MoCHA: Advanced Vision-Language Reasoning with MoE Connector and Hierarchical Group Attention

Yuqi Pang[1,2], **Bowen Yang**[1,2], **Yun Cao**[1,2*], **Rong Fan**[1,2], **Xiaoyu Li**[1,2], **Chen He**[1,2]

[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{pangyuqi, yangbowen, caoyun, fanrong, lixiaoyu, hechen}@iie.ac.cn

Abstract

Vision large language models (VLLMs) are focusing primarily on handling complex and fine-grained visual information by incorporating advanced vision encoders and scaling up visual models. However, these approaches face high training and inference costs, as well as challenges in extracting visual details, effectively bridging across modalities. In this work, we propose a novel visual framework, ***MoCHA***, to address these issues. Our framework integrates four vision backbones (i.e., CLIP, SigLIP, DINOv2 and ConvNeXt) to extract complementary visual features and is equipped with a sparse ***M***ixture ***o***f Experts ***C***onnectors (MoECs) module to dynamically select experts tailored to different visual dimensions. To mitigate redundant or insufficient use of the visual information encoded by the MoECs module, we further design a ***H***ierarchical Group ***A***ttention (HGA) with intra- and inter-group operations and an adaptive gating strategy for encoded visual features. We train MoCHA on two mainstream LLMs (e.g., Phi2-2.7B and Vicuna-7B) and evaluate their performance across various benchmarks. Notably, MoCHA outperforms state-of-the-art open-weight models on various tasks. For example, compared to CuMo (Mistral-7B), our MoCHA (Phi2-2.7B) presents outstanding abilities to mitigate hallucination by showing improvements of 3.25% in POPE and to follow visual instructions by raising 153 points on MME. Finally, ablation studies further confirm the effectiveness and robustness of the proposed MoECs and HGA in improving the overall performance of MoCHA.

Code — https://github.com/Pbhgit/MoCHA
Extended version — https://arxiv.org/abs/2507.22805

Introduction

Vision Large Language Models (VLLMs) (Liu et al. 2024b; Bai et al. 2025) have shown promising results in multimodal reasoning tasks by integrating visual encoders with Large Language Models (LLM) (OpenAI 2023, 2024; Touvron et al. 2023). However, the large model size and extensive training data present significant computational challenges. Additionally, excessive early reliance on language in VLLMs may act as a shortcut (Yuksekgonul et al. 2023) that results in suboptimal learning of effective visual representations. Therefore, exploring a VLLM that balances efficient visual processing and performance is a crucial topic.

Previous works have focused on strengthening vision encoders by utilizing extensive and higher-quality visual instruction data (Zheng et al. 2023), scaling up model size (Chen et al. 2024), or decomposing images into low-resolution patches (Shi et al. 2024). While effective, these approaches are inherently limited by high computational costs and the loss of important details, such as small objects, which in turn can result in hallucinations. Furthermore, Cambrian-1 (Tong et al. 2024), designed with a vision centric approach, introduces a novel paradigm for vision-language fusion through its combination of multiple vision encoders and spatial vision aggregator (SVA), but faces difficulties due to the lack of dynamic feature fusion and shortcomings in fine-grained perception. Specifically, visual features are inherently high-dimensional and uncertain, encompassing objects, scenes, attributes, and spatial relationships. A single vision encoder, or a limited set thereof, is insufficient to comprehensively and adaptively capture these diverse aspects. Sparse Mixture of Experts (MoE) (Fedus, Zoph, and Shazeer 2022) has emerged as an efficient solution that dynamically activates only a subset of expert networks for each input to improve scalability and specialization. Although MoE has been widely adopted in state-of-the-art open-source LLMs, its potential remains largely underexplored in the design of connectors and the broader architecture of VLLMs.

To address the above limitations of existing vision encoders and enhance fine-grained visual understanding, we propose ***MoCHA***, a novel framework integrates multiple vision backbones under a sparse ***M***ixture ***o***f Experts ***C***onnector (MoECs) module and a ***H***ierarchical Group ***A***ttention (HGA) component for vision-extensive reasoning tasks. Specifically, MoCHA consists of multiple vision encoders connected by a mixture of experts module and hierarchical group attention, as illustrated in Figure 1. First, we leverage four distinct yet complementary vision backbones, including CLIP (Radford et al. 2021), SigLIP (Zhai et al. 2023), DINOv2 (Oquab et al. 2024), and ConvNeXt (Liu et al. 2022), to extract diverse and robust image features. However, the efficient integration of these heterogeneous visual signals into a unified vision-language

*Corresponding author.

framework remains a challenge. To address this, we propose the mixture of experts connectors (MoECs) module, in which each connector is implemented as a Top-K sparsely-gated MoE for dynamic expert selection across visual dimensions. This facilitates efficient cross-modal interaction and reduces training costs. Sequential concatenation of features from multiple vision encoders along the token dimension often leads to redundancy and feature overlap. We further design hierarchical group attention (HGA), which fuses features through intra- and inter-group attention. By adaptive gating mechanism, we dynamically balance the aggregated and original features to produce the image representation without additional parameters. Our MoCHA is fully trained on open-source datasets and demonstrates superior performance compared to other VLLMs of similar size. In detail, we implement MoCHA with a two-stage training process on Phi2-2.7B and Vicuna-7B-v1.5 LLMs, achieving higher scores than existing open-weight VLLMs on mainstream vision-language benchmarks. Notably, our 3B-sized MoCHA shows a 3.25% reduction in hallucination on POPE and delivers a 153-point improvement on general visual tasks on MME, outperforming the larger CuMo-7B model (Li et al. 2024a). This demonstrates the efficacy of our MoCHA in mitigating VLM hallucinations and enhancing general perceptual abilities. We further conduct ablation studies to investigate the contribution of each component to overall performance.

