UNLEASHING THE POWER OF SELECTIVE STATE SPACE MODELS IN VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

While emerging multi-modal large language models (MLLM) have demonstrated impressive advances, the quadratic complexity of their Transformer-based LLMs (3B or larger) inevitably leads to considerable computational overhead. On the other hand, the recently proposed selective state space model (i.e., Mamba) enjoys both model capacity and computational efficiency, making it an ideal component to enhance MLLM's efficiency and performance. However, recent attempts to introduce Mamba into MLLMs simply replace their LLMs with Mamba, ignoring the unique characteristics of either side. We argue that such a naive combination cannot exhibit the potential of Mamba in MLLMs. In this paper, we delve into harnessing Mamba's unique properties, and propose tailored designs from both multi-modal input and architectural perspectives to unleash its true power. First, we fully utilize Mamba's linear complexity to construct visual long sequences for a thorough perception at a minor efficiency burden. To integrate the scanning mechanism with the built visual long sequence, we devise a novel cross-stitch scanning approach to capture and fuse spatial and semantic properties simultaneously, enhancing the interaction of visual information and the vision-language alignment. Built upon these designs, we propose MambaVLM, a simple yet effective MLLM framework that exhibits highly competitive results across multiple benchmarks. Moreover, our framework is also compatible with Transformer-based LLMs (e.g., Vicuna), demonstrating remarkable training and inference efficiency. Notably, with only 0.66M data and 14 hours training on a single A800 node, our MambaVLM outperforms LLaVA-1.5 by significant margins and performs on par or even better than the 1.4B data trained Qwen-VL. The appealing results from both effectiveness and efficiency aspects indicate the promising prospects of Mamba in MLLMs.

1 INTRODUCTION

The emergence of large language models (LLM) (Brown et al., 2020; Touvron et al., 2023a;b; Gao et al., 2023; Chiang et al., 2023) has exhibited strong linguistic capabilities and logical reasoning abilities. However, LLMs are limited to processing linguistic tasks only, whereas visual capabilities play a crucial role in human perception and real-world applications. Therefore, multimodal large language models (MLLM) (Alayrac et al., 2022; Li et al., 2022; 2023a; Bai et al., 2023; Liu et al., 2024a; 2023c; Dai et al., 2024; Zhu et al., 2023; Karamcheti et al., 2024) that integrate vision and text have received widespread attention in recent times.

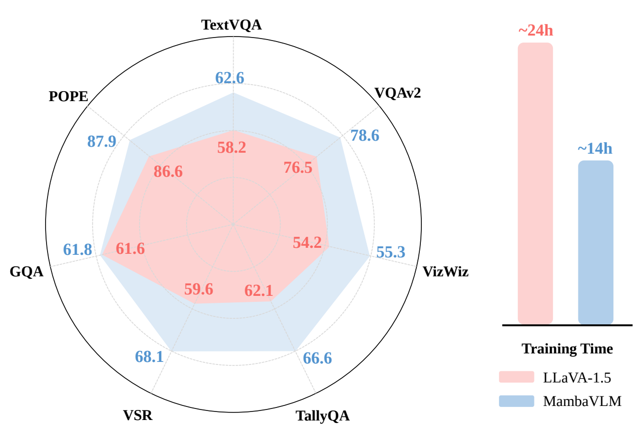

Figure 1: **Comparison with LLaVA-1.5.** Our method outperforms LLaVA-1.5 consistently across 7 benchmarks, while saving more than 40% the training compute.

Typically, MLLMs leverage a vision encoder (ViT) to perceive input images, and project visual tokens into language embedding space. The projected visual tokens, together with tokenized input language data, are sent to an LLM for output response generation that is related to visual content. Bringing LLMs into the vision and language (VL) domain advances a series of multi-modal applications such as visual question answering (VQA) (Antol et al., 2015; Schwenk et al., 2022b), captioning (Karpathy & Fei-Fei, 2015; Vinyals et al., 2015), and referring expression comprehension (REC) (Qiao et al., 2020; Yu et al., 2018).

Current MLLMs typically employ Transformer (Vaswani et al., 2017) as their LLMs. Despite Transformer's excellent ability to model long-range dependencies and numerous successes (Dosovitskiy et al., 2020; Touvron et al., 2021; Liu et al., 2021; Wang et al., 2021), it suffers from a critical issue: the expensive computational cost arising from self-attention's quadratic complexity. Particularly, considering that the transformers utilized in MLLMs are typically large size LLMs with a parameter count of 3B or more (Liu et al., 2023c), using Transformer inevitably incurs significant computational and training overheads. On the other hand, state space models (SSM) (Gu et al., 2021a;c; Smith et al., 2022; Gu et al., 2022) have demonstrated tremendous potential as linear-complexity models in NLP tasks. A representative work is the recently proposed Selective State-Space Model (i.e., Mamba) (Gu & Dao, 2023), which designs an input-dependent selection mechanism to enable the model to choose relevant information flexibly and devises a hardware-friendly algorithm for efficient training and inference. Mamba is shown to outperform Transformer on large-scale datasets and more importantly, its linear complexity endows it with the ability to handle long sequences effectively and superior scaling properties. The success of Mamba naturally leads to a question: Can Mamba perform well in MLLMs, and more importantly, how to unleash the true power of Mamba in MLLMs[1]?

To answer the questions above, we delve into introducing Mamba into MLLM coupled with its unique properties instead of merely using it as the LLM. Facilitated by the linear complexity of Mamba, we can increase the token sequence length at a minor cost. Therefore, we first construct visual long sequences with multiple vision encoders, which not only enrich visual representations but also leverage the advantages of Mamba in handling long sequences (Gu & Dao, 2023). Notably, this design will not undermine the efficiency obviously which is in stark contrast with the common cognition of Transformer-based MLLMs. Then, we also introduce Mamba as a projector to map visual tokens into the language embedding space. Since now we have multiple visual embeddings from the built visual long sequence, existing 1D or 2D token scanning mechanisms can not be applied directly. To solve, we develop a novel 3D token scanning method named cross-stitch scan. By going through multiple visual embeddings with continuous back-and-forth interlace, this design can capture and fuse spatial and semantic properties simultaneously, promoting the comprehensive integration of visual information, thus are well fused for language embedding projection.

