}$. However, the corresponding textual instructions often focus on specific local targets within the image. As a result, the information related to these focal targets is easily compromised when confronted with irrelevant or even contradictory information, leading to distorted judgments by the VLM. To address this issue, we design TG-FOM module to optimize visual features based on textual instructions, thereby endowing VLMs with the advantage of purpose-driven human behavior. Figure 3 illustrates the specific framework of the FOM module.

We begin by initializing a set of learnable latent embeddings $\mathbf{L}_\mathrm{m}$ that are of the same number as visual tokens. The purpose of these latents is to extract linguistic information from the textual instructions and add it as a mask to the original features. Here, we design a double-layer Q-former to parse semantic information from textual instructions, serving as a bridge between global text and visual features. In this structure, the cross-attention layers incorporate the pooled textual instruction features $\mathbf{F}_\mathrm{t}^\mathrm{p}$ encoded by CLIP text encoder $\mathrm{E_t}$ as Key and Value for interaction with Query $\mathbf{L}_\mathrm{m}$, and the final output is a mask generated based on the textual information, which is then applied to the visual features. We additionally introduce a zero-initialized linear layer to ensure that the optimization of the original visual features remains a gradual process. The formulation is:

$$\begin{aligned} \mathbf{M}_\mathrm{t} = \mathrm{Q}(\mathbf{L}_\mathrm{m}, \mathbf{F}_\mathrm{t}^\mathrm{p}) = \mathrm{TB}(\mathrm{FFN}(\mathrm{A}_\mathrm{cross}(\mathrm{A}_\mathrm{self}(\mathbf{L}_\mathrm{m}), \mathbf{F}_\mathrm{t}^\mathrm{p}))), \\ \mathbf{F}_\mathrm{i}^* = \mathbf{F}_\mathrm{i} + \mathrm{Z}(\mathbf{M}_\mathrm{t}), \end{aligned} \tag{2}$$

where $\mathbf{M}_\mathrm{t}$ represents the mask obtained by extracting semantic information from the textual instructions via the learnable $\mathbf{L}_\mathrm{m}$, $\mathrm{A}_\mathrm{cross}$ and $\mathrm{A}_\mathrm{self}$ denote the cross-attention and self-attention modules, respectively, FFN represents the feed-forward neural network, Z represents the zero-initialized linear layer used as a buffer during feature addition, $\mathbf{F}_\mathrm{i}^*$ represents the visual features optimized by text guidance and TB denotes a single Transformer layer, encompassing both $\mathrm{A}_\mathrm{self}$ and FFN.

**Text Guided Detail Perceiver** When observing images, in addition to selecting focal points based on the instruction, humans can also adjust their focus to obtain more detailed information. Following this idea, we design TG-DP, which is responsible for capturing instruction relevant details.

As shown in Figure 2, we scale up the original image $\mathbf{I}$ to preserve more details, then divide it into patches that match

the size of the original image. This design ensures that we can extract all visual features with a single call to the visual encoder. After obtaining the visual features of these patches, we add positional embeddings and a learnable MLP layer to recover the spatial structure information that was disrupted during the division operation, getting corrected visual features $\mathbf{F}_{\mathrm{i}}^{\mathrm{h}}$. So far, the visual tokens containing detailed information are ready. Along with the learnable latent embeddings $\mathbf{L}_{\mathrm{h}}$ and the fine-grained textual instruction features $\mathbf{F}_{\mathrm{t}}^{\mathrm{g}}$, these visual tokens will be fed into the TG-DP module, where they will be selected and integrated according to the guidance of the text.

As shown in Figure 4, we set up $\mathbf{L}_{\mathrm{h}}$ to interact with $\mathbf{F}_{\mathrm{t}}^{\mathrm{g}}$ output by the text encoder $\mathrm{E_t}$. Here, the number of $\mathbf{L}_{\mathrm{h}}$ is much smaller than the number of the original visual tokens, ensuring that the visual tokens input to the LLM do not increase significantly, thus maintaining inference efficiency. Ablation studies demonstrate that this compression does not negatively impact the final results. The proposed TG-DP module has two perception layers:

- The first perception layer is responsible for parsing fine-grained text to generate text guidance tokens. It receives $\mathbf{L}_{\mathrm{h}}$ and $\mathbf{F}_{\mathrm{t}}^{\mathrm{g}}$, maintaining $\mathbf{L}_{\mathrm{h}}$ as the Query and $\mathbf{F}_{\mathrm{t}}^{\mathrm{g}}$ as the Key and Value, with the distinction that $\mathbf{F}_{\mathrm{t}}^{\mathrm{g}}$ is concatenated with $\mathbf{L}_{\mathrm{h}}$.
- The second layer is in charge of generating detail perceiver tokens with the guidance of fine-grained text. In the second layer, the Key and Value are replaced by $\mathbf{F}_{\mathrm{i}}^{\mathrm{h}}$, using textual instruction features parsed through the first layer as Query for a second interaction. The output of the second layer is the compressed visual tokens $\mathbf{F}_{\mathrm{i}}^{\mathrm{h}}$.