Related Work

Large Pre-trained Vision Models

The advent of pre-trained Vision Transformers (ViT) (Dosovitskiy et al. 2021) has significantly propelled the advancement of computer vision. As the original CLIP adopted by conventional visual instruction tuning approaches is trained on noisy image-text pairs, it exhibits specific visual shortcomings, and thus stronger backbones have been introduced to LLMs. For example, SigLIP introduced pairwise sigmoid loss during training, enabling the vision encoder to demonstrate more advanced visual perception capabilities. Some concurrent works have leveraged external assistance from vision-only, self-supervised models such as DINOv2, MoCo-v3 (He et al. 2022), and other advanced frameworks (Chen et al. 2024), as well as domain-specific expert models. Additionally, ConvNeXt contributes to this progress by incorporating a purely convolutional backbone that achieves Transformer-level performance with improved computational efficiency. Collectively, these approaches foster robust multimodal integration and significantly enhance visual reasoning capabilities in contemporary VLLM frameworks by leveraging complementary vision encoders to enrich visual representations and improve fine-grained perception.

Mixture of Experts

Mixture of Experts (MoE) proposes a group of expert networks to handle specific tasks, with a gating network selecting the appropriate experts. Subsequent research on language MoE models has further expanded large MoE-based language models, improving expert stability and load balancing. ST-MoE (Zoph et al. 2022) employs load-balancing loss and router-z loss to ensure an even distribution of experts. The recently popular DeepSeek-V3 adopts the DeepSeekMoE architecture, leveraging shared experts to capture common knowledge and reduce redundancy in routed experts. MoE's success has also extended to the vision domain. LIMoE (Mustafa et al. 2022) replaces dense MLP layers with MoE layers in CLIP, improving zero-shot image classification. AdaMV-MoE (Liu and Luo 2024) introduces an adaptive MoE framework for multi-task learning. Furthermore, CuMo (Li et al. 2024a) integrates co-upcycled Top-K sparse-gating MoE blocks into the visual encoder, MLP connector, and language model, enhancing VLLM inference performance.

Methodology

Overview

To address a critical gap where current VLLMs still struggle to solve vision-extensive multimodal tasks, we present our framework, MoCHA, in this section. Building upon the mainstream LLaVA architecture, MoCHA moves beyond the original CLIP encoder by integrating a diverse set of vision encoders, spanning different architectures and pre-training objectives. This enables the model to effectively capture and process visual information across varying levels of granularity. As illustrated in Figure 1, our framework consists of multiple vision encoders (i.e., OpenAI CLIP ViT-L/14@336, SigLIP ViT-L/16@384, OpenCLIP ConvNeXt-XXL@1024 and DINOv2 ViT-L/14@336) connected by a mixture of experts module and hierarchical group attention.

Selection of Vision Encoders

Although language-supervised models outperform self-supervised and other models across all benchmark categories, Cambrian-1 (Tong et al. 2024) highlights that well-trained self-supervised models like DINOv2 are capable of achieving competitive performance in vision-centric tasks. In particular, high-resolution models excel in chart-related and other vision-centric benchmarks while also demonstrating robust performance across both general VQA and knowledge-based VQA. While ViTs remain the dominant architecture, ConvNet-based models such as OpenCLIP ConvNeXt are well-suited for high-resolution image processing (Vishniakov, Shen, and Liu 2024) delivering outstanding results on OCR, chart interpretation, and other vision-centric benchmarks.

Each vision encoder excels in different aspects of VLLM performance. More analysis on the vision encoders can be found in Appendix A of the extended version. SigLIP consistently excels in vision tasks due to its strong cross-modal semantic understanding and stability, and is therefore used as the baseline vision encoder. In this study, we explore the potential of combining CLIP, SigLIP, ConvNeXt, and DINOv2 to leverage their distinctive representations, aiming to enhance the exploitation of heterogeneous visual features.

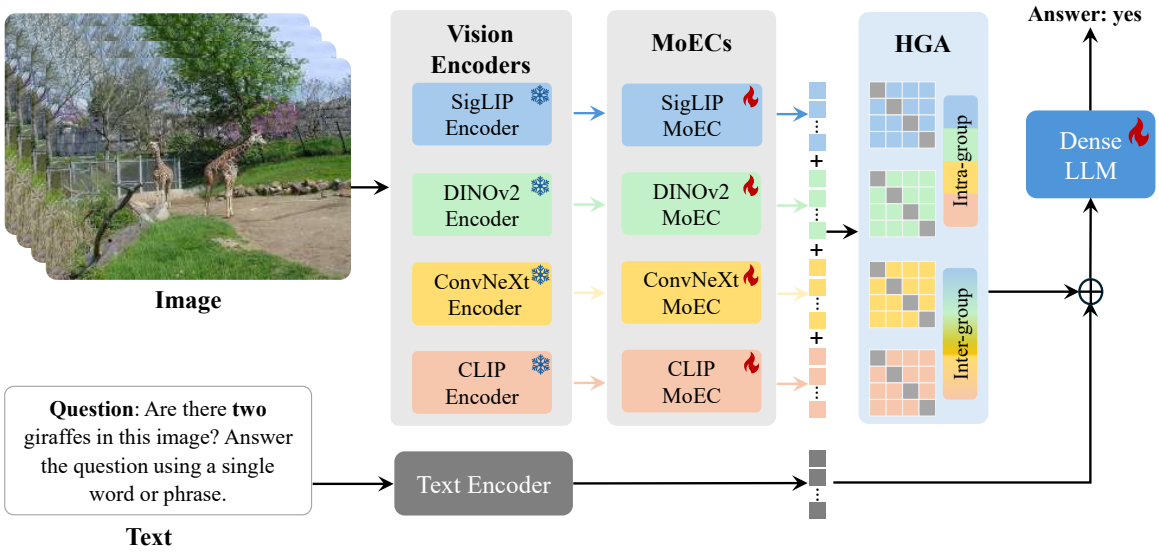

Figure 1: Architecture of MoCHA. MoCHA integrates multiple vision encoders (i.e., CLIP, SigLIP, ConvNeXt, and DINOv2), and consists of a Mixture of Experts Connectors (MoECs) module and Hierarchical Group Attention (HGA) module.