Built upon the above designs, we propose a concise and effective MLLM framework termed MambaVLM. Our approach demonstrates strong performance across various multi-modal benchmarks (Singh et al., 2019; Goyal et al., 2017; Gurari et al., 2018; Li et al., 2023b; Hudson & Manning, 2019; Liu et al., 2023a; Acharya et al., 2019), validating the effectiveness of our design. Furthermore, our framework is also compatible with other LLMs (e.g., Vicuna (Chiang et al., 2023)) and demonstrates remarkable training and data efficiency. For instance, with only 0.66M data and 14 hours training on a single A800 node, our MambaVLM performs on par with or even better than the 1.4B data trained Qwen-VL (Bai et al., 2023) (which requires hundreds or thousands of GPU hours).

2 Related Work

2.1 State Space Models

The concept of state-space model (SSM) (Gu et al., 2021a;c; Smith et al., 2022; Gu et al., 2022) can be traced back to the 1960s (Kalman, 1960). LSSL (Gu et al., 2021b) leverages the advantages of continuous-time models (CTMs), RNNs, and CNNs to enable deep SSMs to solve long-range dependencies, but it suffers from large computational and memory requirements imposed by the state representation. Structured State Space (S4) (Gu et al., 2021a) proposes parameterization catering to continuous-time, recurrent and convolutional view of the state space model, which alleviates the computational bottleneck and effectively model long-range dependencies. Mamba (Gu & Dao, 2023)

[1]In this paper, we do not differentiate MLLM and VLM by assuming both of them process vision-language data for generative LLM outputs.

proposes a novel selection mechanism to build selective structured state space model, which extends S4 to select relevant information flexibly. Mamba also devises a hardware-friendly algorithm for efficient training and inferencem and is shown to outperform Transformer on large-scale datasets and more importantly, its linear complexity endows it with the ability to handle long sequences effectively and superior scaling properties. Given the success of Mamba in NLP, many efforts have been made to expand its application to other domains. For instances, Vim (Zhu et al., 2024) combines bidirectional SSM and positional embedding for location-aware visual understanding, extending Mamba to vision tasks. Vmamba (Liu et al., 2024b) devises a cross-scan mechansim to enable effective 2D scanning and demonstrate effective improvements. In this paper, we delve into the potential of Mamba in the context of MLLMs, a more challenging scenario that better demonstrates its advantages as a linear complexity LLM. Very recently, concurrent works VL-Mamba (Qiao et al., 2024) and Cobra (Zhao et al., 2024) also adopt the idea of introducing Mamba into MLLMs. However, these works merely replace the LLM within existing frameworks (LLaVA-1.5 (Liu et al., 2023c) and Prism (Karamcheti et al., 2024) respectively) while we explore Mamba from architectural perspective and propose elaborate designs to build a strong framework that unleash the power of Mamba in MLLM.

2.2 Multimodal Large Language Model

Researchers have shown keen interest in visual-language models for years (Su et al., 2019; Chen et al., 2020; Li et al., 2020; Zhang et al., 2021; Kim et al., 2021). However, despite the progress made, these models still possess several limitations such as weak instruction-following capabilities, poor generalization abilities, and lack of in-context understanding (Bai et al., 2023). Recently, aided by the rapid gains of large language models (LLM) (Brown et al., 2020; Touvron et al., 2023a;b; Chiang et al., 2023), many researchers are now devoting their efforts to building powerful multimodal large language models (MLLM) (Alayrac et al., 2022; Li et al., 2023a; Liu et al., 2024a; Dai et al., 2024; Zhu et al., 2023; Karamcheti et al., 2024) that leverage the strong capabilities of LLMs. Flamingo (Alayrac et al., 2022) utilizes a gated cross-attention module to align the frozen vision foundation models and LLMs. BLIP-2 (Li et al., 2023a) proposes a Q-Former to bridge the modality gap, demonstrating strong performances. LLaVA (Liu et al., 2024a), MiniGPT-4 (Zhu et al., 2023) and InstructBLIP (Dai et al., 2024) focus on the instruction-following ability of MLLMs, and introduce visual instruction tuning. VILA (Lin et al., 2023) and Prism (Karamcheti et al., 2024) dive into the component ablation of MLLMs. LISA (Lai et al., 2023) and Lenna (Wei et al., 2023) explore the reasoning segmentation and detection of MLLMs respectively, exhibiting expressive capacities. Previous works primarily focus on the data and task dimensions, with little exploration into the architectural framework of MLLMs. It is a common practice for MLLMs to utilize Transformer-based LLMs, whose self-attention can incur expensive computational cost due to the quadratic complexity. Furthermore, current MLLM frameworks typically use CLIP (Radford et al., 2021) to extract visual features and then use a simple MLP layer for aligning visual and textual features, which may not fully leverage the potential of the vision model and LLM. Different from previous arts, our paper explores the potential of Mamba in MLLMs and to better unleash the capabilities of Mamba, we propose a concise and effective framework from the perspective of structural design, demonstrating strong performances across multiple benchmarks.

3 Method

In this section, we first introduce the preliminaries of state space models in Section 3.1. Then, we elaborate on the specific components of the proposed MambaVLM in Section 3.2, which comprises visual long sequence, Mamba projector, and the Mamba LLM.

