

MobileVLM V2: Faster and Stronger Baseline for Vision Language Model

Xiangxiang Chu^{1*}, Limeng Qiao^{1*}, Xinyu Zhang^{1*}, Shuang Xu¹, Fei Wei¹, Yang Yang^{1,3},
Xiaofei Sun¹, Yiming Hu¹, Xinyang Lin¹, Bo Zhang¹, Chunhua Shen²

¹Meituan Inc. ²Zhejiang University, China ³Dalian University of Technology, China

Abstract

We introduce *MobileVLM V2*, a family of significantly improved vision language models upon *MobileVLM*, which proves that a delicate orchestration of novel architectural design, an improved training scheme tailored for mobile VLMs, and rich high-quality dataset curation can substantially benefit VLMs’ performance. Specifically, *MobileVLM V2 1.7B* achieves better or on-par performance on standard VLM benchmarks compared with much larger VLMs at the 3B scale. Notably, our 3B model outperforms a large variety of VLMs at the 7B+ scale. Our models will be released at <https://github.com/Meituan-AutoML/MobileVLM>.

1. Introduction

To date, vision language models (VLMs) [4, 46, 58, 75] have been a crucial research focus of the community of artificial intelligence [71]. Integrating large language models (LLM) with multi-modality features is verified to have unprecedented versatile capability, demonstrated by proprietary models like Gemini [26], and GPT-4V [53] which have exhibited stunning performance on various downstream tasks [1, 26]. However, challenges remain to enable an vision language model as a capable intelligence assistant to be deployed to real scenarios such as mobile devices, self-driving cars, and embodied AI systems, *etc.*

Most recently, *MobileVLM* [15] comes first in exploring the capacity of VLMs at the mobile scale with innovative hardware-oriented architectures. *MoE-LLaVA* [41] manages to adapt the mixture-of-experts method [22, 30] for VLMs which outrageously pushed the limits of smaller models to yield surpassing performance compared with much larger ones. As the latest survey [71] indicates, the progress of VLMs exhibits a trend of extended modalities, refined training pipelines, efficient architectures, and higher-quality training datasets.

In this paper, we establish faster and stronger baselines

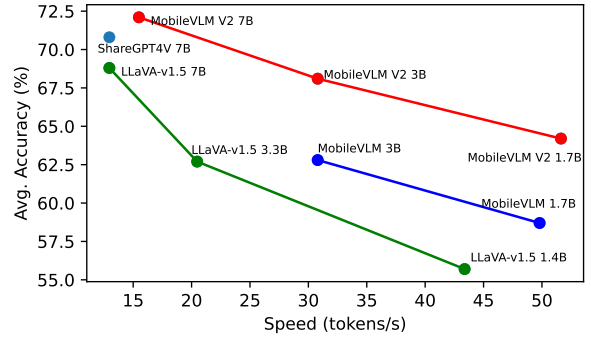


Figure 1. Comparison of SOTA VLMs in terms of average performance across several standard benchmarks and speed (tested on an NVIDIA Jetson Orin with `llama.cpp`). *MobileVLM V2* achieves new state-of-the-art results with much faster inference speed.

built upon *MobileVLM* [15]. Our key improvements are made on mainly three aspects, namely, exploiting contributive training data on small VLMs, exploring effective training strategies, and renovating a high-performance lightweight projector. Specifically, we utilize 1.2 million high-quality image-text pairs conducted by ShareGPT4V [10] to effectively align vision-language features, and incorporate more academic tasks to increase data diversity and instruction-following capacity, such as ScienceQA [49], TextVQA [60], SBU [54], *etc.*. As for the training paradigm, we conduct thorough training of all parameters of projector and language model during both the pretraining and instruction tuning stages, which proves advantageous in harnessing the full potential of superior-quality data. Additionally, we introduce a more streamlined yet potent projection mechanism that bridges vision and language models. By improving the representation of image tokens with enhanced positional information, we can significantly compress the number of image tokens without much performance degradation.

Our main contributions are summarized as follows:

1. We explore and evaluate the performance of increasing

*Equal contribution, sorted solely by the alphabetical order of surnames

training data for small vision language models, which significantly bridges the gap between small VLMs such as MobileVLM [15] and large ones.

2. We dive into better training strategies for mobile scenarios and design a novel training scheme on how to fully exploit the potential of more high-quality multimodal data. We propose a very lightweight projector to significantly reduce the visual tokens with slight performance drops.
3. Our method achieves a new state-of-the-art tradeoff between performance and inference speed across several vision language benchmarks. By scaling our model to 7B parameters, our method outperforms previous SOTA models with clear margins.

2. Related Work

Large Language Models. Over the last few years, the emergence of LLMs has led to significant advancement in natural language processing, including GPT-3 [6], PaLM [13], OPT [72], and BLOOM [59]. A representative work of democratic LLMs is LLaMA, which further enhances the enthusiasm for research on improving LLM effects. Following the representative works InstructGPT [55] and ChatGPT [51], Alpaca and Vicuna finetuned LLaMA with human-guided annotations to enhance the language interaction abilities. Recently, many organizations have proposed their large language models [3, 21, 52, 63, 69], which have been evaluated on multiple benchmarks and achieved excellent performance. Due to limited computational resources of edge devices such as mobile phones, autonomous driving cars, and embodied AI systems, several lite-LLMs [5, 15, 39, 56, 62] have attracted considerable attention. It is worth noting that TinyLLaMA [56] conducts great open-source work on their 1.1B base and chat models. MobileLLaMA [15] scales down the architecture of LLaMA and releases 1B and 3B models trained from scratch with public datasets. Meantime, model compression techniques including quantization and pruning on large language models is actively studied too [23, 24, 68]. Lately, the precision of LLMs is reduced to W4A8 [38] or even to binary [37] for faster inference with minimal accuracy drop.