Complementary Visual Feature Extraction

To address the challenge of varying input resolutions across different vision encoders, a common approach is to employ interpolation for aligning feature dimensions. However, this process can result in information loss. To mitigate this, we opt to forgo interpolation and instead ensure that the output features of all encoders share consistent channel dimensions. In our experiments, we apply pooling to the features produced by ConvNeXt-XXL@1024 to standardize the channel dimensions, while maintaining the configurations of the other encoders unchanged. Each vision encoder processes the image I independently:

$$\begin{aligned} \textbf{SigLIP}(I) &= e_s \in \mathbb{R}^{s \times D_s}, \\ \textbf{DINOv2}(I) &= e_d \in \mathbb{R}^{d \times D_d}, \\ \textbf{ConvNeXt}(I) &= e_n \in \mathbb{R}^{n \times D_n}, \\ \textbf{CLIP}(I) &= e_c \in \mathbb{R}^{c \times D_c}, \end{aligned} \tag{1}$$

here, e_s, e_d, e_n and e_c represent the embeddings of image I generated by the four vision encoders. s, d, n and c denote the number of tokens for each encoder, and D_s, D_d, D_n and D_c represents the dimension of the feature channel.

Mixture of Experts Connectors (MoECs)

The output of LLMs is typically semantically uniform and exhibits minimal variation across dimensions, which renders specialized experts unnecessary. In contrast, vision models generate diverse information across multiple dimensions, such as objects, scenes, attributes, and spatial relationships, introducing greater uncertainty and variability. This complexity makes them well-suited for processing by multiple experts, each specializing in different visual aspects.

As illustrated in Figure 2, we introduce an MoE into the connector of VLLM. We refer to this design as the Mixture of Experts Connector (MoEC). MoEC preserves the stability and generality of vision-language architectures while effectively capturing the multi-dimensional and uncertain nature of visual information. MoEC retains the core functionality of the connector by projecting visual tokens into the word embedding space and aligning them with textual representations.

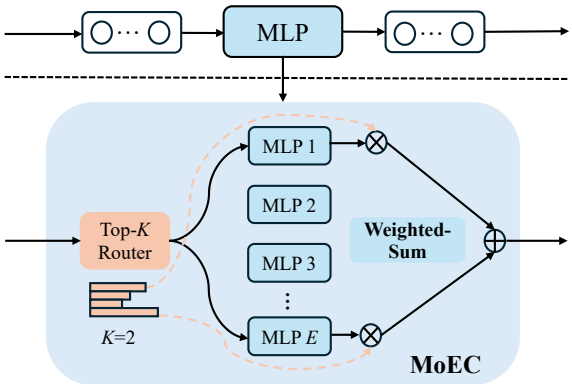

Figure 2: Sparse Top-K MoEC block. Replacing MLP connector with MoEC for visual-text token alignment.

By leveraging the connector's small scale, flexibility and focus on intermodal fusion, MoEC dynamically selects expert strategies tailored to different visual dimensions, thereby enhancing the efficiency of modality interaction and reducing training complexity.

Previous mainstream approaches replace dense MLP blocks with sparsely-gated mixture of experts blocks. As an example, consider the SigLIP vision encoder embedding. Given input $e_s \in \mathbb{R}^{s \times D_s}$ and an MLP block, the hidden rep-

resentation is computed as:

$$e_{s,h} = \text{MLP}(e_s) \in \mathbb{R}^{s \times D}, \tag{2}$$

To scale up the model with multiple MLP blocks in parallel, a sparse MoE block incorporates a router network that selects the Top-K experts from a total of E experts, as illustrated in Figure 2. The router network uses a linear layer to compute a normalized weight matrix based on the inputs e_s, enabling expert selection:

$$W_s = \text{Softmax}(\text{Linear}(e_s)) \in \mathbb{R}^{s \times E}, \tag{3}$$

Based on W_s, the Top-K experts are selected for each token, and the corresponding weights are re-normalized as:

$$W_{s,K} = \text{Softmax}(\text{Top}K(W_s)) \in \mathbb{R}^{s \times K}, \tag{4}$$

Each selected expert corresponds to an MLP block, and the final hidden representation is obtained through a re-weighted sum:

$$e_{s,h} = \sum_{i}^{K} W_{s,K}^{i} \circ \text{MLP}_i(e_s) \in \mathbb{R}^{s \times D}. \tag{5}$$

The hidden representation of SigLIP MoEC maintains the same dimensionality as that of a single dense MLP block. Similarly, the hidden representations for DINOv2 MoEC, ConvNeXt MoEC, and CLIP MoEC are obtained in the same way: $e_{d,h} \in \mathbb{R}^{d \times D}$, $e_{n,h} \in \mathbb{R}^{n \times D}$, $e_{c,h} \in \mathbb{R}^{c \times D}$.

Sequence Append (Token Dimension): While channel concatenation (Lin et al. 2023) achieves performance comparable to sequence append (Fan et al. 2024), it requires strict spatial alignment of encoder outputs, rendering it less compatible with encoders of differing resolutions and architectures. Sequence append along the token dimension obviates token count alignment and removes the requirement for interpolation, resampling, or padding. As a result, encoders with different resolutions, patch sizes, or feature map dimensions can contribute arbitrary numbers of tokens. In this study, we adopt sequence append, concatenating features from different encoders directly along the token dimension:

$$X_{\text{in}} = (\text{Concatenate}[e_{s,h}, e_{d,h}, e_{n,h}, e_{c,h}],\ \text{dim=token}) \in \mathbb{R}^{N \times D}, \tag{6}$$

where $s + d + n + c = N$.

Hierarchical Group Attention

While sequence append is straightforward to implement, it has limitations, particularly the lack of selective attention to tokens, as all tokens are processed equally, which may introduce redundancy. To address the above challenge, we propose a hierarchical group attention (HGA) that enables adaptive feature fusion via intra- and inter-group attention, allowing the model to selectively focus on visual tokens.