3.1 Preliminaries

State space models (SSM) (Gu et al., 2021a;c; Smith et al., 2022; Gu et al., 2022) can be regarded as linear time-invariant systems that maps a 1-D function or sequence $\mathrm{x}(\mathrm{t}) \in \mathbb{R}$ to the out response $\mathrm{y}(\mathrm{t}) \in \mathbb{R}$ through a hidden state $\mathrm{h}(\mathrm{t}) \in \mathbb{R}^{\mathrm{N}}$. This system can be formulated as linear ordinary differential equations (ODEs), using $\mathbf{A} \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ as the evolution parameter and $\mathbf{B} \in \mathbb{R}^{\mathrm{N}\times 1}$, $\mathbf{C} \in \mathbb{R}^{1\times\mathrm{N}}$ as the projection parameters.

$$\begin{aligned} \mathrm{h}'(\mathrm{t}) &= \mathbf{A}\mathrm{h}(\mathrm{t}) + \mathbf{B}\mathrm{x}(\mathrm{t}), \\ \mathrm{y}(\mathrm{t}) &= \mathbf{C}\mathrm{h}(\mathrm{t}). \end{aligned} \tag{1}$$

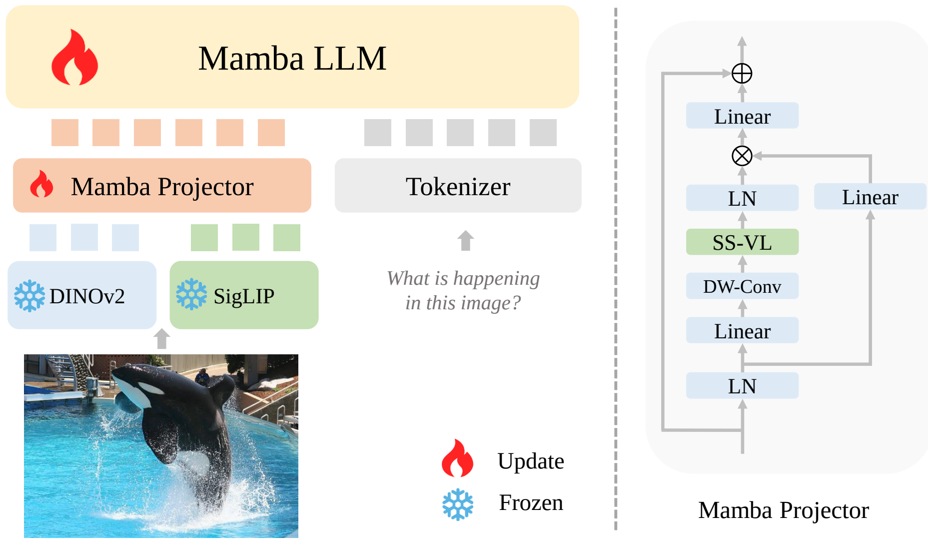

Figure 2: **Overview of MambaVLM framework.** It contains a visual long sequence (built with DINOv2 and SigLIP), a Mamba projector, and a LLM. We utilize the pre-trained Mamba-2.8B and Vicuna-7B as its language models.

Continuous-time SSMs need to be discretized to be integrated into deep models, and the discretization includes a timescale parameter $\mathbf{\Delta}$ to transform the continuous parameters $\mathbf{A}$, $\mathbf{B}$ to discrete parameters $\overline{\mathbf{A}}$, $\overline{\mathbf{B}}$. Typically, this transformation is achieved with zero-order hold (ZOH) method as follows:

$$\begin{aligned} \overline{\mathbf{A}} &= \exp\left(\mathbf{\Delta A}\right), \\ \overline{\mathbf{B}} &= \left(\mathbf{\Delta A}\right)^{-1}\left(\exp\left(\mathbf{\Delta A}\right) - \mathbf{I}\right). \end{aligned} \tag{2}$$

After the discretization, Eq. 1 can be reformulated with the step size $\mathbf{\Delta}$ as:

$$\begin{aligned} \mathrm{h}_t &= \overline{\mathbf{A}}\mathrm{h}_{t-1} + \overline{\mathbf{B}}\mathrm{x}_t, \\ y_t &= \mathbf{C}\mathrm{h}_t. \end{aligned} \tag{3}$$

Then, the models compute output through a global convolution:

$$\begin{aligned} \overline{\mathbf{K}} &= \left(\mathbf{C}\overline{\mathbf{B}}, \mathbf{C}\overline{\mathbf{A}}\overline{\mathbf{B}}, ..., \mathbf{C}\overline{\mathbf{A}}^{\mathbf{M}-1}\overline{\mathbf{B}}\right), \\ \mathbf{y} &= \mathbf{x} * \overline{\mathbf{K}}. \end{aligned} \tag{4}$$

where $\mathbf{M}$ is the length of the input sequence $\mathbf{x}$, and $\overline{\mathbf{K}} \in \mathbb{R}^{\mathbf{M}}$ is a structured convolutional kernel.

Based on the above structured SSM, the recent work Mamba (Gu & Dao, 2023) explored integrating a selective scan technique. Specifically, the matrices $\overline{\mathbf{B}}$, $\mathbf{C}$, and $\mathbf{\Delta}$ are derived from the input data and thus input-dependent. This change empowers the model with the capability to selectively propagate or discard information based on the sequential input tokens.

3.2 MambaVLM

As illustrated in Fig. 2, our framework MambaVLM mainly comprises three components: a visual long sequence constructed by dual vision encoders, a Mamba projector, and a Mamba LLM. We elaborate on the implementation details for each component below.