Due to the significant difference between the feature space of $\mathbf{F}_{\mathrm{i}}^{\mathrm{h}}$ and the original VLM visual features, we design a dedicated connector $\mathrm{C^h}$ for $\mathbf{F}_{\mathrm{i}}^{\mathrm{h}}$. The entire process can be formalized as:

$$\begin{aligned}\mathbf{F}^{\mathrm{L1}} &= \mathrm{BN}^{\mathrm{L1}}(\mathrm{A}_{\mathrm{cross}}^{\mathrm{L1}}(\mathbf{L}_{\mathrm{h}}, \mathrm{CAT}(\mathbf{F}_{\mathrm{t}}^{\mathrm{g}}, \mathbf{L}_{\mathrm{h}}))),\\ \mathbf{F}_{\mathrm{i}}^{\mathrm{h}} &= \mathrm{C^h}(\mathrm{BN}^{\mathrm{L2}}(\mathrm{A}_{\mathrm{cross}}^{\mathrm{L2}}(\mathbf{F}^{\mathrm{L1}}, \mathrm{CAT}(\mathbf{F}_{\mathrm{i}}^{\mathrm{h}}, \mathbf{F}^{\mathrm{L1}})))),\end{aligned} \tag{3}$$

where $\mathrm{L}i$ denotes the $i^{th}$ layer within DP module, $\mathbf{F}^{\mathrm{L1}}$ is the output of first layer and $\mathrm{CAT}$ represents the concatenation operation. $\mathrm{BN}^{\mathrm{L}i}(i \in (1,2))$ denotes Bind Network which can be formalized as:

$$\mathrm{BN}(\mathbf{X}) = \mathbf{X} + \left(\mathbf{X}\mathrm{W_2}^{\mathrm{up}} \cdot \mathrm{SiLU}\left(\mathbf{X}\mathrm{W_1}^{\mathrm{up}}\right)\right)\mathrm{W_3}^{\mathrm{down}}. \tag{4}$$

## Overall

At this point, with the guidance of input text, we have obtained the optimized visual features $\mathbf{F}_{\mathrm{i}}^{*}$, as well as the detail perceiver tokens $\mathbf{F}_{\mathrm{i}}^{\mathrm{h}}$. We then concatenate the features obtained from the original VLM connector $\mathrm{C}$ with $\mathbf{F}_{\mathrm{i}}^{\mathrm{h}}$, which together form the final visual tokens $\mathbf{T}_{\mathrm{v}}^{\mathrm{fin}}$ input for the VLM. The final prediction process of VLM can be represented as :

$$\begin{aligned}\mathbf{T}_{\mathrm{v}}^{\mathrm{fin}} &= \mathrm{CAT}(\mathrm{C}(\mathbf{F}_{\mathrm{i}}^{*}), \mathbf{F}_{\mathrm{i}}^{\mathrm{h}}),\\ p\left(\mathbf{T}_{\mathrm{a}} \mid \mathbf{T}_{\mathrm{v}}^{\mathrm{fin}}, \mathbf{T}_{\mathrm{t}}\right) &= \prod_{i=1}^{L} p_\theta\left(t_i \mid \mathbf{T}_{\mathrm{v}}^{\mathrm{fin}}, \mathbf{T}_{\mathrm{t},<i}, \mathbf{T}_{\mathrm{a},<i}\right).\end{aligned} \tag{5}$$

# Experiment

In this section, we first present the detailed experimental setup. We then enumerate the improvements brought by our proposed TG-LLaVA over the baseline across multiple evaluation metrics, and compare our method with several state-of-the-art (SoTA) approaches under various configurations. Specifically, we visualize the attention map to demonstrate the efficacy of proposed TG-LLaVA. Finally, we conduct ablation studies and provide an analysis of the results.