Multimodal Large Language Models. Several works [36, 58, 61] have proposed a series of multimodal model architectures, which are primarily composed of the vision encoder and language model. [12, 66] explore visual language reasoning tasks under a unified transformer architecture. The paradigm of the vision-language model demonstrates better generalizability than traditional vision models designed for specific tasks. With the rapid development of LLMs, many works [4, 10, 46, 75] have concentrated on infusing visual knowledge into LLMs. FROMAGE [32] and LLaVA [46] directly feed visual tokens to LLMs, thereby

enabling the LLM to comprehend semantic information of images and correctly respond to queries for visual contents. InternLM-XComposer2 [20] proposes a Partial LoRA (PLoRA) approach to preserve the integrity of pre-trained language knowledge, which strikes a balance between precise vision understanding and text composition with literary talent. Its highly competitive performance shows remarkable proficiency in the realm of multimodal understanding. To address the urgent demand for deployment of MLLMs on edge devices, several works [15, 26, 41, 67, 76] have been proposed. Gemini [26], a leader in this field, has released lightweight vision-language models Gemini Nano with 1.8B/3.25B parameters, which are specifically designed for smartphones. However, their models and data are not open-sourced. MobileVLM [15] may be the first approach to offer open-source 1B/3B vision-language models under resource-constrained scenarios. MoE-LLaVA [41] proposes an MoE-based sparse model architecture to enhance visual understanding capabilities. LLaVA-Phi [76] utilizes Phi-2.7B as a language foundation model to achieve superior performance in vision language reasoning tasks. Vary [67] introduces an improved vision vocabulary and enhance the image features, leading to better generality. Notably, Vary [67] demonstrates superior potential in fine-grained perceptual tasks.

Multimodal Instruction-Tuning. To address the modality gap between vision and language, works such as [58] employ contrastive learning to align visual and textual feature representations. Furthermore, methods in [33, 57] leveraged the capabilities of large language models to achieve universal detection or segmentation with strong zero-shot capabilities. Inspired by the success in natural language processing, Liu et al. [46] introduced visual instruction tuning aiming at creating a general-purpose multimodal model with language as task instructions. SVIT [73] scales up visual instruction tuning by constructing a dataset of 4.2 million visual instruction tuning data, which is featured by the high quality and rich diversity generated by prompting GPT-4 with the abundant manual annotations of images. ShareGPT4V [10] constructs 100K high-quality captions from GPT4-Vision for instruction tuning, which have been expanded to 1.2 million detailed and informative captions for pre-training, covering world knowledge, object attributes, spatial relationships, and aesthetic assessments. LVIS-instruct4v [65] builds 220K visually aligned and context-aware instructions produced by prompting the powerful GPT-4V with images from LVIS.

3. Method

Our method follows a similar framework as MobileVLM [15]. As shown in Figure 2, the overall architecture of MobileVLM V2 consists of a pre-trained vision encoder to extract image features, a pre-trained large language model

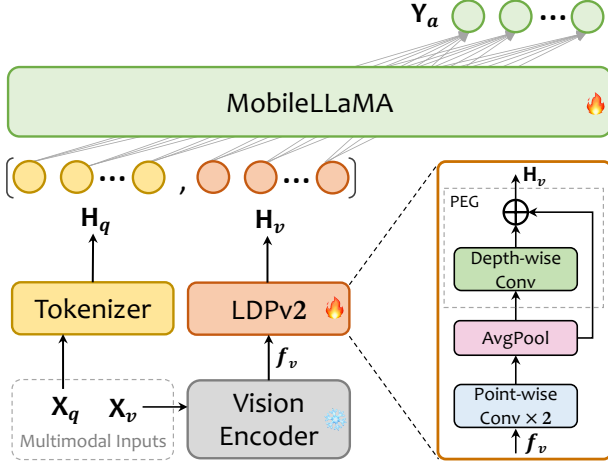


Figure 2. **MobileVLM V2’s architecture.** \mathbf{X}_v and \mathbf{X}_q indicate image and language instruction, respectively, and \mathbf{Y}_a refers to the text response from the language model MobileLLaMA. The diagram in the lower right corner is a detailed description of LDPv2, *i.e.*, the lightweight downsample projector v2.

MobileLLaMA to process multi-modal tokens and generate final answers, and a mobile-friendly projector, *i.e.*, lightweight downsample projector (denoted as LDPv2), to align image features with the language model. We introduce the details of each component in the following.

3.1. Vision Encoder

Following MobileVLM, we use CLIP ViT-L/14 [58] as the vision encoder \mathbf{F}_{enc} , which is contrastively pre-trained on millions of image-language pairs and proven to be effective for VLMs [15]. Particularly, images $\mathbf{X}_v \in \mathbb{R}^{H \times W \times C}$ are first resized to a resolution of 336×336 , and flattened into patches with a stride P of 14. High-level visual embeddings $f_v \in \mathbb{R}^{N_v \times D_v}$ are then extracted to represent the semantic information of images, where $N_v = HW/P^2$ and D_v denote the sequence length and the hidden size of visual embeddings. Formally we have,

$$f_v = \mathbf{F}_{enc}(\mathbf{X}_v). \quad (1)$$

3.2. Language Model

We employ the MobileLLaMA [15] series as the foundational large language model (LLM). This choice has three advantages. Firstly, MobileLLaMA is designed to facilitate off-the-shelf deployment and has demonstrated real-time speed on resource-limited devices with impressive performance. Keeping the language model unchanged helps us conduct controlled experiments to explore many other aspects such as scaling up the data corpus, improving training strategy, renovating new designs for projectors, and so on. Secondly, MobileLLaMA shares the same tokenizer with

LLaMA2, which helps to perform distillation without any pain. Last but not least, it is trained on open datasets and has no risk of evaluation pollution from data leakage. This also helps to confirm whether and how this model can outperform other proprietary counterparts.

Specifically, we adopt MobileLLaMA-1.4B-Chat and MobileLLaMA-2.7B-Chat. The text input \mathbf{X}_q is first tokenized and processed to the text tokens $\mathbf{H}_q \in \mathbb{R}^{N_t \times D_t}$, where N_t denotes the sequence length of text tokens and D_t is the hidden size of the word embedding space. The text tokens \mathbf{H}_q and the visual tokens \mathbf{H}_v are transformed by the projector, to be concatenated as the input of the language model. The final response \mathbf{Y}_a with a length L is generated in an autoregressive manner as follows,

$$p(\mathbf{Y}_a | \mathbf{H}_v, \mathbf{H}_q) = \prod_{i=1}^L p(y_i | \mathbf{H}_v, \mathbf{H}_q, y_{<i}). \quad (2)$$

3.3. Lightweight Downsample Projector

Inspired by the LDP design of MobileVLM [15], we introduce a new projector to perform better vision-language feature alignment with fewer parameters. It contains three components, *i.e.*, *feature transformation*, *token reduction*, and *positional information enhancement*. First, we employ two point-wise convolution layers on image tokens to match the feature dimension of LLM. Then, we introduce an average pooling layer to extremely compress the number of image tokens. Finally, a very simple but effective module PEG [16] with skip connection is applied to enhance positional information. Compared with LDP [15], this positional part is more efficient and reduces **99.8%** number of parameters, and is slightly faster in running speed.