Visual long sequence. Following Cobra (Zhao et al., 2024), we utilize pretrained DINOv2 (Oquab et al., 2023) and SigLIP (Zhai et al., 2023) as our vision encoders to capture low-level spatial properties and the semantic properties simultaneously. However, different from Cobra that fuse the features of dual encoders along the channel dimension, we argue that this would greatly reduce the effective visual information and waste the rich representation of dual encoders. This is because the projector maps visual features to the dimensions of LLM's text features, so regardless of how many

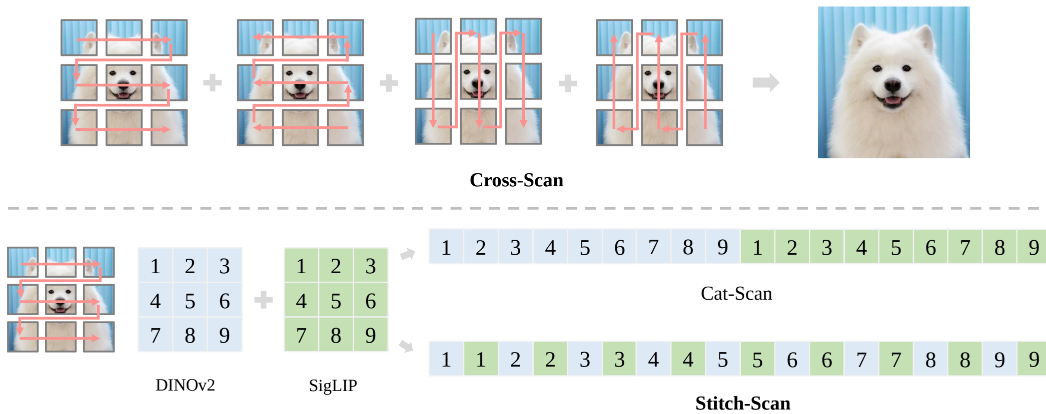

Figure 3: **Cross-Stitch scanning of MambaVLM.** Firstly, we employ four cross-scan orders to scan each of the two feature maps independently. Then, for each scanning order, we intermittently stitch-scan different feature maps to form an interpolated scanning sequence.

channels the visual tokens have, they are compressed to a fixed number of channels. Merely using a lightweight projector to conduct this mapping will inevitably result in the loss of visual information. Moreover, channel collapse (Woo et al., 2023) and redundancy (Yu et al., 2023; Chen et al., 2024) phenomena are common in neural networks, thus the effectiveness of fusing features in the channel dimension can be further undermined.

Considering the advantages of Mamba in handling long sequences and to alleviate the above issues, we propose to construct a visual long sequence to better utilize visual representations. Specifically, we concatenate the features of the two encoders along the sequence dimension, turning them into a longer sequence, thereby mitigating the visual information loss caused by feature collapse. Formally, given an image $\mathrm{X_v}$ as input, the vision encoder splits the image into $\mathrm{N_v}$ same-size patches. Both two vision encoders take the same patchified image as the input token sequence and we concat the output of two encoders along sequence dimension to get the visual representations $\mathrm{R_v} \in \mathbb{R}^{2\mathrm{N_v} \times \mathrm{D_v}}$:

$$\mathrm{R_v} = \mathrm{Concat}\left[\mathbf{f}_{\mathrm{DINOv2}}(\mathrm{X_v}); \mathbf{f}_{\mathrm{SigLIP}}(\mathrm{X_v})\right] \tag{5}$$

Mamba projector. Current mainstream MLLM frameworks (e.g., LLaVA (Liu et al., 2023c)) typically utilize a single MLP layer for vision-language alignment. Nevertheless, given that we have constructed visual long sequences to preserve richer visual information, we argue that a simple MLP layer may not be able to accomplish sufficient vision-language alignment and interaction of different visual features. Therefore, we devise a lightweight mamba projector to effectively promote feature interaction within visual long sequence and enhance vision-language alignment. The specific structure of the proposed mamba projector is illustrated in Fig. 2.

The core of our Mamba projector lies in its scanning mechanism. While scanning mechanism has been introduced into 2D images (Liu et al., 2024b; Zhu et al., 2024), the presence of multiple feature maps in visual long sequence renders previous scanning approaches inapplicable. To solve, we propose a cross-stitch scanning mechanism as shown in Fig. 3. Specifically, we first employ four cross-scan orders to scan each of the two feature maps independently. Then, for each scanning order, we devise two ways to intermittently scan different feature maps to form an interpolated scanning sequence. We refer to this two-stage scanning method as cross-cat scan and **cross-stitch scan**, respectively. We conduct ablations of these two scanning ways in Section 4.4 and use cross-stitch by default. After scanning, we get four interleaved sequences:

$$\begin{aligned} \mathrm{H_v} &= \mathrm{DWConv}(\mathrm{W_1} * \mathrm{R_v}) \\ \mathrm{H_{v1}}, \mathrm{H_{v2}}, \mathrm{H_{v3}}, \mathrm{H_{v4}} &= \mathrm{Cross\ Stitch}(\mathrm{H_v}) \end{aligned} \tag{6}$$

Then, all four sequences are fed into the mamba block separately and reshaped back into the original image patch order:

$$\mathrm{H_{vi}} = \mathrm{SSM}(\mathrm{H_{vi}}), \text{for } \mathrm{i} = 1, 2, 3, 4 \tag{7}$$

At last, all four sequences are merged to get a comprehensive representation $\mathrm{H_v}$:

$$\begin{aligned} \mathrm{H_v} &= \mathrm{Merge}(\mathrm{H_{v1}}, \mathrm{H_{v2}}, \mathrm{H_{v3}}, \mathrm{H_{v4}}) \\ \mathrm{H_v} &= \mathrm{R_v} + \mathrm{W_3} * (\mathrm{H_v} * (\mathrm{W_2} * \mathrm{R_v})) \end{aligned} \tag{8}$$

Here $\mathrm{W_1}, \mathrm{W_2}, \mathrm{W_3}$ are three independent linear layers, we omit layer norm for brevity.