As illustrated in Figure 3, output tokens from the four vision encoders (i.e., SigLIP, DINOv2, ConvNeXt and CLIP) are regarded as independent feature groups. The intra-group attention mechanism is used to select the Top-M most

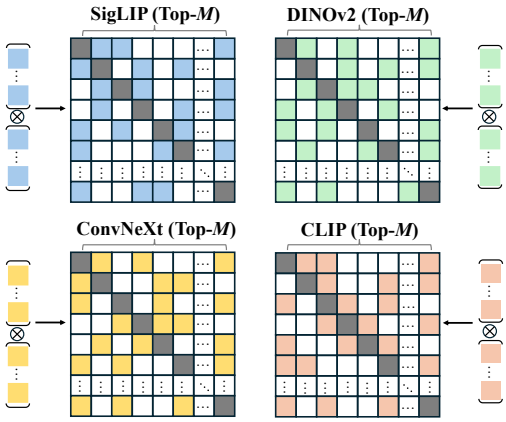

Figure 3: Intra-group Attention. Each model selects and aggregates the Top-M features for its tokens within the same group.

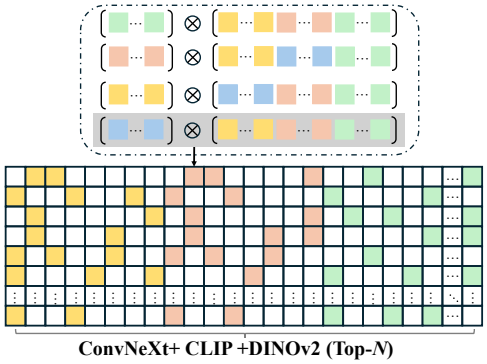

Figure 4: Inter-group Attention. SigLIP selects and aggregates the Top-N features for each token across different models (groups).

salient token features within each group by computing pairwise similarity scores followed by a self-masking operation:

$$I_{\text{intra},s,M} = \text{Top}M(\text{sim}(e_{s,h}, e_{s,h}) \odot (1 - I)). \tag{7}$$

Inter-group attention captures semantic correlations across encoders in high-dimensional space, rather than simple feature redundancy. As illustrated in Figure 4, it extracts complementary Top-N token information from different encoder features:

$$I_{\text{inter},s,N} = \text{Top}N(\text{sim}(e_{s,h}, [e_{n,h}, e_{c,h}, e_{d,h}])). \tag{8}$$

Hierarchical information from both intra- and inter-group attention is integrated into the aggregated feature representation $X_{\text{in,agg}}$ for each token. An adaptive gating mechanism is then applied to balance the contributions of $X_{\text{in,agg}}$ and the original features X_{in}:

$$\begin{aligned} \text{gate} &= \text{Sigmoid}[10 * (X_{\text{in,agg}} - X_{\text{in}} - 0.2)], \\ X_{\text{out}} &= (1 - \text{gate}) * X_{\text{in}} + \text{gate} * X_{\text{in,agg}}. \end{aligned} \tag{9}$$

Loss Function

To maintain a load balance among experts in each MoEC block, we adopt auxiliary losses based on the language

modeling cross-entropy loss. This auxiliary loss consists of a load balancing loss (Wang et al. 2024) and a router z-loss (Zoph et al. 2022). Hence, the total loss is:

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{ce} + \alpha_b(\mathcal{L}_{b,s} + \mathcal{L}_{b,d} + \mathcal{L}_{b,n} + \mathcal{L}_{b,c}) \\ +\alpha_z(\mathcal{L}_{z,s} + \mathcal{L}_{z,d} + \mathcal{L}_{z,n} + \mathcal{L}_{z,c}),\end{aligned} \tag{10}$$

where $\mathcal{L}_{ce}$ represents the autoregressive language modeling loss, which computes the cross-entropy of next-token predictions. α_b and α_z denote coefficients for loading balance loss (i.e., $\mathcal{L}_{b,s}$, $\mathcal{L}_{b,d}$, $\mathcal{L}_{b,n}$, $\mathcal{L}_{b,c}$) and router z-loss (i.e., $\mathcal{L}_{z,s}$, $\mathcal{L}_{z,d}$, $\mathcal{L}_{z,n}$, $\mathcal{L}_{z,c}$), set to 0.1 and 0.01, respectively, across all experiments. These auxiliary losses are applied within the MoECs.

Experiments

Data Details

Training Data. The training process utilizes a 558K subset of the LAION-CC-SBU dataset with BLIP-generated captions, followed by fine-tuning on a 665K mixture of instruction-following data from LLaVA-v1.5.

Evaluation Benchmarks. We focus on academic VQA datasets, including GQA (Hudson and Manning 2019), Science-QA (Lu et al. 2022), and TextVQA (Singh et al. 2019), as well as instruction-following VLLM benchmarks, such as POPE (Li et al. 2023), MME (Fu et al. 2024), MM-Bench (Liu et al. 2024c), and MM-Vet (Yu et al. 2024). Furthermore, we evaluate the challenging MathVista (Lu et al. 2024) dataset to assess VLLM visual reasoning capabilities.

Implementation Details

Two-Stage Training. To enhance training stability, we adopt a two-stage training strategy. More training details and hyperparameters can be found in Appendix B.

Parameter Settings. We train for one epoch using 8 NVIDIA L40 (48GB) GPUs. During the pre-training stage, the learning rate is set to 1e-3, with a batch size of 8×16 (1 vision encoder), 8×8 (2 vision encoders) and 8×4 (3 or 4 vision encoders). During visual instruction fine-tuning, the learning rate is reduced to 2e-5, with a batch size of 8×8 (1 vision encoder) and 8×4 or 2 (2, 3, or 4 vision encoders). Each MoEC has $E = 4$ experts, with $K = 2$ activated per forward pass. For adaptive feature fusion through intra- and inter-group attention, we aggregate $M = 3$ tokens within each group and $N = 7$ tokens across groups.

Training Settings. We employ pre-trained OpenAI CLIP ViT-L/14@336, SigLIP ViT-L/16@384, OpenCLIP ConvNeXt-XXL@1024, and DINOv2 ViT-L/14@336 as vision encoders. The vision-language connector is implemented as an MoEC, which consists of a router and expert networks. Each expert consists of two linear layers. For the language model, we utilize Phi2-2.7B and Vicuna-7B-v1.5.