In formulation, the lightweight downsample projector LDPv2 (connoted as \mathbf{P}) transforms the visual embedding $f_v \in \mathbb{R}^{N_v \times D_v}$ to the modality-aligned visual tokens \mathbf{H}_v with positional enhancement. It is worth noting that this design is also deployment-friendly since it is composed of well-supported operators by mainstream inference frameworks. Given an average kernel k , the number of remaining tokens is only $1/k^2$ of the input features. Concretely, we formulate the proposed LDPv2 as follows,

$$\mathbf{H}_v = \mathbf{P}(f_v) = \begin{cases} f_0 & = PW(GELU(PW(f_v))) \\ f_1 & = AvgPool_{2 \times 2}(f_0) \\ \mathbf{H}_v & = DW(f_1) + f_1. \end{cases} \quad (3)$$

where PW and DW are pointwise and depthwise convolutions respectively, $GELU$ is a GELU [28] activation layer, and $AvgPool_{2 \times 2}$ is a 2×2 average pooling layer.

3.4. Training Strategy

Our training process is split into two stages: pre-training and multi-task training. As illustrated in Table 1, unlike

Method	Pre-training			Multi-task training		
	\mathcal{V}	\mathcal{P}	\mathcal{L}	\mathcal{V}	\mathcal{P}	\mathcal{L}
LLaVA-1.5-7B [44]	\times	\checkmark	\times	\times	\checkmark	\checkmark
ShareGPT4V-7B [10]	\checkmark	\checkmark	\checkmark	\times	\checkmark	\checkmark
MobileVLM [15]	\times	\checkmark	\times	\times	\checkmark	\checkmark
MobileVLM V2	\times	\checkmark	\checkmark	\times	\checkmark	\checkmark

Table 1. **Comparison of training strategy.** \mathcal{V} , \mathcal{P} and \mathcal{L} respectively represent the vision encoder, the projector, and the language model. \checkmark indicates that the corresponding model parameters are optimized in training, \times signifies that the parameters are frozen, and \checkmark represents that a subset of model parameters is frozen.

previous training paradigm of LLaVA-1.5 [44] and MobileVLM [15], MobileVLM V2 opens the projector and large language model consistently in both stages, with the visual encoder frozen.

3.4.1 Pre-training

Most Vision-Language Models (VLMs) [7, 44] commonly freeze the visual encoder and the language model during pre-training to avoid optimization difficulties. ShareGPT-4V [10] partially freezes the visual encoder and makes the language model trainable. In our case, we initialize the weights of the visual encoder and language model from CLIP ViT-L/14 [58] and MobileLLaMA [15], respectively. This initialization serves as a robust foundation for the subsequent unified training process. We allow for full training of the projector and the LLM while fixing the vision encoder. Note that freezing ViT also reduces the training cost. The model’s training objective is then concentrated on the prediction of the next token, utilizing an autoregressive loss function. By honing in on this specific task, the model is better equipped to learn the intricacies of language generation in the context of visual information, leading to improved performance on multimodal tasks.

As reported in Table 2, during the pretraining stage, our model was trained using the ShareGPT4V-PT dataset [10], comprising 1.2 million image-text pairs. This dataset is instrumental in enhancing the model’s image-text alignment capabilities, which is a critical aspect of multimodal representation learning.

3.4.2 Multi-task Training

After the pre-training phase of image-text alignment learning, MobileVLM V2 has acquired a rudimentary capability to comprehend image content. However, it lacks proficiency in utilizing visual information for analysis and dialogue in a suite of downstream tasks. Consequently, during

Datasets	Type	Samples
Pretraining		
ShareGPT4V-PT [10]	Caption	1.2M
Multi-task training		
Visual Dialog [19]	Conversation	123K
Text-VQA [60]	VQA(Open)	35K
VSR [43]	VQA(Open)	13K
VIGC [64]	VQA(Open)	37K
IConQA [50]	VQA(MC)	107K
SQA [49]	VQA(MC)	13K
COCO [11]	Caption	592K
SBU [54]	Caption	844K
ShareGPT4V [10]	Mixed	665K
Total		3.6M

Table 2. **Datasets used for MobileVLM V2 training.** Column “Type” is the task of the dataset, where “Mixed” indicates that the dataset contains samples belonging to various tasks. Column “Samples” is the number of image-text pairs in each dataset.

the multi-task training phase, we introduce multiple vision-language tasks, engaging parameters in the training process to endow the model with the capacity for multi-task analysis and image-text conversing.

In the multi-task training phase, we employ a plethora of datasets featuring a variety of tasks to further develop the model’s array of skills. As outlined in Table 2, these datasets are meticulously chosen to improve conversational abilities using the Visual Dialog dataset [19], OCR skills through the TextVQA dataset [60], scene understanding capabilities via COCO Caption [11] and SBU [54] datasets, and location understanding by the VSR dataset [43] *etc.* Note that we have cleaned and refined the SBU dataset, where the data volume might not match exactly with the officially released one. In total, the aggregated data for this phase consists of 2.4 million samples, ensuring a comprehensive learning experience across different modalities and tasks. Examples regarding the dialogue formats across the various datasets are provided in the Appendix A.

4. Experiment

4.1. Training

Pre-training As listed in Table 3, LDPv2 is randomly initialized in this stage, and the language model and the vision encoder are initialized with the pre-trained weights from MobileLLaMA and CLIP ViT-L/14, respectively. For optimization, we utilize the AdamW optimizer [48] with no weight decay. Maximum learning rates for LDPv2 and other components are configured as $1e^{-3}$ and $2e^{-5}$, respectively, following a cosine learning rate schedule. The pre-training phase involves a global batch size of 256 across 8

NVIDIA A100 GPUs for about 5 hours.

Multi-task Training During this training stage, the weights of MobileVLM V2 are initialized from the first stage. Most training hyper-parameters are similar to the pre-training phase. Additionally, the learning rate is set to $4e^{-5}$. Training in this stage requires 8 NVIDIA A100 GPUs for around 9 hours with a global batch size of 128. Detailed training settings are shown in Table 3.

Configuration	Pre-training	Multi-task training
Vision encoder init	CLIP ViT-L	CLIP ViT-L
LLM init	MobileLLaMA	MobileVLM V2 PT
Projector init	Random	MobileVLM V2 PT
Image resolution	336 ²	336 ²
Image token num	144	144
Global batch size	256	128
Training steps	5K	19K
Optimizer	AdamW	AdamW
LR schedule	Cosine decay	Cosine decay
Projector LR	$1e^{-3}$	$4e^{-5}$
Base LR	$2e^{-5}$	$4e^{-5}$
Weight decay	0	0
Warm-up ratio	0.03	0.03
DeepSpeed Stage	2	3

Table 3. **Training hyperparameters of MobileVLM V2.** Note that MobileVLM V2 PT indicates the parameters saved after the pre-training phase. LR denotes learning rate.