## Experimental Settings

**Implementation Details** We implement the proposed improvement strategy on top of LLaVA-1.5 (Liu et al. 2024a). Specifically, we maintain consistency with LLaVA-1.5 by employing CLIP-ViT-L/14-336px as the visual encoder. To further validate the generalizability of our proposed method, we also incorporate SigLIP-SO400m-patch14-384, another leading choice, for comparative analysis. In terms of LLM, we compare our method against the baseline using Vicuna-7/13B and extend our approach to Llama3-8B (Meta AI 2024a) and Qwen2-7B (Yang et al. 2024), thereby demonstrating the versatility of our method. For training configurations, we adhere strictly to the settings outlined in the original LLaVA-1.5 paper to ensure fairness, with learning rates of 1e-3 and 2e-5 for pre-training and instruction fine-tuning phases, respectively, and maintaining batch sizes of 256 and 128. DP module introduces 64 additional visual tokens. The training process for TG-LLaVA utilizes the PyTorch framework and employs 8 H100-80G GPUs.

**Datasets** Focusing on proposing a novel optimization method for the VLM framework, we do not incorporate any additional data beyond the LLaVA-1.5 open-source dataset (Liu et al. 2024a), which has 558K image captions for pre-training and 665K conversations for instruction tuning. We also apply our TG-LLaVA to the Mini-Gemini dataset (Reid et al. 2024), which consists of 1.2M + 1.5M data. For evaluation, we conduct extensive experiments and report results on widely-adopted VLM benchmarks using the VLMEvalKit (Duan et al. 2024) platform to provide robust and comprehensive performance validation for the proposed TG-LLaVA. The evaluation datasets include: MMBench (MMB) (Liu et al. 2023a), MMS (MM-Star) (Chen et al. 2024a), MMMU (Yue et al. 2024), MV (MathVista) (Lu et al. 2023), OCRB (OCRBench) (Liu et al. 2023b), AI2D (Hiippala et al. 2021), HB (HallusionBench)(Guan et al. 2024), LB (LLaVABench) (Liu et al. 2024c), SQA (ScienceQA) (Saikh et al. 2022), and MME (Fu et al. 2024).

## Genuine Improvement Over the Baseline

In Table 1, we present the performance improvements of the proposed method across various configurations compared to the baseline. According to the experimental results, we can draw several phenomenons:

- The proposed text-guided strategy demonstrates substantial improvements over the baseline. Compared with the original LLaVA-1.5, TG-LLaVA achieve much better performance. As shown in the first four rows, our

| Method | LM | VE | PT + IT | MMB | MMS | MMMU | MV | OCRB | AI2D | HB | LB | SQA | MME |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Performance comparison against the baseline* | | | | | | | | | | | | | |
| LLaVA-1.5 | Vicuna-7B | CLIP-L | 0.5M+0.6M | 59.1 | 33.1 | 35.7 | 25.6 | 31.8 | 55.5 | 27.6 | 61.8 | 69.2 | **1808** |
| TG-LLaVA | Vicuna-7B | CLIP-L | 0.5M+0.6M | **61.3** | **35.5** | **38.1** | **26.7** | **32.6** | **56.9** | **29.2** | **65.0** | **70.6** | 1779 |
| LLaVA-1.5 | Vicuna-13B | CLIP-L | 0.5M+0.6M | 64.0 | 34.3 | 37.0 | **27.7** | 33.7 | **61.1** | 24.5 | 66.1 | 72.6 | 1781 |
| TG-LLaVA | Vicuna-13B | CLIP-L | 0.5M+0.6M | **65.3** | **35.9** | **39.6** | 27.6 | **34.4** | 61.0 | **25.9** | **67.8** | **72.7** | **1858** |
| *Expanding to larger training datasets* | | | | | | | | | | | | | |
| LLaVA-1.5 | Vicuna-7B | CLIP-L | 1.2M+1.5M | 62.8 | 39.0 | 35.2 | **32.6** | 37.3 | 69.8 | 25.4 | **60.7** | 70.5 | 1810 |
| TG-LLaVA | Vicuna-7B | CLIP-L | 1.2M+1.5M | **63.5** | **39.4** | **37.2** | 32.4 | **37.9** | **70.0** | **27.8** | 59.9 | **70.9** | **1840** |
| *Expanding to robust visual encoder* | | | | | | | | | | | | | |
| LLaVA-1.5 | Vicuna-7B | SigLIP-SO | 0.5M+0.6M | 62.8 | 34.9 | 38.6 | 27.0 | 36.3 | **59.3** | 28.1 | 66.8 | **70.6** | 1764 |
| TG-LLaVA | Vicuna-7B | SigLIP-SO | 0.5M+0.6M | **63.1** | **37.7** | **38.9** | **27.7** | **37.3** | 58.4 | **28.5** | **67.9** | 70.0 | **1803** |
| *Expanding to other LLMs* | | | | | | | | | | | | | |
| LLaVA-1.5 | Llama3-8B | CLIP-L | 0.5M+0.6M | **66.7** | 38.5 | 40.7 | 26.7 | **33.4** | **61.8** | 27.4 | 64.3 | 74.8 | 1789 |
| TG-LLaVA | Llama3-8B | CLIP-L | 0.5M+0.6M | 65.2 | **40.5** | **41.0** | **28.6** | 32.8 | 60.2 | **29.2** | **65.6** | **75.9** | **1801** |
| LLaVA-1.5 | Qwen2-7B | CLIP-L | 0.5M+0.6M | 70.9 | 42.1 | 43.6 | **32.2** | **33.6** | **65.3** | 28.3 | 65.9 | 74.2 | 1849 |
| TG-LLaVA | Qwen2-7B | CLIP-L | 0.5M+0.6M | **71.2** | **43.5** | **44.7** | 31.3 | 33.4 | 64.6 | **29.2** | **66.3** | **75.1** | **1941** |