Evaluation Settings. We follow the LLaVA series settings, using greedy decoding for all benchmarks. Data and questions are converted into visual instructions to prompt VLLMs. For GPT API-based evaluation, we use GPT-3.5-turbo for MathVista.

Main Results

We scale the LLM backbone from 2.7B to 7B parameters for both pre-training and fine-tuning, achieving competitive performance. Comparative analysis is conducted against other instruction-tuned VLLMs, categorized by backbone size (e.g., 2.7B, 7B, 8B and 13B).

MoCHA Stands Out Among Other VLLMs. In Table 1, we evaluate MoCHA on 8 multimodal benchmarks spanning from general (e.g., GQA, MMVet) to math and reasoning multimodal tasks (e.g., SQA IMG, MathVista). Under the same LLM and fine-tuning settings, MoCHA (Phi2-2.7B) consistently outperforms LLaVA-Phi and MoE-LLaVA (Phi2-2.7B) across all benchmarks. Moreover, MoCHA (Phi2-2.7B) achieves overall performance comparable to, and in some cases surpassing, many 7B-based VLLMs as well as larger models. Specifically, MoCHA (Phi2-2.7B) outperforms SPHINX-Plus (LLaMA2-13B) by 0.85% on POPE and 243.9 points on MME. These findings indicate that smaller models can exhibit strong visual perception and excel at fine-grained detail recognition.

As we scale up the LLM backbone, MoCHA (Vicuna-7B) demonstrates stronger visual reasoning capabilities than MoCHA (Phi-2 2.7B). Notably, MoCHA (Vicuna-7B) also surpasses models with even larger LLMs across multiple visual benchmarks. For instance, compared to LLaVA-v1.5 (Vicuna-13B), it achieves improvements of 2.99%, 4.42%, and 4.08% on SQA IMG, TextVQA, and POPE, respectively. Moreover, in contrast to CuMo—which applies a global mixture of experts (MoE) across the LLM, vision encoder, and connector—MoCHA (Vicuna-7B) outperforms CuMo (Mistral-7B) on several key vision benchmarks. These results provide strong evidence that, compared to the conventional single-vision-encoder paradigm, MoCHA's dynamic MoE design and customized attention mechanisms offer a more effective approach for multimodal learning (ML). We present additional examples comparing LLaVA and MoCHA on visual reasoning, such as multi-object recognition, in Appendix D.

MoCHA Helps Reduce Visual Hallucination. We follow the evaluation protocol of POPE, a popular benchmarking for evaluating visual object hallucination. MoCHA shows a marked improvement in mitigating hallucination issues in Table 1, outperforming models with larger parameter sizes. It produces object descriptions that are highly consistent with the input images and exhibits a rich potential for hallucination inhibition. Specifically, we observe that MoCHA surpasses LLaVA in popular sampling, adversarial sampling, and random sampling, despite having fewer parameters. Additionally, the yes ratio of MoCHA remains relatively balanced, which indicates that our framework is capable of providing accurate feedback based on the given questions.

The MoCHA Design Optimizes Model Efficiency. To validate the efficiency use of our MoECs, we further present training GFLOPs, parameters, and average inference time of models in Table 2. We make sure that all numbers are obtained under identical configurations to ensure the validity of the comparison. From the table, MoCHA achieves

Method	LLM	GQA	SQA IMG	Text VQA	MM Vet	POPE	MME	MMB EN	MathVista
Qwen-VL-Chat (Bai et al. 2023)	Qwen-7B	57.5	68.2	61.5	-	-	1487.5	60.6	-
InstructBLIP (Dai et al. 2023)	Vicuna-7B	49.2	60.5	50.1	26.2	-	-	36.0	-
LLaVA-LLaMA3 (Contributors 2023)	LLaMA3-8B-IT	62.6	72.9	59.0	-	86.4	1469	72.3	-
Mini-Gemini (Li et al. 2024b)	Vicuna-7B	-	65.2	-	40.8	-	1523	69.3	31.4
SPHINX-Intern2 (Liu et al. 2024a)	InternLM2-7B	56.2	70.4	58.1	36.5	86.9	1260.4	57.9	35.5
LLaVA-NeXT (Liu et al. 2024b)	Vicuna-7B	64.2	70.1	64.9	43.9	86.5	1519	67.4	34.6
LLaVA-NeXT (Liu et al. 2024b)	Mistral-7B	64.8	72.8	65.7	47.3	86.7	1498	68.7	**37.7**
LLaVA-v1.5 (Liu et al. 2023)	Vicuna-7B	62.0	66.8	58.2	30.5	85.9	1510.7	64.3	-
VILA (Lin et al. 2024b)	Vicuna-7B	62.3	68.2	64.4	34.9	85.5	1533	68.9	-
CuMo (Li et al. 2024a)	Mistral-7B	64.9	73.9	**67.0**	**51.0**	86.7	1548.6	**73.0**	35.1
InstructBLIP (Dai et al. 2023)	Vicuna-13B	49.5	63.1	50.7	25.6	78.9	1212.8	-	-
LLaVA-v1.5 (Liu et al. 2023)	Vicuna-13B	63.3	71.6	61.3	35.4	85.9	1531.3	67.7	27.6
Mini-Gemini (Li et al. 2024b)	Vicuna-13B	-	65.9	-	46.0	-	1565	68.5	37.0
InternVL-Chat (Chen et al. 2024)	Vicuna-13B	**66.6**	-	61.5	-	87.6	1586.4	-	-
LLaMA-VID (Li, Wang, and Jia 2024)	Vicuna-13B	65.0	70.0	-	-	86.0	1542.3	66.6	-
SPHINX-Plus (Liu et al. 2024a)	LLaMA2-13B	-	74.2	65.7	47.9	89.1	1457.7	71.0	36.8
MobileVLM (Chu et al. 2023)	MobileLLaMA-2.7B	59.0	61.0	47.5	-	84.9	1288.9	59.6	-
TinyGPT-V (Yuan et al. 2024)	Phi2-2.7B	33.6	-	-	-	-	-	-	-
LLaVA-Phi (Zhu et al. 2024)	Phi2-2.7B	-	68.4	48.6	28.9	85.0	1335.1	59.8	-
MoE-LLaVA-2.7B×4-Top2 (Lin et al. 2024a)	Phi2-2.7B	61.4	68.5	51.4	34.3	86.3	1423	65.2	-
MoCHA (SigLIP+DINOv2+ConvNeXt+CLIP)	Phi2-2.7B	65.24	73.24	63.13	32.74	89.95	1701.60	69.67	32.24†
MoCHA (SigLIP+DINOv2+ConvNeXt+CLIP)	Vicuna-7B	65.73	**74.59**	65.72	36.68	**89.98**	**1744.09**	71.06	35.60†

Table 1: Comparisons between MoCHA and other VLLMs on competitive benchmarks. These models are grouped by the size of the base LLM. Numbers† are averaged by three inference runs of querying GPT API.