4.2. Comparisons with State-of-the-art Methods

Following MobileVLM, we adopt a list of benchmarks including image question answering series GQA [29], SQA [49], TextVQA [60], comprehensive benchmarks MME [25], MMBench [47], and object hallucination benchmark POPE [40].

Comparison with SOTA methods. We evaluate the performance of MobileVLM V2 and show the accuracy result in Table 4. Note that previous models mainly focus on accuracy improvement, regardless of the run time latency. Although our models are targeted for real applications regarding two aspects, they outperform most of the previous models with clear margins. With an advantage of 75% faster speed, MobileVLM V2 3B still outperforms a very recent work MoE-LLaVA-2.7B \times 4 [41] by 1.4 points on the average score. Noting MoE-LLaVA-2.7B \times 4 [41] already achieves comparable or better accuracy over many 7B+ VLMs. Many of them introduce large amounts of training costs. In contrast, our method achieves new state-of-the-art results while our training cost is comparable to computation-friendly LLaVA-1.5 [45].

Latency Comparison. Since many models haven’t been supported by advanced mobile inference frameworks, we

compare the latency of a larger spectrum of recent models using the PyTorch framework on an NVIDIA A100 GPU, as shown in Figure 3. MobileVLM V2 models are generally faster and stronger in terms of token generation and average scores on tested benchmarks. Especially, MobileVLM V2 1B/3B is 37.37 tokens/s and 28.97 tokens/s respectively, which are both $1.65\times$ faster than their counterparts of MoE-LLaVA, yet with higher average performance.

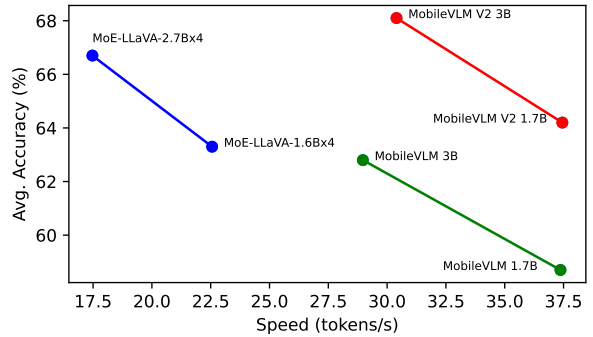


Figure 3. Speed comparison of SOTA VLMs on an NVIDIA A100 GPU. The accuracy is averaged on six VLM benchmarks (see Table 4). (tested with a batch size of 1, generating 256 tokens).

Comparison with MoE-LLaVA. MoE-LLaVA utilizes several experts to achieve good performance, and each expert is a small model to improve the inference speed. Although only a proportion of parameters is activated, it still requires storing the whole parameters, which inevitably incurs IO overhead in mobile scenarios. Moreover, it’s non-trivial to apply model compressing tricks [14, 24, 37, 74] to these models. In contrast, our method can be well-supported and optimized for deployment. In a word, our inference speed advantage evaluated on the Tesla A100 GPU will be further enlarged if tested on resource-limited environments. In principle, MobileVLM V2 can also be combined with its MoE design. But how to combine it without sacrificing the memory and latency advantage of MobileVLM V2 remains as our future work.

Comparison with MobileVLM. Table 4 shows that MobileVLM V2 significantly improves the accuracy performance of MobileVLM. *The average accuracy is boosted by 5.3 points.* Since these two methods share the same encoders both for vision and language modalities, we attribute the improvement to the enhanced data, training strategies, and the new design of the projector. This good performance of MobileVLM V2 indicates that MobileLLaMA from [15] is also a good baseline for small language models because it’s built on the open resource dataset Redpajama [17], which is reproducible and has a low risk of evaluation data leakage. Exploring open and more powerful small language models remains as our future work.

Method	LLM	Res.	GQA	SQA ^I	VQA ^T	POPE	MME ^P	MMB ^{dev}	Avg.
IDEFICS-80B [34]	LLaMA-65B	224	45.2	—	30.9	—	—	54.5	—
BLIP-2 [35]	Vicuna-13B	224	41.0	61.0	42.5	85.3	1293.8	—	—
InstructBLIP [18]	Vicuna-13B	224	49.5	63.1	50.7	78.9	1212.8	—	—
Shikra [9]	Vicuna-13B	224	—	—	—	—	—	58.8	—
Openflamingo [2]	MPT-7B	336	—	—	33.6	—	—	4.6	—
Qwen-VL [4]	Qwen-7B	448	59.3	67.1	63.8	—	1487.6	38.2	—
mPLUG-Owl [70]	LLaMA-7B	224	—	—	—	—	967.3	49.4	—
IDEFICS-9B [34]	LLaMA-7B	224	38.4	—	25.9	—	—	48.2	—
MiniGPT-v2 [8]	LLaMA-7B	448	60.3	—	—	—	—	12.2	—
MiniGPT-4 [75]	Vicuna-7B	224	32.2	—	—	—	581.7	23.0	—
InstructBLIP [18]	Vicuna-7B	224	49.2	60.5	50.1	—	—	36.0	—
LLaVA-1.5 [44]	Vicuna-7B	336	62.0	66.8	58.2	85.9	1510.7	64.3	68.8
ShareGPT4V [10]	Vicuna-7B	336	63.3	68.4	60.4	85.7	1567.4	68.8	70.8
MoE-LLaVA-1.6B×4 [41]	StableLM-1.6B	336	60.4	62.6	47.8	84.3	1300.8 [*]	59.4	63.3
MoE-LLaVA-2.7B×4 [41]	Phi-2.7B	336	61.1	68.7	50.2	85.0	1396.4 [*]	65.5	66.7
MobileVLM 1.7B [15]	MobileLLaMA 1.4B	336	56.1	57.3	41.5	84.5	1196.2	53.2	58.7
MobileVLM V2 1.7B	MobileLLaMA 1.4B	336	59.3	66.7	52.1	84.3	1302.8	57.7	64.2
MobileVLM 3B [15]	MobileLLaMA 2.7B	336	59.0	61.2	47.5	84.9	1288.9	59.6	62.8
MobileVLM V2 3B	MobileLLaMA 2.7B	336	61.1	70.0	57.5	84.7	1440.5	63.2	68.1
MobileVLM V2 7B	Vicuna-7B	336	62.6	74.8	62.3	85.3	1560.7	69.2	72.1
MobileVLM V2 7B w/o AvgPool	Vicuna-7B	336	64.6	74.8	66.8	86.1	1558.7	70.8	73.5

Table 4. **Comparison with SOTA methods on six VLM benchmarks.** GQA [29]; SQA^I: ScienceQA-IMG [49]; VQA^T: TextVQA [60]; POPE [40]; MME^P: MME Perception [25]; MMB^{dev}: MMBench-dev [47]; Column *Res.* is the image resolution of vision model. Column *Avg.* indicates the average accuracy on six evaluation benchmarks. The values in the MME^P column should be divided by 2000 before being included in the average accuracy calculation. * represents that results not provided in the paper [41] were evaluated using their latest repo (commit 5ba14e8). The value of MobileVLM on SQA^I is erroneous in [15] and is corrected in this version.