Mamba LLM. We use the pre-trained Mamba LLM (Gu & Dao, 2023) $\mathbf{f}_{\mathrm{Mamba}}$ as the language model, which is a stack of 64 identical Mamba blocks. For a given text query $\mathrm{H_t}$, we first use the tokenizer and embedding module $\mathbf{f}_{\mathrm{T}}$ to map the text input into the embedding space. Then we concatenate the output of mamba projector and text embedding, feeding it into the Mamba LLM to get the final response $\mathrm{R} = \{\mathrm{r_i}\}_{\mathrm{i=1}}^{\mathrm{L}}$ in an auto-regressive manner:

$$\begin{aligned} \mathrm{R} &= \mathbf{f}_{\mathrm{Mamba}}\left(\mathrm{H_v}, \mathbf{f}_{\mathrm{T}}\left(\mathrm{H_t}\right)\right), \\ \mathrm{p}\left(\mathrm{R}|\mathrm{H_v}, \mathbf{f}_{\mathrm{T}}\left(\mathrm{H_t}\right)\right) &= \prod_{\mathrm{i=1}}^{\mathrm{L}} \mathrm{p}\left(\mathrm{r_i}|\mathrm{H_v}, \mathbf{f}_{\mathrm{T}}\left(\mathrm{H_t}\right), \mathrm{r_{<i}}\right). \end{aligned} \tag{9}$$

Finally, the output tokens R will be decoded to the response answer in natural language.

Note that our designs are highly coupled. Firstly, our framework constructs visual long sequences to preserve richer visual features. Then, in order to leverage the rich visual features we propose the Mamba projector to effectively promote the interaction of visual information and vision-language alignment, thus providing high-quality representations for the Mamba LLM to unleash its true power. We give thorough ablations in section 4.4.

4 Experiments

In this section, we first introduce the settings and training recipe of MambaVLM in Section 4.1. Then to evaluate MambaVLM, we conduct experiments with other methods on four open-ended visual question answering (VQA) datasets and three challenge datasets in Section 4.2& 4.3. In section 4.4, we conduct detailed ablation studies to validate the effectiveness of our proposed designs. Finally, we give a efficiency comparison in Section 4.5 and present some qualitative results to demonstrate the superiority of our approach in Section 4.6.

4.1 Settings

We ensemble DINOv2 and SigLIP features to construct visual long sequence, the input resolution for both encoders is 384×384. The Mamba projector is always randomly initialized. For the LLM backbone, we use the official Mamba-2.8B-SlimPj, and we also experiment with Vicuna-7B to further demonstrate our framework's effectiveness. We employ AdamW with a momentum of 0.9, a total batch size of 128, and a weight decay of 0.05 to optimize models. We train the MambaVLM-2.8B for 2 epochs and MambaVLM-7B for 1 epoch respectively, the initial learning rate is 2×10^5 with a warmup ratio 0.03. Experiments are conducted on 8 A800 GPUs. We use the Pytorch Fully Sharded Data Parallel (Zhao et al., 2023) framework to accelerate training. Training details can be found in Appendix A.

For MambaVLM-Mamba-2.8B, we use a combination of three datasets to train it: The 665K multi-modal instruct tuning dataset in LLaVA-1.5 (Liu et al., 2023c), the LVIS-Instruct-4V (Wang et al., 2023) dataset and the LRV-Instruct (Liu et al., 2023b) dataset. This combination results in a 1231K dataset, which is the same as that in Cobra. For MambaVLM-Vicuna-7B, We only use the 665K dataset to train it since we empirically find 665K is enough for MambaVLM to have competitive performances. A detailed pretraining dataset composition is provided in Appendix B.

4.2 Evaluation on Open-Ended VQA

For open-ended visual question answering, we evaluate MambaVLM on four datasets: TextVQA (Singh et al., 2019), GQA (Hudson & Manning, 2019), VQA-v2 (Goyal et al., 2017) and VizWiz (Gurari et al., 2018). Specifically, TextVQA evaluates the optical character recognition

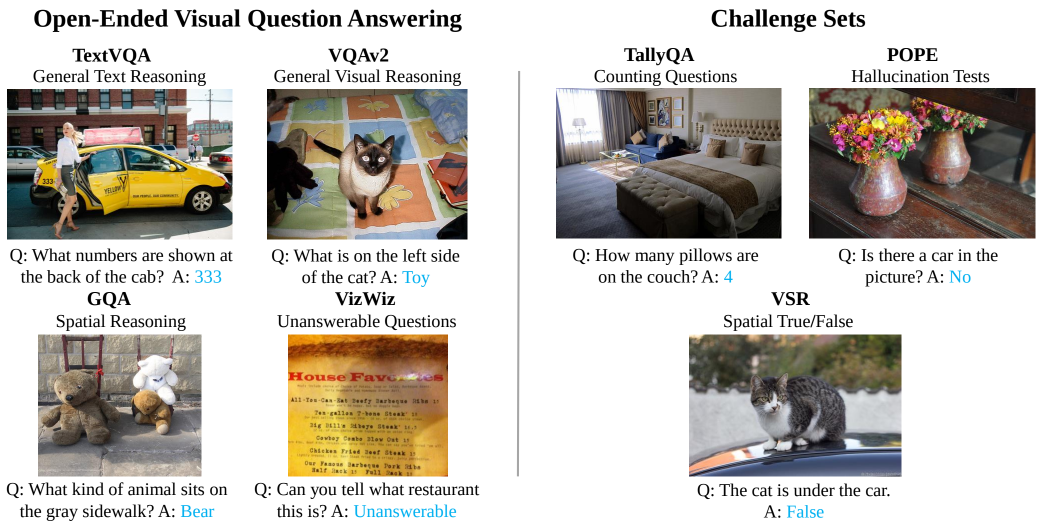

Figure 4: **Overview of evaluation benchmarks.** We evaluate MambaVLM across four open-ended VQA datasets and three challenge sets, giving us fine-grained assessment of our design choices.