Model	LLM	Avg. Infer. Time	Total Params	Trainable Params	GFLOPs
LLaVA-v1.5	Vicuna-13B	0.81s	13.35B	13.05B	16894.72
LLaVA-v1.5	Vicuna-7B	0.44s	7.06B	6.76B	8877.05
InternVL-Chat	Vicuna-13B	0.83s	18.96B	13.06B	23334.93
MoE-LLaVA	Phi2-2.7B	**0.41s**	5.61B	5.30B	**7338.97**
MoCHA	Phi2-2.7B	0.57s	**4.97B**	**4.07B**	12014.64

Table 2: Quantitative comparison of MoCHA and other VLLMs.

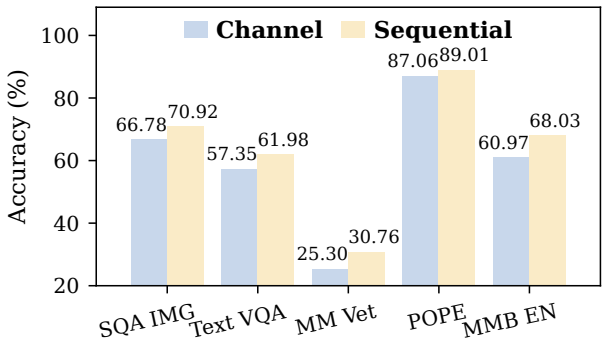

Figure 5: Ablation of Channel vs. Sequential Concatenation for MoECs with four encoders.

an inference speed of 0.57s with only 4.97B parameters and 12014.64 GFLOPs. This not only significantly reduces both the parameter size and computational cost, but also outperforms LLaVA-v1.5 (Vicuna-13B) and InternVL-Chat (Vicuna-13B) in inference efficiency. Furthermore, it approaches the inference speed of MoE-LLaVA, despite using even fewer parameters. These notable advantages in both computational and parameter efficiency highlight the strong potential of scaling up MoCHA as a compelling alternative to existing vision encoder paradigms in ML.

Ablation Studies

Effect of Channel v.s. Sequential Concatenation. We conduct ablation experiments on the heterogeneous features dynamically selected by the MoECs module with sequential concatenation and channel concatenation to evaluate the differences between the two strategies. As illustrated in Figure 5, sequence concatenation consistently outperforms channel concatenation across all vision tasks under our MoEC settings. Furthermore, channel concatenation aggregates features from multiple encoders within each visual token; if one encoder (e.g., CLIP) has dominant feature magnitudes, it may overshadow fine-grained information from other encoders, leading the model to focus disproportionately on its features.

Effect of Multiple Vision Encoders. To investigate the use of different combinations of vision encoders under the MLP connector, ablation studies were conducted by individually extracting features from each vision encoder. As shown in Table 3, our study indicates that only using the SigLIP exhibits inferior performance across various bench-

Method (MLP)	GQA	SQA IMG	POPE	MMB EN
SigLIP	59.00	64.35	84.64	59.88
SigLIP+ConvNeXt	60.78	67.18	86.67	64.95
SigLIP+CLIP+ConvNeXt	57.21	65.39	85.72	58.42
SigLIP+DINOv2+ConvNeXt+CLIP	**61.86**	**69.01**	**88.47**	**66.51**

Table 3: Ablation study only on different vision encoder combinations.

Method (MoEC)	GQA	SQA IMG	POPE	MMB EN
SigLIP	56.77	64.60	82.85	57.65
SigLIP+ConvNeXt	57.77	67.92	86.72	63.49
SigLIP+CLIP+ConvNeXt	59.10	66.58	87.06	60.83
SigLIP+DINOv2+ConvNeXt+CLIP	**63.35**	**70.92**	**89.01**	**68.03**

Table 4: Ablation study of different vision encoder combinations under MoEC settings.

Method (MoEC + HGA)	GQA	SQA IMG	POPE	MMB EN
SigLIP	56.11	62.76	82.27	57.90
SigLIP+ConvNeXt	57.02	63.97	87.84	61.06
SigLIP+CLIP+ConvNeXt	60.34	68.73	88.07	62.11
SigLIP+DINOv2+ConvNeXt+CLIP	**65.24**	**73.24**	**89.95**	**69.67**

Table 5: Ablation Study of MoCHA with different vision encoder combinations. Settings for results in Table 1 are highlighted in blue.

Top-K	GQA	SQA IMG	POPE	MMB EN
1	59.86	68.13	85.93	65.59
2	**65.24**	**73.24**	**89.95**	**69.67**
3	62.90	70.18	88.21	67.35
4	58.32	67.64	86.02	64.97

Table 6: Ablation study on the value of Top-K in MoCHA.

marks. The inclusion of ConvNeXt improves performance on benchmarks such as GQA, SQA IMG, and POPE. However, the subsequent addition of CLIP leads to a performance decline. This phenomenon can be attributed to the substantial feature similarity and redundancy between SigLIP and CLIP, since both encoders are trained with large-scale image-text contrastive objectives. The configuration that integrates SigLIP, DINOv2, ConvNeXt, and CLIP achieves the best overall performance. In particular, DINOv2 provides a balanced contribution by capturing both local details and global semantic information.