4.3. Model Scaling Analysis

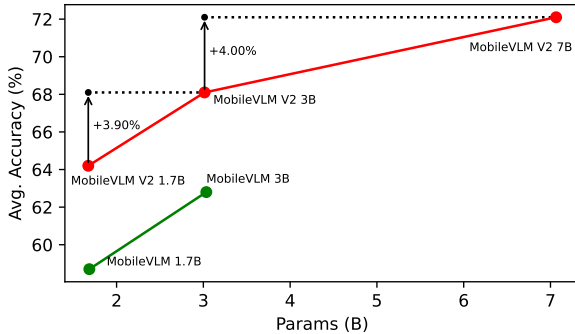


Figure 4. Average performance improvement on six VLM benchmarks when scaling MobileVLM V2’s models across several standard tasks (see also Table 4).

While our target is to design powerful multimodal models for resource-constrained scenarios, we further scale up our model to verify the upper bound of performance. This also forms an apple-to-apple comparison with many existing VLMs. Specifically, we utilize Vicuna-7B as the LLM

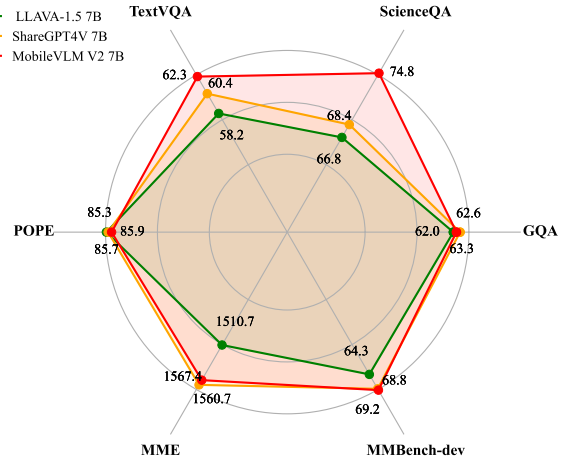


Figure 5. Radar plot of MobileVLM V2’s performance compared with its peers on 6 standard benchmarks.

model and show the result in Figure 4, where a comprehensive multimodal performance improvement emerges. We compare our MobileVLM V2 7B with mainstream large-scale VLMs like LLaVA-1.5 7B [44] and ShareGPT4V 7B

[10] in both accuracy and the inference speed, which is shown in Figure 5 and Table 5. It can be seen that our MobileVLM V2 7B not only secures a significant performance edge across multiple benchmarks but also establishes a clear lead in inference speed over large-scale representatives. While being nearly 20% faster, MobileVLM V2 7B outperforms ShareGPT4V by 1.3 points regarding the average performance. This further illustrates the effectiveness of our data scaling strategy, training strategy, and the novel projector design.

Given the observation that the latency gap is narrowed upon the 7B model, we remove the token reduction component, *i.e.*, average pooling with a kernel of 2. Under this setting, our 7B model has the same latency speed as ShareGPT4V. We show the result in the last row of Table 4. MobileVLM V2 7B (w/o AvgPool) achieves an average score of 73.5, which significantly outperforms LLaVA-1.5 by 4.7 points. Compared with MobileVLM V2 7B (with AvgPool), the performance gain mainly comes from the increased score on the TextVQA task, which is an OCR task. This task contains many small objects where token reduction could be harmful. We retain how to effectively make use of high-resolution input in our future work.

4.4. Latency Measurement on Mobile Devices

To maintain consistency, we measure the inference latency of MobileVLM V2 on the NVIDIA AGX Jetson Orin platform with identical configurations as MobileVLM [15]. The `llama.cpp` [27] framework is employed as the inference framework. Specifically, for the Jetson Orin platform, we develop more efficient CUDA implementations for LDPv2 (Section 3.3) to fully leverage the hardware to have the best performance. Table 5 reports the comparison results of inference latency. To more objectively evaluate the inference performance, we introduce $Eval_{avg}$, which denotes the actual generation speed of output tokens, calculated by dividing the total time by the number of 256 output tokens.

Inference Latency on NVIDIA Jetson Orin. On the Jetson Orin platform, we conclude that MobileVLM V2 demonstrates a lower inference latency than its counterparts at the same parameter scale. As shown in Table 5, MobileVLM V2 achieves the best performance in the inference speed, which can be attributed to a more lightweight design of the projector: we optimize the original 576 tokens of visual prompts to 144 ones, while the performance on the average accuracy remains the same or becomes even better.

5. Ablation Study

5.1. Effect of Data Scaling

In this section, we explore the effect of the data enhancement strategy. As shown in Table 6, the first row represents

Model	Avg. accuracy (tokens/s)	$Eval_{avg}$	Total (s)
ShareGPT4V-7B	70.8	13.00	19.69
LLaVA-1.5 7B	68.8	12.96	19.75
MobileVLM V2 7B	72.1	15.49	16.53
LLaVA-1.5 3.3B	62.7	20.45	12.52
MobileVLM 3B	62.8	30.80	8.31
MobileVLM V2 3B	68.1	30.80	8.38
LLaVA-1.5 1.4B	55.7	43.39	5.90
MobileVLM 1.7B	58.7	49.80	5.14
MobileVLM V2 1.7B	64.2	51.63	4.96

Table 5. Latency comparison of mobile-scale VLMs on NVIDIA Jetson Orin. The language model of VLMs is quantized to 4-bit with `llama.cpp`. The average accuracy is evaluated on the same six benchmarks as Table 4.