(OCR) and the reasoning around text capacities; GQA assesses multi-step reasoning in real-world images; VQA-v2 and VizWiz both evaluate the general visual reasoning capacity while VizWiz has additional unanswerable questions. An overview of datasets is illustrated in Fig. 4.

As shown in Table 1, our MambaVLM has consistently strong performances across these benchmarks. For instance, our method outperforms Cobra by large margins: +4.2 gains on TextVQA, +0.4 gains on VQA-v2, +0.5 gains on VizWiz and +1.0 gains on GQA. Moreover, when scaling to larger LLM (i.e., Vicuna-7B), our framework still exhibits exceptional performance and data efficiency. In particular, trained with only 665K data, MambaVLM performs on par with 1.4B trained Qwen-VL and surpasses LLaVA-1.5 by significant margins (+4.4 gains on TextVQA, +2.1 gains on VQA-v2, and +1.1 gains on VizWiz), further demonstrating our framework's effectiveness.

Table 1: **Comparison with open-source VLM models on four open-ended VQA benchmarks.** Our MambaVLM has consistently strong performances across these benchmarks, surpassing strong baselines by large margins. *denotes using Mamba-2.8B-Zephyr, which is finetuned based on Mamba-2.8B thus a stronger LLM.

Method	LLM	Data	TextVQA	VQA^{v2}	VizWiz	GQA
OpenFlamingo	MPT-7B	2B	33.6	52.7	27.5	N/A
IDEFICS	LLaMA-7B	353M+1M	25.9	50.9	35.5	38.4
BLIP-2	Vicuna-13B	129M	42.5	41.0	N/A	41.0
MiniGPT-4	Vicuna-7B	5M+5K	N/A	N/A	N/A	32.2
Shikra	Vicuna-13B	600K+5.5M	N/A	77.4	N/A	N/A
Instruct-BLIP	Vicuna-7B	129M+1.2M	50.1	76.1	32.0	49.2
Qwen-VL	Qwen-7B	1.4B+50M	63.8	78.8	35.2	59.3
LLaVA-1.5	Vicuna-7B	558K+665K	58.2	76.5	54.2	61.6
MambaVLM	Vicuna-7B	665K	**62.6**	**78.6**	**55.3**	**61.8**
LLaVA-Phi	Phi-2.7B	558K+665K	48.6	71.4	35.9	N/A
MobileVLM	MobileLLaMA-2.7B	558K+665K	47.5	N/A	N/A	59.0
VL-Mamba	Mamba-2.8B	558K+665K	48.9	76.6	N/A	56.2
Cobra	Mamba-2.8B*	1231K	46.0	75.9	52.0	58.5
MambaVLM	Mamba-2.8B	1231K	**50.2**	**76.3**	**52.5**	**59.5**

Table 2: **Comparison with open-source VLM models on three challenge set benchmarks.** Our MambaVLM has consistently strong performances.

Method	LLM	Data	TallyQA	POPE	VSR
BLIP-2	Vicuna-13B	129M	N/A	85.3	N/A
Instruct-BLIP	Vicuna-7B	129M+1.2M	N/A	84.3	58.9
LLaVA-1.5	Vicuna-7B	558K+665K	62.1	86.6	59.6
MambaVLM	Vicuna-7B	665K	**66.6**	**87.9**	**68.1**
LLaVA-Phi	Phi-2.7B	558K+665K	N/A	85.0	N/A
MobileVLM	MobileLLaMA-2.7B	558K+665K	N/A	84.9	N/A
VL-Mamba	Mamba-2.8B	558K+665K	N/A	84.4	N/A
Cobra	Mamba-2.8B*	1231K	58.2	88.0	63.6
MambaVLM	Mamba-2.8B	1231K	**59.1**	**87.7**	**66.7**

4.3 Evaluation on Challenge Sets

To comprehensively assess MambaVLM's capabilities, we further evaluate it on three challenge sets: TallyQA (Acharya et al., 2019), POPE (Li et al., 2023b) and Visual Spatial Reasoning (VSR) (Liu et al., 2023a). These three datasets are all closed-set prediction tasks. In particular, TallyQA comprises questions that test MLLM's ability to count objects described in language with varying levels of complexity; POPE aims at evaluating object hallucinations, which is a binary classification task that prompts the model to answer whether an object exists or not; VSR provides a thorough assessment of the models to see if they can understand individual spatial relationships between diverse scenes.

The experimental results in Table 2 demonstrate that our MambaVLM has consistently powerful performance on these three datasets, which is not only applicable to Mamba LLM, but also can be extended to larger LLM. Specifically, we outperform two strong frameworks (i.e., Cobra and LLaVA-1.5 respectively) when equipped with different LLMs. We omit some of the methods that appear in Table 1 because they did not report results on these datasets in their papers.

4.4 Ablation Study

We conduct ablation experiments on the two core designs of our framework in Table 3: visual long sequence and mamba projector. We start by introducing the baseline. Our baseline's vision encoder is DINOv2 and SigLIP, concating their features in the channel dimension, which is the same as Cobra. It's projector is a simple MLP layer and LLM is the same Mamba-2.8B. The training data and recipe keep the same as that in Section 4.1.

Next, we extend the baseline to MambaVLM step by step. Firstly, we build the visual long sequence, then we replace the MLP with our mamba projector, and finally we ablate the scanning mechanism. Experiments demonstrate that our proposed cross-stitch scan results in the best performance. Note that all model variants in Table 3 use both DINOv2 and SigLIP as the vision encoders, so the effectiveness of our designs does not come from using more vision encoders. These ablations effectively backups the validity of our proposed designs.