Table 1: Performance comparison between various baselines and TG-LLaVA. The results of the first and the third line are sourced from the official OpenCompass publicly available leaderboard (Duan et al. 2024), while the remaining results are derived from our own replication. The best results are **bold**. LM, VE, PT and IT denote Language Model, Vision Encoder, pre-training data and instruction fine-tuning data, respectively.

method leads on the majority of evaluation datasets. It is noteworthy that TG-LLaVA demonstrates an average improvement of 1.5% over the original LLaVA-1.5 across ten datasets when using Vicuna-7B, highlighting the method's significant value. When juxtaposed with the baseline LLaVA-1.5 Vicuna-7B model, we enhance performance metrics by +2.2% on MMBench, +2.4% on both MMStar and MMMU, and +3.2% on LLaVABenchs, respectively. For LLaVA-1.5 with Vicuna-13B, we also achieve an average performance improvement of 1%. Specifically, we see a +1.6% gain on MMStar, a +2.0% gain on MMMU, and a +3.2% gain on MME. These impressive results further validate the contribution of the proposed TG-LLaVA architecture to visual feature optimization, highlighting the favorable impact of our method.

- The proposed TG-FOM and TG-DP modules can be universally applied as a modular plug-in to mainstream VLM frameworks. As shown in the rest part of Table 1. we further validate the versatility of our proposed method under various settings. We replace CLIP with SigLIP and substitute Vicuna with Llama3 and Qwen2 on top of the original LLaVA-1.5 framework. We compare these settings with our method as the baseline. The results in Table 1 confirm that our method continues to maintain a leading advantage across most datasets, demonstrating that the proposed TG-LLaVA exhibits excellent generalizability and possesses strong potential for adaptation to a wide range of VLM architectures.

## Comparison with other LLaVA-based Methods

In Table 2, we compare the proposed proposed TG-LLaVA with other concurrent works which also take LLaVA as baseline. The methods we include for comparison are Seeing the Image (Xiao et al. 2024), TokenPacker (Li et al. 2024a), and DenseConnector (Yao et al. 2024). Since these comparison methods are relatively new and have not been evaluated on the OpenCompass leaderboard, we employ the evaluation scripts from LLaVA-1.5 to maintain a fair and consistent framework for our comparisons.

| Method | Source | MME | MMB | MMVet | GQA |
|---|---|---|---|---|---|
| LLaVA-1.5 | NeurIPS 23 | 1531 | 67.7 | 35.4 | 63.3 |
| Seeing the image | Arxiv 2405 | 1567 | - | - | - |
| TokenPacker | Arxiv 2407 | - | 68.0 | 34.5 | 62.5 |
| DenseConnector$^\star$ | Arxiv 2405 | 1540 | 70.0 | - | - |
| TG-LLaVA | - | **1603** | **70.2** | **36.6** | **63.4** |

Table 2: Performance comparison with contemporaneous methods. $\star$ denotes results obtained with official code reproductions. Note: MME metric here considers only the Perception part.

## Quantitative Comparison with SoTAs

We further compare our method with several leading approaches. The methods included in the comparison are MiniGPT4 (Zhu et al. 2023), Qwen-VL(Bai et al. 2023b), VisualGLM(GLM et al. 2024), PandaGPT(Su et al. 2023), mPLUG-Owl2(Ye et al. 2023), Emu2-chat(Sun et al. 2024), Yi-VL(Young et al. 2024) and ShareGPT-4V(Chen et al. 2023a). Table 3 presents the performance comparison across multiple benchmarks.