	D	T	GQA	SQA ^I	VQA ^T	POPE	MME ^P	MMB ^{dev}	Avg.
(a)	✗	✗	56.1	57.3	41.5	84.5	1196.2	53.2	58.7
(b)	✓	✗	57.5	63.9	49.8	83.9	1157.5	51.6	60.8
(c)	✓	✓	58.5	65.4	50.8	83.4	1262.6	55.4	62.8

Table 6. Ablation results of Data Strategy and Training Strategy on MobileVLM 1.7B. ‘D’ and ‘T’ represent our Data Strategy and Training strategy, respectively.

the baseline of MobileVLM 1.7B. We replace the pretraining dataset with the ShareGPT4V dataset [10] and expand the instruction tuning dataset to 2.4M, as described in Section 3.4.2. Note that the training strategy and model architecture remain the same as the baseline. Performance gains on GQA, SQA, and TextVQA demonstrate that the model’s ability in cognition and dialogue has been improved after data enhancement. However, we also observe that in the configuration of more high-quality data, maintaining the original training strategy cannot fully exploit the data benefits and leads to performance degradation on MME and MMBench.

5.2. Effect of Training Strategy

Based on the observations from Section 5.1, we further explore a more reasonable training strategy. Under the setting of enhanced data and small-scale VLMs, unlocking the language model in the whole training stage enables more effective learning. Therefore, we opt for simultaneous fine-tuning of the projector as well as the language model. Compared results are shown in the second and third rows in Table 6. We can see that this training setup enables a comprehensive improvement in model performance across most VLM benchmarks, with a 2-point gain on average accuracy.

VL Projector Architecture Design	w/o D&T	Tokens	Params.	GQA	SQA ¹	VQA ^T	POPE	MME	MMB ^{dev}	Avg.
$[PW]_{\times 2}$	✗	576	6.30M	56.9	57.1	43.7	85.7	1137.7	52.8	58.8
$[PW]_{\times 2}[DW^{\kappa=1}PW]_{\times 1}[DW^{\kappa=2}PW]_{\times 1}$	✗	144	18.94M	56.1	57.3	41.5	84.5	1196.2	53.2	58.7
$[PW]_{\times 2}$	✓	576	6.30M	59.9	63.7	53.9	85.0	1271.3	56.0	63.7
$[PW]_{\times 2}[DW^{\kappa=1}PW]_{\times 1}[DW^{\kappa=2}PW]_{\times 1}$	✓	144	18.94M	58.5	65.4	50.8	83.4	1262.6	55.4	62.8
$[PW]_{\times 2}[AvgPool^{\rho=2}]_{\times 1}$	✓	144	6.30M	59.3	65.0	53.1	84.0	1292.2	54.5	63.2
$[PW]_{\times 2}[AvgPool^{\rho=2}]_{\times 1}[LearnablePE]_{\times 1}$	✓	144	6.59M	59.1	67.1	52.3	84.3	1286.7	55.5	63.7
$[PW]_{\times 2}[AvgPool^{\rho=2}]_{\times 1}[DW^{\kappa=1}]_{\times 1}$	✓	144	6.32M	59.3	66.7	52.1	84.3	1302.8	57.7	64.2

Table 7. The exploration of projector design based on MobileLLaMA 1.4B. The PW represents pointwise-conv and DW is depthwise-conv. The subscript \times indicates the number of times the corresponding module is stacked repeatedly. The superscript κ indicates the 3×3 conv stride and ρ indicates the pooling kernel size. Note that D represents our proposed Data Strategy and T for Training Strategy in MobileVLM V2. The green row is the proposed LDPv2 in our MobileVLM V2.

5.3. Effect of Projector

Looking back at MobileVLM, the proposed LDPv1 reduces the number of tokens by 75% ($576 \rightarrow 144$) while maintaining almost equivalent performance, see the first and second rows in Table 7. Based on the newly proposed data configuration and training strategy, the above corresponding architectures can achieve 5.4 and 4.5 average improvements respectively (the third and fourth rows). However, when we try to continue to explore better alignment methods of visual and language features, we observe that increasing the number of learnable parameters can easily lead to optimization dilemmas, resulting in a decline in the overall alignment effect. Based on this observation, we firstly replace the $[DW^{\kappa}PW]$ block in LDPv1 with an embarrassingly simple parameter-free operation, *i.e.*, 2×2 average pooling, and find that we could obtain an average performance improvement of 0.4 points (see row 5 in Table 7). Secondly, to make the aligned visual features with stronger positional information, the simplest idea is to employ learnable positional encoding for enhancement, which brings 0.5 performance gains in the 6th row. Further, inspired by the claims of PEG [16], whose position encoding is dynamically generated and conditioned on the local neighborhood of the input tokens, we then replace the learnable PE with a PEG layer to obtain better-enhanced features. The seventh row in Table 7 indicates an improvement of 0.5 points. It is worth mentioning that, a PEG layer applied in MobileVLM V2 1.7B only contains $0.02M$ ($2048 \times 3 \times 3$) parameters. Compared with the $[DW^{\kappa}PW]$ block in MobileVLM, the number of learnable parameters has dropped by nearly $630 \times$ ($12.64M \rightarrow 0.02M$), but the overall performance is improved by 1.4 points, which demonstrates the effectiveness of our design.

6. Conclusion

In a nutshell, we propose a series of efficient vision language models called MobileVLM V2 based on Mo-

bileVLM [15]. We explore data scaling schemes, improved training strategies, and efficient modality alignment design to improve the overall performance under the setting of small VLM models. With a comparable training cost as [44], our method achieves a new state-of-the-art Pareto front in terms of accuracy and latency, targeted for real product environments. Our model outperforms many larger models with substantial inference advantages, which paves a promising way to enjoy advanced AI on resource-limited scenarios.

Acknowledgements: This work was in part supported by National Key R&D Program of China (No. 2022ZD0118-700).

References

- [1] Syeda Nahida Akter, Zichun Yu, Aashiq Muhamed, Tianyue Ou, Alex Bäuerle, Ángel Alexander Cabrera, Krish Dholakia, Chenyan Xiong, and Graham Neubig. An in-depth look at gemini’s language abilities. *arXiv preprint arXiv:2312.11444*, 2023. 1
- [2] Anas Awadalla, Irena Gao, Joshua Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. Openflamingo, Mar. 2023. 6
- [3] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 2
- [4] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023. 1, 2, 6
- [5] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O’Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyz-