Table 3: **Ablation studies on our framework.** We extend the baseline to MambaVLM step by step. Note that all variants use both DINOv2 and SigLIP as vision encoders. **Thus our gains do not come from using two vision encoders but from our tailored designs.**

Method	Long	Scan	TextVQA	VizWiz	VSR	Average
baseline	✖	N/A	47.1	51.4	62.6	53.7
+ long sequence	✔	N/A	48.0	51.3	65.3	54.9
++ mamba projector	✔	Cross-Cat	49.7	50.7	66.4	55.6
MambaVLM	✔	Cross-Stitch	**50.2**	**52.5**	**66.7**	**56.5**

Table 4: **Inference speed comparison.** We compare with two transformer-based MLLMs of the same parameter scale (3B). Note that increasing the number of input visual tokens typically result in greater inference burden. However, our method still holds significant speed advantage, indicating that our design tailored for Mamba does not vanish Mamba's speed merits.

Method	LLM	Visual Tokens	Output Tokens	**Speed (tokens/s)**
TinyLLaVA	Phi2-2.7B	576	256	39.64
MobileVLMv2	MobileLLaMA-2.7B	144	256	49.50
MambaVLM	Mamba-2.8B	1458	256	**131.07**

4.5 Efficiency Comparison

Our MambaVLM framework enjoys exceptional data and training efficiencies. Specifically, we measure the training time of MambaVLM and LLaVA-1.5 on the same machine (i.e., 8 NVIDIA A800 GPUs) and find that MambaVLM can save more than 40% of the training time as shown in Fig. 1. This is remarkable considering it outperforms LLaVA-1.5 consistently across seven evaluation benchmarks, further demonstrating the superiority of our framework.

We further evaluate the inference speed of MambaVLM. Specifically, we compare it with two transformer-based MLLMs of the same parameter scale (~3B). We evaluate them under the same setting (i.e., the same input image and the same text prompt). We set the number of output tokens to 256 for all models. The differences are the input visual token length and the LLM type (Mamba v.s. Transformer). As shown in Table 4, although MambaVLM has much more visual tokens, it still holds significant speed advantage over transformer-based MLLMs. This phenomenon is due to Mamba's linear complexity to token sequence length, so that constructing visual long sequence will not vanish Mamba's inference speed merits. Therefore, this experiment indicates that our visual long sequence design is tailored for Mamba-based MLLMs, which can effectively improve performance while incurring only minor side effects on speed.

4.6 Qualitative Result

In this section, we elaborately design some questions as case studies to exhibit the qualitative results of MambaVLM. As shown in Fig. 5, our method demonstrates exceptional and comprehensive performances in reasoning, hallucinations, counting and spatial perceptions.

- As shown in the first image, our method can not only accurately get how many plates of cake there are, but also reasonably deduce that the single piece of cake could have been served to himself by the person who prepared the cake.
- For the second image, we construct a common hallucination problem and then proceed to provide instruction to test whether the model overfits the training data. Cobra clearly overfits to the hallucination, while MambaVLM exhibits stronger instruction following ability.
- For the third image, we design a more complex counting problem. The model not only requires to count, but also needs to distinguish the size of elephants, and our method successfully address this problem.
- For the last image, there is a mixed problem of hallucination and spatial perception. Cobra fails at the basic hallucination level while our method not only identifies hallucination, but also provides accurate spatial positional relationships.

5 Conclusion

In this paper, we explore unleashing the true power of Mamba in MLLMs instead of merely using it as the LLM. We construct visual long sequences to provide stronger representation, and devise a novel cross-stitch scanning mechanism to facilitate visual information interaction and vision-language alignment. Built upon these designs, we propose a simple yet strong MLLM framework termed MambaVLM. Extensive experiments across multiple benchmarks demonstrate our method's effectiveness, suggesting the significant potential and promising prospects of Mamba in MLLMs.

In What might happened recently with the cake?

Cobra Recently, someone might have sliced the cake and served it to the guests, as there are three slices of cake on plates and a fork on the table. The cake appears to be a carrot cake with orange icing, and it is placed on a wooden table. The presence of the fork suggests that the guests are ready to enjoy the cake.

Ours Recently, someone might have sliced the cake and served it on plates, as there are two plates of cake and a fork present in the image. The cake appears to be a delicious, freshly baked dessert, which is likely to be enjoyed by the people who are about to consume it. The presence of the fork suggests that the cake is ready to be eaten, and the person who prepared it might has already served that piece to himself.

In If there are factual errors in the questions, point it out and proceed to answering the question. What's happening in the desert? Describe the picture.

Cobra There is no mention of a desert in the given information.

Ours There are no deserts visible in the image. The image features a city skyline with a beach on the water, and the ocean is visible in the background. The city is lit up at night, and there are buildings and palm trees in the foreground.

In How many adult elephants and how many baby elephants are in the picture?

Cobra There are two adult elephants and one baby elephant in the picture.

Ours There are three adult elephants and two baby elephants in the picture.

In Is the bicycle parked at the right of the dog?

Cobra The image has a bicycle parked at the left of the dog, not the right.

Ours The image has a boat, not a bicycle, and the dog is sitting in the boat.

Figure 5: **Qualitative Results.** We elaborately design some questions as case studies to exhibit the qualitative results of MambaVLM. Our method exhibits exceptional performances in reasoning, hallucinations, counting and spatial perceptions.

REFERENCES

Manoj Acharya, Kushal Kafle, and Christopher Kanan. Tallyqa: Answering complex counting questions. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 8076–8084, 2019.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

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

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.

Honghao Chen, Xiangxiang Chu, Yongjian Ren, Xin Zhao, and Kaiqi Huang. Pelk: Parameter-efficient large kernel convnets with peripheral convolution. *arXiv preprint arXiv:2403.07589*, 2024.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pp. 104–120. Springer, 2020.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6, 2023.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36, 2024.

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

Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, et al. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023.

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

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021a.

Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021b.

Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021c.