Remarkably, despite relying solely on settings from LLaVA-1.5, our TG-LLaVA achieves performance that matches or surpasses the benchmarks set by leading SoTA methods, with a comparatively smaller volume of pre-training and instruction fine-tuning data.

| Method | LLM | VE | PT + IT | MMB | MMS | MMMU | MV | OCRB | AI2D | HB | LB | SQA | MME |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MiniGPT4 | Vicuna-7B | EVA-G | 5M+3.5K | 20.8 | 16.3 | 23.6 | 20.4 | 17.2 | 28.4 | 31.9 | 45.1 | 39.6 | 1047 |
| Qwen-VL | Qwen-7B | ViT-G/16 | 1.4B+50M | 32.9 | 32.5 | 29.6 | 15.5 | 12.7 | 57.7 | 29.9 | 12.9 | 61.1 | 483 |
| VisualGLM | ChatGLM-6B | EVA-CLIP | 330M | 35.7 | 25.9 | 29.9 | 21.9 | 17.0 | 41.2 | 25.0 | 37.3 | 56.1 | 738 |
| PandaGPT | Vicuna-13B | IB-H | 160K | 34.5 | 25.6 | 32.9 | 25.0 | 26.9 | 48.3 | 21.6 | 57.2 | 61.8 | 1076 |
| mPLUG-Owl2 | Llama 2-7B | CLIP-L | 348M+1.2M | 60.8 | 34.8 | 34.7 | 25.4 | 25.5 | 55.7 | 29.4 | 59.9 | 69.5 | 1786 |
| Emu2-chat | Llama-33B | EVA-CLIP | - | 52.8 | 40.7 | 35.0 | 30.7 | **43.6** | 49.7 | 29.5 | 56.4 | 68.2 | 1678 |
| Yi-VL | Yi-6B | CLIP-L | 100M+26M | 64.2 | 33.7 | 40.3 | 29.7 | 29.0 | 59.8 | **36.0** | 51.9 | 72.6 | 1915 |
| ShareGPT-4V | Vicuna-7B | CLIP-L | 1.2M+0.7M | 61.6 | 35.7 | 37.2 | 26.5 | 37.1 | 58.0 | 28.6 | 66.9 | 69.5 | 1914 |
| TG-LLaVA | Vicuna-7B | SigLIP-SO | 0.5M+0.6M | 63.1 | 37.7 | 38.9 | 27.7 | 37.3 | 58.4 | 28.5 | **67.9** | 70.0 | 1803 |
| TG-LLaVA | Vicuna-7B | CLIP-L | 1.2M+1.5M | 63.5 | 39.4 | 37.2 | **32.4** | 37.9 | **70.0** | 27.8 | 59.9 | 70.9 | 1840 |
| TG-LLaVA | Llama3-8B | CLIP-L | 0.5M+0.6M | 65.2 | 40.5 | 41.0 | 28.6 | 32.8 | 60.2 | 29.2 | 65.6 | **75.9** | 1801 |
| TG-LLaVA | Qwen2-7B | CLIP-L | 0.5M+0.6M | **71.2** | **43.5** | **44.7** | 31.3 | 33.4 | 64.6 | 29.2 | 66.3 | 75.1 | **1941** |

Table 3: Comparison with SoTA methods. The best results are **bold** and the second-best results are underlined. Results of all other methods are obtained from the OpenCompass public leaderboard.

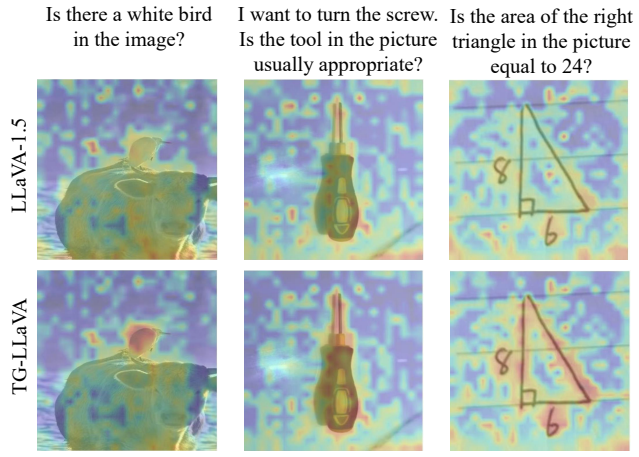


Figure 5: Attention maps for TG-LLaVA versus LLaVA-1.5 using Llama3-8B.

## Qualitative Analysis via Visualization

To demonstrate the optimization effect of the proposed method, we visualize the attention maps between the visual features and text instructions in both the baseline model and our model. These visualizations provide insights into how the proposed visual feature optimization module operates. We aggregate the attention scores between image tokens and textual instruction tokens across all layers to compute the results. As shown in Figure 5, the TG-LLaVA architecture push the model to focus on regions highlighted by the textual instructions, assigning greater attention weights to them.