- ing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR, 2023. 2
- [6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 2
- [7] Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh. Honeybee: Locality-enhanced projector for multimodal llm. *arXiv preprint arXiv:2312.06742*, 2023. 4
- [8] Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny. Minigt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*, 2023. 6
- [9] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm’s referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023. 6
- [10] Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*, 2023. 1, 2, 4, 6, 7
- [11] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server, 2015. 4
- [12] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In *International Conference on Machine Learning*, pages 1931–1942. PMLR, 2021. 2
- [13] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022. 2
- [14] Xiangxiang Chu, Liang Li, and Bo Zhang. Make repvgg greater again: A quantization-aware approach. In *AAAI*, 2024. 5
- [15] Xiangxiang Chu, Limeng Qiao, Xinyang Lin, Shuang Xu, Yang Yang, Yiming Hu, Fei Wei, Xinyu Zhang, Bo Zhang, Xiaolin Wei, et al. Mobilevlm: A fast, reproducible and strong vision language assistant for mobile devices. *arXiv preprint arXiv:2312.16886*, 2023. 1, 2, 3, 4, 5, 6, 7, 8
- [16] Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, and Chunhua Shen. Conditional positional encodings for vision transformers. In *The Eleventh International Conference on Learning Representations*, 2023. 3, 8
- [17] Together Computer. Redpajama: An open source recipe to reproduce llama training dataset, 2023. 5
- [18] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*, 2023. 6
- [19] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 326–335, 2017. 4
- [20] Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Xilin Wei, Songyang Zhang, Haodong Duan, Maosong Cao, Wenwei Zhang, Yining Li, Hang Yan, Yang Gao, Xinyue Zhang, Wei Li, Jingwen Li, Kai Chen, Conghui He, Xingcheng Zhang, Yu Qiao, Dahua Lin, and Jiaqi Wang. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *arXiv preprint arXiv:2401.16420*, 2024. 2
- [21] Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335, 2022. 2
- [22] David Eigen, Marc’Aurelio Ranzato, and Ilya Sutskever. Learning factored representations in a deep mixture of experts. 2013. 1
- [23] Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot, 2023. 2
- [24] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022. 2, 5
- [25] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 5, 6
- [26] Chaoyou Fu, Renrui Zhang, Zihan Wang, Yubo Huang, Zhengye Zhang, Longtian Qiu, Gaoxiang Ye, Yunhang Shen, Zhang Mengdan, Peixian Chen, Sirui Zhao, Shaohui Lin, Deqiang Jiang, Di Yin, Peng Gao, Ke Li, Hongsheng Li, and Xing Sun. A challenger to gpt-4v? early explorations of gemini in visual expertise. *arXiv preprint arXiv:2312.12436*, 2023. 1, 2
- [27] Georgi Gerganov. llama.cpp. <https://github.com/ggerganov/llama.cpp>. [Accessed: 2023-11-07]. 7
- [28] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 3
- [29] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019. 5, 6
- [30] Robert A Jacobs, Michael I Jordan, Stuart J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991. 1
- [31] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. Accessed: 2023-03-01. 12

- [32] Jing Yu Koh, Ruslan Salakhutdinov, and Daniel Fried. Grounding language models to images for multimodal generation. *arXiv preprint arXiv:2301.13823*, 2023. **2**
- [33] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. *arXiv preprint arXiv:2308.00692*, 2023. **2**
- [34] Hugo Laurençon, Lucile Saulnier, Léo Tronchon, Stas Bekman, Amanpreet Singh, Anton Lozhkov, Thomas Wang, Siddharth Karamcheti, Alexander M Rush, Douwe Kiela, et al. Obelisc: An open web-scale filtered dataset of interleaved image-text documents. *arXiv preprint arXiv:2306.16527*, 2023. **6**
- [35] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. **6**
- [36] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021. **2**
- [37] Liang Li, Qingyuan Li, Bo Zhang, and Xiangxiang Chu. Norm tweaking: High-performance low-bit quantization of large language models. In *AAAI*, 2024. **2, 5**
- [38] Qingyuan Li, Ran Meng, Yiduo Li, Bo Zhang, Liang Li, Yifan Lu, Xiangxiang Chu, Yerui Sun, and Yuchen Xie. A speed odyssey for deployable quantization of llms. *arXiv preprint arXiv:2311.09550*, 2023. **2**
- [39] Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. Textbooks are all you need ii: phi-1.5 technical report, 2023. **2**
- [40] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023. **5, 6**
- [41] Bin Lin, Zhenyu Tang, Yang Ye, Jiayi Cui, Bin Zhu, Peng Jin, Junwu Zhang, Munan Ning, and Li Yuan. Moe-llava: Mixture of experts for large vision-language models, 2024. **1, 2, 5, 6**
- [42] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Eur. Conf. Comput. Vis.*, pages 740–755. Springer, 2014. **12**
- [43] Fangyu Liu, Guy Edward Toh Emerson, and Nigel Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 2023. **4**
- [44] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. **4, 6, 8, 12**
- [45] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv:2310.03744*, 2023. **5**
- [46] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. **1, 2**
- [47] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023. **5, 6**
- [48] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. **4**
- [49] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems*, pages 27730–27744, 2022. **1, 4, 5, 6**
- [50] Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. *arXiv preprint arXiv:2110.13214*, 2021. **4**
- [51] OpenAI. ChatGPT. <https://openai.com/blog/ChatGPT/>, 2023. Online; accessed 2023-01-01. **2**
- [52] OpenAI. Gpt-4 technical report. 2023. Technical Report. **2**
- [53] OpenAI. Gpt-4v(ision) system card. 2023. **1**
- [54] Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. Im2text: Describing images using 1 million captioned photographs. In *Neural Information Processing Systems (NIPS)*, 2011. **1, 4**
- [55] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022. **2**
- [56] Tianduo Wang Peiyuan Zhang, Guangtao Zeng and Wei Lu. Tinyllama, Sep 2023. **2**
- [57] Renjie Pi, Jiahui Gao, Shizhe Diao, Rui Pan, Hanze Dong, Jipeng Zhang, Lewei Yao, Jianhua Han, Hang Xu, and Lingpeng Kong Tong Zhang. Detgpt: Detect what you need via reasoning. *arXiv preprint arXiv:2305.14167*, 2023. **2**
- [58] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. **1, 2, 3, 4**
- [59] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022. **2**
- [60] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019. **1, 4, 5, 6**
- [61] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. **2**