Albert Gu, Karan Goel, Ankit Gupta, and Christopher Ré. On the parameterization and initialization of diagonal state space models. *Advances in Neural Information Processing Systems*, 35:35971–35983, 2022.

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

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*, pp. 6700–6709, 2019.

Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic vlms: Investigating the design space of visually-conditioned language models. *arXiv preprint arXiv:2402.07865*, 2024.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128–3137, 2015.

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

Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International conference on machine learning*, pp. 5583–5594. PMLR, 2021.

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.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pp. 12888–12900. PMLR, 2022.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023a.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pp. 121–137. Springer, 2020.

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*, 2023b.

Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. *arXiv preprint arXiv:2312.07533*, 2023.

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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023a.

Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Mitigating hallucination in large multi-modal models via robust instruction tuning. In *The Twelfth International Conference on Learning Representations*, 2023b.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023c.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024a.

Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024b.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pp. 3195–3204, 2019.

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

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Yanyuan Qiao, Chaorui Deng, and Qi Wu. Referring expression comprehension: A survey of methods and datasets. *IEEE Transactions on Multimedia*, 23:4426–4440, 2020.

Yanyuan Qiao, Zheng Yu, Longteng Guo, Sihan Chen, Zijia Zhao, Mingzhen Sun, Qi Wu, and Jing Liu. Vl-mamba: Exploring state space models for multimodal learning. *arXiv preprint arXiv:2403.13600*, 2024.

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*, pp. 8748–8763. PMLR, 2021.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European Conference on Computer Vision*, pp. 146–162. Springer, 2022a.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European Conference on Computer Vision*, pp. 146–162. Springer, 2022b.

Teams ShareGPT. Sharegpt: Share your wildest chatgpt conversations with one click, 2023.

Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. Textcaps: a dataset for image captioning with reading comprehension. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pp. 742–758. Springer, 2020.

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*, pp. 8317–8326, 2019.

Jimmy TH Smith, Andrew Warrington, and Scott W Linderman. Simplified state space layers for sequence modeling. *arXiv preprint arXiv:2208.04933*, 2022.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pp. 10347–10357. PMLR, 2021.

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

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3156–3164, 2015.

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.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 568–578, 2021.

Fei Wei, Xinyu Zhang, Ailing Zhang, Bo Zhang, and Xiangxiang Chu. Lenna: Language enhanced reasoning detection assistant. *arXiv preprint arXiv:2312.02433*, 2023.

Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16133–16142, 2023.

Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1307–1315, 2018.

Weihao Yu, Pan Zhou, Shuicheng Yan, and Xinchao Wang. Inceptionnext: When inception meets convnext. *arXiv preprint arXiv:2303.16900*, 2023.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11975–11986, 2023.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5579–5588, 2021.

Han Zhao, Min Zhang, Wei Zhao, Pengxiang Ding, Siteng Huang, and Donglin Wang. Cobra: Extending mamba to multi-modal large language model for efficient inference. *arXiv preprint arXiv:2403.14520*, 2024.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, et al. Pytorch fsdp: experiences on scaling fully sharded data parallel. *arXiv preprint arXiv:2304.11277*, 2023.

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

Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024.

A TRAINING CONFIGURATION

We list the detailed training configuration and recipe for Cobra in Table 5. For MambaVLM-7B, since there is no publicly available Mamba-7B model, we utilize the widely used Vicuna-7B to validate the advantages of our framework when extended to larger LLMs. Note that in addition to the difference in the LLM for MambaVLM-2.8B&MambaVLM-7B, the data used and the number of training epochs is also different.

Table 5: Training configuration and recipe of MambaVLM.

Configuration	MambaVLM-2.8B	MambaVLM-7B
Vision Encoder	DINOv2 + SigLIP	
Projector init	Random	
Image resolution	384×384	
Global batch size	128	
Optimizer	AdamW	
LR schedule	Cosine decay	
Learning Rate	2e-5	
Weight decay	0.1	
Warmup ratio	0.3	
LLM init	Mamba-2.8B-Slimpj	Vicuna-1.5-7B
Data	1231K	665K
Epochs	2	1

B PRETRAINING DATASET COMPOSITION

We use The 665K multi-modal instruct tuning dataset in LLaVA-1.5 Liu et al. (2023c), the LVIS-Instruct-4V Wang et al. (2023) dataset and the LRV-Instruct Liu et al. (2023b) dataset. We list the detailed example sources of the 665K instrut-tuning dataset as follows:

LLaVa Synthetic Data (158K). This dataset is a conversation, fine-grained description, and question-and-answer dataset synthesized by prompting GTP-4 Achiam et al. (2023), with image caption and object bounding box from COCO Lin et al. (2014). This dataset is explicitly generated in instruction form.

Standard VQA Data (224K). This dataset is a combination of visual question-answering datasets including VQAv2 Goyal et al. (2017), GQA Hudson & Manning (2019), OK-VQA Marino et al. (2019), and OCR-VQA Mishra et al. (2019). The questions cover many aspects such as general question answering, spatial and compositional reasoning, external knowledge-based and text-based reasonings.

Multiple Choice VQA Data (50K). This dataset is an external knowledge-based multiple choice QA task sourced from A-OKVQA Schwenk et al. (2022a). The model is required to output the corresponding option letter.

Captioning Data (22K). This dataset is an image caption dataset sourced from TextCaps Sidorov et al. (2020).

Referring Expression Data (116K). This dataset comprises referring expression grounding (bounding box prediction) and region captioning data sourced from RefCOCO Kazemzadeh et al. (2014). For bounding box prediction (localization), the model is asked to output normalized bounding box coordinates in a natural language manner.

ShareGPT (Language-Only) (40K). This dataset consists of user-uploaded conversations generated by ChatGPT from ShareGPT ShareGPT (2023). This dataset is also explicitly generated in instruction form.