## Ablation Studies

We further conduct in-depth ablation studies to analyze the effectiveness of each component of our approach. Results are listed in Table 4. By sequentially introducing the FOM and DP modules, we observe significant improvements in model performance, underscoring the effectiveness of our proposed visual feature optimization algorithm. Additionally, we conduct experiments on the number of additional visual tokens introduced by the DP module. The results show that introducing too few tokens yields suboptimal performance gains, while introducing too many tokens can actually harm performance. Therefore, we choose a balanced configuration to achieve optimal performance.

| Setting | MMB | MV | AI2D | SQA |
|---|---|---|---|---|
| Baseline | 59.1 | 25.6 | 55.5 | 69.2 |
| Only FOM | 60.4 | 26.0 | 55.1 | 67.5 |
| Only DP | 58.7 | 26.1 | 56.0 | 69.3 |
| DP patch 32 | 60.7 | 26.1 | 56.7 | 70.5 |
| DP patch 128 | 60.5 | 26.2 | 56.9 | 69.1 |
| DP patch 256 | 61.3 | 25.9 | 55.4 | 68.5 |
| Final | 61.3 | 26.7 | 56.9 | 70.6 |

Table 4: Ablation study results on FOM and DP modules, and impact of the additional visual token count introduced in DP module.

## Conclusion

In this paper, we introduce TG-LLaVA, an innovative VLM optimization technique that guides the vision encoder using text. By emulating human-like purpose-driven logic, we leverage learnable embeddings to analyze text and enhance the vision encoder. Our experiments reveal that TG-LLaVA outperforms similar methods and is adaptable to various frameworks, consistently yielding improvements. This text-guided enhancement of the visual encoder opens up a new pathway for advancing VLMs. For future work, we aim to further refine the visual feature extraction process guided by text to achieve even better performance.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62401471; in part by 2024 Gusu Innovation and Entrepreneurship Leading Talents Program under Grant ZXL2024333; in part by National Key R&D Program of China, 62206244.

## References


Alayrac, J.-B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; Lenc, K.; Mensch, A.; Millican, K.; Reynolds, M.; et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35: 23716–23736.

Awadalla, A.; Gao, I.; Gardner, J.; Hessel, J.; Hanafy, Y.; Zhu, W.; Marathe, K.; Bitton, Y.; Gadre, S.; Sagawa, S.; et al. 2023. Openflamingo: An open-source framework for training large autoregressive vision-language models. arXiv:2308.01390.

Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; Fan, Y.; Ge, W.; Han, Y.; Huang, F.; et al. 2023a. Qwen technical report. arXiv:2309.16609.

Bai, J.; Bai, S.; Yang, S.; Wang, S.; Tan, S.; Wang, P.; Lin, J.; Zhou, C.; and Zhou, J. 2023b. Qwen-vl: A frontier large vision-language model with versatile abilities. arXiv:2308.12966.

Chen, L.; Li, J.; Dong, X.; Zhang, P.; He, C.; Wang, J.; Zhao, F.; and Lin, D. 2023a. ShareGPT4V: Improving Large Multi-Modal Models with Better Captions. arXiv:2311.12793.

Chen, L.; Li, J.; Dong, X.; Zhang, P.; Zang, Y.; Chen, Z.; Duan, H.; Wang, J.; Qiao, Y.; Lin, D.; et al. 2024a. Are We on the Right Way for Evaluating Large Vision-Language Models? arXiv:2403.20330.

Chen, X.; Wang, X.; Beyer, L.; Kolesnikov, A.; Wu, J.; Voigtlaender, P.; Mustafa, B.; Goodman, S.; Alabdulmohsin, I.; Padlewski, P.; et al. 2023b. Pali-3 vision language models: Smaller, faster, stronger. arXiv:2310.09199.

Chen, Z.; Wu, J.; Wang, W.; Su, W.; Chen, G.; Xing, S.; Zhong, M.; Zhang, Q.; Zhu, X.; Lu, L.; et al. 2024b. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24185–24198.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3): 6.

Chu, X.; Qiao, L.; Zhang, X.; Xu, S.; Wei, F.; Yang, Y.; Sun, X.; Hu, Y.; Lin, X.; Zhang, B.; et al. 2024. Mobilevlm v2: Faster and stronger baseline for vision language model. arXiv:2402.03766.

Dong, X.; Zhang, P.; Zang, Y.; Cao, Y.; Wang, B.; Ouyang, L.; Wei, X.; Zhang, S.; Duan, H.; Cao, M.; et al. 2024. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model.