- [62] Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. Galactica: A large language model for science. 2022. [2](#)
- [63] InternLM Team. Internlm: A multilingual language model with progressively enhanced capabilities. <https://github.com/InternLM/InternLM>, 2023. [2](#)
- [64] Bin Wang, Fan Wu, Xiao Han, Jiahui Peng, Huaping Zhong, Pan Zhang, Xiaoyi Dong, Weijia Li, Wei Li, Jiaqi Wang, et al. Vigc: Visual instruction generation and correction. *arXiv preprint arXiv:2308.12714*, 2023. [4](#)
- [65] Junke Wang, Lingchen Meng, Zejia Weng, Bo He, Zuxuan Wu, and Yu-Gang Jiang. To see is to believe: Prompting gpt-4v for better visual instruction tuning. *arXiv preprint arXiv:2311.07574*, 2023. [2](#)
- [66] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022. [2](#)
- [67] Haoran Wei, Lingyu Kong, Jinyue Chen, Liang Zhao, Zheng Ge, Jinrong Yang, Jianjian Sun, Chunrui Han, and Xiangyu Zhang. Vary: Scaling up the vision vocabulary for large vision-language models. *arXiv preprint arXiv:2312.06109*, 2023. [2](#)
- [68] Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, pages 38087–38099. PMLR, 2023. [2](#)
- [69] Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*, 2023. [2](#)
- [70] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023. [6](#)
- [71] Duzhen Zhang, Yahan Yu, Chenxing Li, Jiahua Dong, Dan Su, Chenhui Chu, and Dong Yu. Mm-llms: Recent advances in multimodal large language models. *arXiv preprint arXiv:2401.13601*, 2024. [1](#)
- [72] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. OPT: Open pre-trained transformer language models. 2022. [2](#)
- [73] Bo Zhao, Boya Wu, and Tiejun Huang. Svit: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087*, 2023. [2](#)
- [74] Sifan Zhou, Liang Li, Xinyu Zhang, Bo Zhang, Shipeng Bai, Miao Sun, Ziyu Zhao, Xiaobo Lu, and Xiangxiang Chu. Lidar-ptq: post-training quantization for point cloud 3d object detection. *International Conference on Learning Representations (ICLR 2024)*, 2024. [5](#)
- [75] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. [1](#), [2](#), [6](#)
- [76] Yichen Zhu, Minjie Zhu, Ning Liu, Zhicai Ou, Xiaofeng Mou, and Jian Tang. Llava- ϕ : Efficient multi-modal assistant with small language model. *arXiv preprint arXiv:2401.02330*, 2024. [2](#)

A. Dialogue formats of various datasets.

During the pre-training phase, we utilized the 1.2 million image-text pairs from the pre-training phase of ShareGPT4V, which primarily includes COCO [42], SAM [31], and LLaVA-1.5 pre-training data [44]. Subsequently, in the multi-task training phase, we collected 2.4 million high-quality instruction data. A detailed format of these data is provided in Table 8 and Table 9.

B. Examples of MobileVLM V2.

Figure 6 shows qualitative results of MobileVLM V2 1.7B in multimodal conversations in various scenarios including of fine-grained attribute understanding, image understanding and relation reasoning, attributes and spatial relationships understanding, OCR and world knowledge, *etc.*


Dataset	Image	Dialogue Content
VIGC		<p>User: <image>\nWhat is the person in the image doing?</p> <p>Assistant: The person in the image is skiing down a snow-covered slope.</p>
IconQA		<p>User: <image>\nSelect the picture that shows equal parts.A. First image.\nB. Second image.\nC. Third image.\nAnswer with the option's letter from the given choices directly.</p> <p>Assistant: C.</p>
TextVQA		<p>User:<image>\nWhat brand name is on the box?\n Reference OCR token: Cergo, AUDREY, is, all, you, need, 3com, This \nAnswer the question using a single word or phrase.</p> <p>Assistant: 3com.</p>
COCO Caption	 	<p>User: <image>\nWhat are the key elements in this picture?</p> <p>Assistant: A bicycle replica with a clock as the front wheel.</p> <p>User: <image>\nWhat is this photo about?</p> <p>Assistant: A room with blue walls and a white sink and door.</p>

Table 8. Examples from the multi-task training data for MobileVLM V2 (Part 1).

Dataset	Image	Dialogue Content
Visual Dialog		<p>User: <image>\nHow many bikes there?</p> <p>Assistant: 3.</p> <p>User: What color are bikes?</p> <p>Assistant: I see green red and white.</p> <p>User: Are they parked on stock parking?</p> <p>Assistant: NO.</p> <p>User: What color of their helmets?</p> <p>Assistant: green and white and other is black.</p>
SBU		<p>User: <image>\nCan you elaborate on the elements of the picture provided?</p> <p>Assistant: A wooden chair in the living room.</p>
SQA		<p>User:<image>\nWhich of these states is farthest east? \nContext:N/A\nOptions:(A) Montana (B) New Mexico (C) Arkansas (D) Nebraska\n</p> <p>Assistant: Maps have four cardinal directions, or main directions. Those directions are north, south, east, and west.\nA compass rose is a set of arrows that point to the cardinal directions. A compass rose usually shows only the first letter of each cardinal direction.\nThe north arrow points to the North Pole. On most maps, the north is at the top of the map. To find the answer, look at the compass rose. Look at which way the east arrow is pointing. Arkansas is the farthest east. The answer is C.</p>
VSR		<p>User: <image>\nIs the pizza above the couch?\nAnswer the question using a single word or phrase.</p> <p>Assistant: No.</p>

Table 9. Examples from the multi-task training data for MobileVLM V2 (Part 2).

Fine-grained attribute understanding



User <image> What's her mood like?

MobileVLM V2 The young woman in the image is smiling, which suggests that she is in a positive and cheerful mood.

User What color sweater is she wearing?

MobileVLM V2 She is wearing a red sweater.

Image understanding and relation reasoning



User <image> What is this photo about?

MobileVLM V2 A man and a boy fishing on a lake.

User How many person there?

MobileVLM V2 There are 2 people in the image.

User What might be the relationship between these people?

MobileVLM V2 Based on the image, it is likely that the man and the boy are father and son. They are sitting together on a log, fishing in a body of water, and the father is holding a fishing rod. This suggests that they share a close bond and enjoy spending time together outdoors, engaging in a recreational activity like fishing. The father's presence and the father-son relationship are further supported by the fact that they are fishing together, which is a common bonding activity for parents and their children.

Attributes and spatial relationships understanding



User <image> What are they doing?

MobileVLM V2 A baby is playing with a teddy bear.

User What is on the right of this image?

MobileVLM V2 There is a teddy bear on the right side of the image.

OCR and world knowledge



User <image> Describe this image in detail.

MobileVLM V2 The image captures a charming storefront, nestled in a quaint street corner. The store's name, "The Kitchen Store", is prominently displayed in white letters above the door, inviting passersby to explore its offerings. Above the door, a wreath of white and green berries adds a touch of nature to the scene. The store is adorned with a variety of items, including a sign that reads "The Root Kitchen Store", suggesting a range of culinary delights within. The window display showcases an array of items, including a vase and a bowl, hinting at the store's diverse range of products. The overall scene paints a picture of a cozy and inviting store, ready to welcome customers into its warm and welcoming interior.

Figure 6. Examples of MobileVLM V2 1.7B in various scenes.