Upgrade MLP connector to MoEC. To evaluate MoEC's capability in efficiently integrating heterogeneous and multi-dimensional features, we initiate this ablation study by replacing each MLP connector with an upcycled MoEC, as shown in Table 4. The results in Table 3 indicate that with the plain MLP as the connector, the model performance with a single SigLIP remains weak. While the addition of ConvNeXt further improves performance, the full integration of CLIP yields the largest gain among all settings. On the other hand, in Table 4, the combination of SigLIP, DINOv2, ConvNeXt, and CLIP under our MoEC results in notable performance improvements of 1.49%, 1.91%, 0.54%, and 1.52% on the GQA, SQA IMG, POPE, and MMBench benchmarks, respectively, compared to results using vanilla MLP in Table 3. These findings show that MoECs effectively handle diverse visual features, including object recognition, scene understanding, attribute identification, and spatial reasoning. Even as the number of vision encoders grows, MoECs incur minimal parameter overhead while keeping inference time and computational cost on par with a standard MLP. Further quantitative results can be found in Appendix C.

Effect of Hierarchical Group Attention (HGA). We further ablate our hierarchical group attention (HGA) to evaluate its effectiveness in feature fusion. As shown in Table 5, continual performance gains are observed as ConvNeXt, CLIP, and DINOv2 are introduced sequentially. This confirms that the proposed intra- and inter-group adaptive attention fusion in HGA enhances the synergy and complementarity among multiple vision encoders. Compared to previous results, the performance of the single SigLIP encoder slightly drops relative to both MoEC (Table 4) and the standard MLP baseline (Table 3), suggesting that intra-group feature aggregation within a single encoder may introduce redundancy. Notably, incorporating ConvNeXt leads to a 5.57% improvement on the POPE benchmark, indicating that HGA effectively leverages ConvNeXt's fine-grained local features. The full combination of SigLIP, DINOv2, ConvNeXt, and CLIP yields the best overall performance, further demonstrating the robustness of MoCHA.

Effect of the Number of Activated Experts. To evaluate the effect of the number of activated experts, we compare different Top-K strategies. As shown in Table 6, $K = 2$ achieves the best performance, while larger K values provides limited improvement, increases computational and memory costs, lowers parameter utilization, and may cause expert imbalance. Hardware constraints further restrict $K = 4$ to a batch size of 1, which may degrade performance. Accordingly, we set $K = 2$ to fully exploit the advantages of the MoEC architecture.

Conclusion

In this work, we introduce MoCHA, a novel framework for efficiently training vision-language models with advanced visual reasoning capabilities. It addresses two key challenges in visual detail extraction and in heterogeneous feature fusion. The MoECs module enables dynamic expert selection, effectively integrating diverse visual signals within a unified vision-language framework. Customized HGA further enhances the complementarity of heterogeneous features. Our framework can be readily adapted to incorporate additional vision backbones and larger language models.

References

Bai, J.; Bai, S.; Yang, S.; Wang, S.; Tan, S.; Wang, P.; Lin, J.; Zhou, C.; and Zhou, J. 2023. Qwen-VL: A Versatile Vision-Language Model for Understanding, Localization, Text Reading, and Beyond. arXiv:2308.12966.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; Zhong, H.; Zhu, Y.; Yang, M.; Li, Z.; Wan, J.; Wang, P.; Ding, W.; Fu, Z.; Xu, Y.; Ye, J.; Zhang, X.; Xie, T.; Cheng, Z.; Zhang, H.; Yang, Z.; Xu, H.; and Lin, J. 2025. Qwen2.5-VL Technical Report. arXiv:2502.13923.

Chen, Z.; Wu, J.; Wang, W.; Su, W.; Chen, G.; Xing, S.; Zhong, M.; Zhang, Q.; Zhu, X.; Lu, L.; et al. 2024. 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.

Chu, X.; Qiao, L.; Lin, X.; Xu, S.; Yang, Y.; Hu, Y.; Wei, F.; Zhang, X.; Zhang, B.; Wei, X.; and Shen, C. 2023. MobileVLM : A Fast, Strong and Open Vision Language Assistant for Mobile Devices. arXiv:2312.16886.

Contributors, X. 2023. Xtuner: a toolkit for efficiently finetuning LLM. https://github.com/InternLM/xtuner.

Dai, W.; Li, J.; Li, D.; Tiong, A. M. H.; Zhao, J.; Wang, W.; Li, B.; Fung, P.; and Hoi, S. 2023. InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Fan, X.; Ji, T.; Jiang, C.; Li, S.; Jin, S.; Song, S.; Wang, J.; Hong, B.; Chen, L.; Zheng, G.; Zhang, M.; Huang, C.; Zheng, R.; Xi, Z.; Zhou, Y.; Dou, S.; Ye, J.; Yan, H.; Gui, T.; Zhang, Q.; Qiu, X.; Huang, X.; Wu, Z.; and Jiang, Y.-G. 2024. MouSi: Poly-Visual-Expert Vision-Language Models. arXiv:2401.17221.

Fedus, W.; Zoph, B.; and Shazeer, N. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120): 1–39.

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.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked Autoencoders Are Scalable Vision Learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, 15979–15988. IEEE.

Hudson, D. A.; and Manning, C. D. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 6700–6709. Computer Vision Foundation / IEEE.

Li, J.; Wang, X.; Zhu, S.; Kuo, C.-W.; Xu, L.; Chen, F.; Jain, J.; Shi, H.; and Wen, L. 2024a. CuMo: Scaling Multimodal LLM with Co-Upcycled Mixture-of-Experts. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Li, Y.; Du, Y.; Zhou, K.; Wang, J.; Zhao, W. X.; and Wen, J.-R. 2023. Evaluating Object Hallucination in Large Vision-Language Models.

Li, Y.; Wang, C.; and Jia, J. 2024. Llama-vid: An image is worth 2 tokens in large language models. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XLVI*, volume 15104 of *Lecture Notes in Computer Science*, 323–340. Springer.