Duan, H.; Yang, J.; Qiao, Y.; Fang, X.; Chen, L.; Liu, Y.; Dong, X.; Zang, Y.; Zhang, P.; Wang, J.; Lin, D.; and Chen, K. 2024. VLMEvalKit: An Open-Source Toolkit for Evaluating Large Multi-Modality Models. arXiv:2407.11691.

Fu, C.; Chen, P.; Shen, Y.; Qin, Y.; Zhang, M.; Lin, X.; Yang, J.; Zheng, X.; Li, K.; Sun, X.; Wu, Y.; and Ji, R. 2024. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models. arXiv:2306.13394.

GLM, T.; Zeng, A.; Xu, B.; Wang, B.; Zhang, C.; Yin, D.; Rojas, D.; Feng, G.; Zhao, H.; Lai, H.; Yu, H.; Wang, H.; Sun, J.; Zhang, J.; Cheng, J.; Gui, J.; Tang, J.; Zhang, J.; Li, J.; Zhao, L.; Wu, L.; Zhong, L.; Liu, M.; Huang, M.; Zhang, P.; Zheng, Q.; Lu, R.; Duan, S.; Zhang, S.; Cao, S.; Yang, S.; Tam, W. L.; Zhao, W.; Liu, X.; Xia, X.; Zhang, X.; Gu, X.; Lv, X.; Liu, X.; Liu, X.; Yang, X.; Song, X.; Zhang, X.; An, Y.; Xu, Y.; Niu, Y.; Yang, Y.; Li, Y.; Bai, Y.; Dong, Y.; Qi, Z.; Wang, Z.; Yang, Z.; Du, Z.; Hou, Z.; and Wang, Z. 2024. ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools. arXiv:2406.12793.

Guan, T.; Liu, F.; Wu, X.; Xian, R.; Li, Z.; Liu, X.; Wang, X.; Chen, L.; Huang, F.; Yacoob, Y.; et al. 2024. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14375–14385.

Han, J.; Zhang, R.; Shao, W.; Gao, P.; Xu, P.; Xiao, H.; Zhang, K.; Liu, C.; Wen, S.; Guo, Z.; et al. 2023. Imagebind-llm: Multi-modality instruction tuning. arXiv:2309.03905.

Hiippala, T.; Alikhani, M.; Haverinen, J.; Kalliokoski, T.; Logacheva, E.; Orekhova, S.; Tuomainen, A.; Stone, M.; and Bateman, J. A. 2021. AI2D-RST: A multimodal corpus of 1000 primary school science diagrams. *Language Resources and Evaluation*, 55: 661–688.

Laurençon, H.; Tronchon, L.; Cord, M.; and Sanh, V. 2024. What matters when building vision-language models? arXiv:2405.02246.

Li, B.; Zhang, Y.; Chen, L.; Wang, J.; Pu, F.; Yang, J.; Li, C.; and Liu, Z. 2023a. Mimic-it: Multi-modal in-context instruction tuning. arXiv:2306.05425.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023b. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 19730–19742. PMLR.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 12888–12900. PMLR.

Li, W.; Yuan, Y.; Liu, J.; Tang, D.; Wang, S.; Zhu, J.; and Zhang, L. 2024a. TokenPacker: Efficient Visual Projector for Multimodal LLM. arXiv:2407.02392.

Li, Y.; Zhang, Y.; Wang, C.; Zhong, Z.; Chen, Y.; Chu, R.; Liu, S.; and Jia, J. 2024b. Mini-gemini: Mining the potential of multi-modality vision language models. arXiv:2403.18814.

Lin, B.; Tang, Z.; Ye, Y.; Cui, J.; Zhu, B.; Jin, P.; Zhang, J.; Ning, M.; and Yuan, L. 2024. Moe-llava: Mixture of experts for large vision-language models. arXiv:2401.15947.

Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2024a. Improved Baselines with Visual Instruction Tuning. arXiv:2310.03744.

Liu, H.; Li, C.; Li, Y.; Li, B.; Zhang, Y.; Shen, S.; and Lee, Y. J. 2024b. Llava-next: Improved reasoning, ocr, and world knowledge.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2024c. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Liu, Y.; Duan, H.; Zhang, Y.; Li, B.; Zhang, S.; Zhao, W.; Yuan, Y.; Wang, J.; He, C.; Liu, Z.; et al. 2023a. Mmbench: Is your multi-modal model an all-around player? arXiv:2307.06281.

Liu, Y.; Li, Z.; Yang, B.; Li, C.; Yin, X.; Liu, C.-l.; Jin, L.; and Bai, X. 2023b. On the hidden mystery of ocr in large multimodal models. arXiv:2310.02255.

Lu, P.; Bansal, H.; Xia, T.; Liu, J.; Li, C.; Hajishirzi, H.; Cheng, H.; Chang, K.-W.; Galley, M.; and Gao, J. 2023. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. arXiv:2310.02255.

Meta AI. 2024a. Introducing Meta Llama 3: The most capable openly available LLM to date. https://ai.meta.com/blog/meta-llama-3/. Accessed: 2024-4-18.

Meta AI. 2024b. Meta Llama 3.1. https://ai.meta.com/blog/meta-llama-3-1/. Accessed: 2024-7-23.

OpenAI. 2023a. ChatGPT. https://openai.com/index/chatgpt/. Accessed: 2022-11-30.

OpenAI. 2023b. GPT-4V(ision) system card. https://openai.com/index/gpt-4v-system-card/. Accessed: 2023-9-25.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Reid, M.; Savinov, N.; Teplyashin, D.; Lepikhin, D.; Lillicrap, T.; Alayrac, J.-b.; Soricut, R.; Lazaridou, A.; Firat, O.; Schrittwieser, J.; et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. arXiv:2403.05530.

Saikh, T.; Ghosal, T.; Mittal, A.; Ekbal, A.; and Bhattacharyya, P. 2022. Scienceqa: A novel resource for question answering on scholarly articles. *International Journal on Digital Libraries*, 23(3): 289–301.

Su, Y.; Lan, T.; Li, H.; Xu, J.; Wang, Y.; and Cai, D. 2023. Pandagpt: One model to instruction-follow them all. arXiv:2305.16355.

Sun, Q.; Cui, Y.; Zhang, X.; Zhang, F.; Yu, Q.; Luo, Z.; Wang, Y.; Rao, Y.; Liu, J.; Huang, T.; and Wang, X. 2024. Generative Multimodal Models are In-Context Learners. arXiv:2312.13286.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. arXiv:2302.13971.

Wu, W.; Yao, H.; Zhang, M.; Song, Y.; Ouyang, W.; and Wang, J. 2023. GPT4Vis: what can GPT-4 do for zero-shot visual recognition? arXiv:2311.15732.

Xiao, X.; Wu, B.; Wang, J.; Li, C.; Zhou, X.; and Guo, H. 2024. Seeing the Image: Prioritizing Visual Correlation by Contrastive Alignment. arXiv:2405.17871.

Yang, A.; Yang, B.; Hui, B.; Zheng, B.; Yu, B.; Zhou, C.; Li, C.; Li, C.; Liu, D.; Huang, F.; et al. 2024. Qwen2 technical report. arXiv:2407.10671.

Yao, H.; Wu, W.; Yang, T.; Song, Y.; Zhang, M.; Feng, H.; Sun, Y.; Li, Z.; Ouyang, W.; and Wang, J. 2024. Dense Connector for MLLMs. arXiv:2405.13800.

Ye, Q.; Xu, H.; Ye, J.; Yan, M.; Hu, A.; Liu, H.; Qian, Q.; Zhang, J.; Huang, F.; and Zhou, J. 2023. mPLUG-Owl2: Revolutionizing Multi-modal Large Language Model with Modality Collaboration. arXiv:2311.04257.

Young, A.; Chen, B.; Li, C.; Huang, C.; Zhang, G.; Zhang, G.; Li, H.; Zhu, J.; Chen, J.; Chang, J.; et al. 2024. Yi: Open foundation models by 01. ai. arXiv:2403.04652.

Yue, X.; Ni, Y.; Zhang, K.; Zheng, T.; Liu, R.; Zhang, G.; Stevens, S.; Jiang, D.; Ren, W.; Sun, Y.; et al. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9556–9567.

Zhai, X.; Mustafa, B.; Kolesnikov, A.; and Beyer, L. 2023. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11975–11986.

Zhang, D.; Yu, Y.; Li, C.; Dong, J.; Su, D.; Chu, C.; and Yu, D. 2024. Mm-llms: Recent advances in multimodal large language models. arXiv:2401.13601.

Zhang, R.; Han, J.; Liu, C.; Gao, P.; Zhou, A.; Hu, X.; Yan, S.; Lu, P.; Li, H.; and Qiao, Y. 2023. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. arXiv:2303.16199.

Zhao, B.; Wu, B.; He, M.; and Huang, T. 2023. Svit: Scaling up visual instruction tuning. arXiv:2307.04087.

Zhu, D.; Chen, J.; Shen, X.; Li, X.; and Elhoseiny, M. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. arXiv:2304.10592.

Zhu, L.; Ji, D.; Chen, T.; Xu, P.; Ye, J.; and Liu, J. 2024. Ibd: Alleviating hallucinations in large vision-language models via image-biased decoding. arXiv:2402.18476.