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.; Huang, J.; Zhang, J.; Pang, Y.; Jin, P.; Ning, M.; Luo, J.; and Yuan, L. 2024a. MoE-LLaVA: Mixture of Experts for Large Vision-Language Models. arXiv:2401.15947.

Lin, J.; Yin, H.; Ping, W.; Lu, Y.; Molchanov, P.; Tao, A.; Mao, H.; Kautz, J.; Shoeybi, M.; and Han, S. 2024b. VILA: On Pre-training for Visual Language Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, 26679–26689. IEEE.

Lin, Z.; Liu, C.; Zhang, R.; Gao, P.; Qiu, L.; Xiao, H.; Qiu, H.; Lin, C.; Shao, W.; Chen, K.; Han, J.; Huang, S.; Zhang, Y.; He, X.; Li, H.; and Qiao, Y. 2023. SPHINX: The Joint Mixing of Weights, Tasks, and Visual Embeddings for Multi-modal Large Language Models. arXiv:2311.07575.

Liu, D.; Zhang, R.; Qiu, L.; Huang, S.; Lin, W.; Zhao, S.; Geng, S.; Lin, Z.; Jin, P.; Zhang, K.; Shao, W.; Xu, C.; He, C.; He, J.; Shao, H.; Lu, P.; Li, H.; Qiao, Y.; and Gao, P. 2024a. SPHINX-X: Scaling Data and Parameters for a Family of Multi-modal Large Language Models. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

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. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36: 34892–34916.

Liu, Y.; Duan, H.; Zhang, Y.; Li, B.; Zhang, S.; Zhao, W.; Yuan, Y.; Wang, J.; He, C.; Liu, Z.; Chen, K.; and Lin,

D. 2024c. MMBench: Is Your Multi-modal Model an All-around Player?

Liu, Z.; and Luo, J. 2024. AdaMoLE: Fine-Tuning Large Language Models with Adaptive Mixture of Low-Rank Adaptation Experts. arXiv:2405.00361.

Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; and Xie, S. 2022. A ConvNet for the 2020s. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, 11966–11976. IEEE.

Lu, P.; Bansal, H.; Xia, T.; Liu, J.; Li, C.; Hajishirzi, H.; Cheng, H.; Chang, K.-W.; Galley, M.; and Gao, J. 2024. MathVista: Evaluating Mathematical Reasoning of Foundation Models in Visual Contexts.

Lu, P.; Mishra, S.; Xia, T.; Qiu, L.; Chang, K.-W.; Zhu, S.-C.; Tafjord, O.; Clark, P.; and Kalyan, A. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Mustafa, B.; Riquelme, C.; Puigcerver, J.; Jenatton, R.; and Houlsby, N. 2022. Multimodal Contrastive Learning with LIMoE: the Language-Image Mixture of Experts. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

OpenAI. 2023. ChatGPT. https://openai.com/blog/chatgpt/.

OpenAI. 2024. Hello gpt-4o. https://openai.com/index/hello-gpt-4o/.

Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F.; El-Nouby, A.; Assran, M.; Ballas, N.; Galuba, W.; Howes, R.; Huang, P.-Y.; Li, S.-W.; Misra, I.; Rabbat, M.; Sharma, V.; Synnaeve, G.; Xu, H.; Jegou, H.; Mairal, J.; Labatut, P.; Joulin, A.; and Bojanowski, P. 2024. DINOv2: Learning Robust Visual Features without Supervision. *Trans. Mach. Learn. Res.*, 2024.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, 8748–8763. PMLR.

Shi, B.; Wu, Z.; Mao, M.; Wang, X.; and Darrell, T. 2024. When Do We Not Need Larger Vision Models? In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part VIII*, volume 15066 of *Lecture Notes in Computer Science*, 444–462. Springer.

Singh, A.; Natarajan, V.; Shah, M.; Jiang, Y.; Chen, X.; Batra, D.; Parikh, D.; and Rohrbach, M. 2019. Towards vqa models that can read. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 8317–8326. Computer Vision Foundation / IEEE.

Tong, S.; Brown, E.; Wu, P.; Woo, S.; Middepogu, M.; Akula, S. C.; Yang, J.; Yang, S.; Iyer, A.; Pan, X.; Wang, Z.; Fergus, R.; LeCun, Y.; and Xie, S. 2024. Cambrian-1: A Fully Open, Vision-Centric Exploration of Multimodal LLMs. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

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 preprint arXiv:2302.13971*.

Vishniakov, K.; Shen, Z.; and Liu, Z. 2024. ConvNet vs Transformer, Supervised vs CLIP: Beyond ImageNet Accuracy. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Wang, L.; Gao, H.; Zhao, C.; Sun, X.; and Dai, D. 2024. Auxiliary-Loss-Free Load Balancing Strategy for Mixture-of-Experts. arXiv:2408.15664.

Yu, W.; Yang, Z.; Li, L.; Wang, J.; Lin, K.; Liu, Z.; Wang, X.; and Wang, L. 2024. MM-Vet: Evaluating Large Multimodal Models for Integrated Capabilities.

Yuan, Z.; Li, Z.; Huang, W.; Ye, Y.; and Sun, L. 2024. TinyGPT-V: Efficient Multimodal Large Language Model via Small Backbones. arXiv:2312.16862.

Yuksekgonul, M.; Bianchi, F.; Kalluri, P.; Jurafsky, D.; and Zou, J. 2023. When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Zhai, X.; Mustafa, B.; Kolesnikov, A.; and Beyer, L. 2023. Sigmoid Loss for Language Image Pre-Training. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, 11941–11952. IEEE.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E. P.; Zhang, H.; Gonzalez, J. E.; and Stoica, I. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Zhu, Y.; Zhu, M.; Liu, N.; Xu, Z.; and Peng, Y. 2024. Llava-phi: Efficient multi-modal assistant with small language model. In *Proceedings of the 1st International Workshop on Efficient Multimedia Computing under Limited*, 18–22.

Zoph, B.; Bello, I.; Kumar, S.; Du, N.; Huang, Y.; Dean, J.; Shazeer, N.; and Fedus, W. 2022. ST-MoE: Designing Stable and Transferable Sparse Expert Models. arXiv:2202.08906.