arXiv:2502.06788v2 [cs.CV] 24 Jul 2025

# EVEv2: Improved Baselines for Encoder-Free Vision-Language Models

Haiwen Diao[1,2*] Xiaotong Li[3,2*] Yufeng Cui[2*] Yueze Wang[2*]
Haoge Deng[4,2] Ting Pan[5,2] Wenxuan Wang[6,5,2] Huchuan Lu[1†] Xinlong Wang[2†]
[1]DLUT [2]BAAI [3]PKU [4]BUPT [5]UCAS [6]CASIA

## Abstract

*Existing encoder-free vision-language models (VLMs) are rapidly narrowing the performance gap with their encoder-based counterparts, highlighting the promising potential for unified multimodal systems with structural simplicity and efficient deployment. We systematically clarify the performance gap between VLMs using pre-trained vision encoders, discrete tokenizers, and minimalist visual layers from scratch, deeply excavating the under-examined characteristics of encoder-free VLMs. We develop efficient strategies for encoder-free VLMs that rival mainstream encoder-based ones. After an in-depth investigation, we launch* ***EVEv2.0****, a new and improved family of encoder-free VLMs. We show that: (i) Properly decomposing and hierarchically associating vision and language within a unified model reduces interference between modalities. (ii) A well-designed training strategy enables effective optimization for encoder-free VLMs. Through extensive evaluation, our* ***EVEv2.0*** *represents a thorough study for developing a decoder-only architecture across modalities, demonstrating superior data efficiency and strong vision-reasoning capability. Code is publicly available at: https://github.com/baaivision/EVE.*

## 1. Introduction

With the recent rapid advancements in the large language models (LLMs) [8, 9, 59, 77, 92] and large vision models (LVMs) [18, 22, 24, 44, 68], vision-language models (VLMs) [2, 4, 20, 21, 55, 93] have made remarkable strides, demonstrating impressive capabilities in multi-modal understanding and reasoning applications. As illustrated in Figure 1(1), typical practice adopts pre-trained vision encoders to extract visual semantics, which are then translated into the text embedding space as "Foreign" language inputs for subsequent LLMs (*e.g.*, BLIP [43] and LLaVA [51]). In contrast, another representative branch transforms visual features from vision encoders' last layer across each layer of LLMs through cross-attention modules, like Flamingo [1] and LLaMA-3.2V [72]. Thanks to well-aligned representations across modalities through large-scale contrastive learning [61, 66, 98], these studies can achieve promising performance and strong multi-modality capability after joint training. However, the inductive biases during visual pre-training, *e.g.*, image resolution, aspect ratio, and semantic priors, limit the flexibility and applicability of the visual learning in diverse real-world scenarios [74, 75, 90].

Unlike them, Fuyu [6] takes an early step to explore monolithic VLMs via removing pre-trained visual encoders, while EVE [19] first pioneers a transparent, efficient, and practical path for advancing the encoder-free VLM direction. The subsequent PaliGemma [7] constructs an encoder-free VLM that shows strong scaling efficiency while progressively approaching its encoder-based counterpart. With extensive training data and computing resources, Mono-InternVL [57] narrows the gap and matches the performance of InternVL1.5 [15], starting with the same LLM capabilities.

However, constructing encoder-free VLMs remains challenging, particularly in learning vision perception from scratch and reducing vision-language interference within a unified model. To date, three solutions have been put forward: (i) Visual feature supervision [19]; (ii) Incremental training recipes [13, 19, 57]; (iii) Mixture-of-Expert (MoE) detachment [49, 57] in Figure 1(2c). Nevertheless, we empirically discover that visual supervision can be substituted by large-scale, high-quality image-text datasets annotated by a powerful captioning engine. During training, properly merging language and multimodal data helps mitigate knowledge-forgetting issues, while pressuring the development of multimodal capabilities. From the view of VLM structure, decoupling partial visual functions from the unified model through a MoE design [5, 41, 82] aids in relieving vision-language interference to some extent. However, we discover significant weight shifts across various network layers between the VLMs and the original LLMs, revealing the insufficiency of the current decoupling degree. Notably, we further exploit a re-parameterize architecture in Figure 1(2b) for seamless LLM adaptation. Although yielding better gains over prototype EVE in Figure 2(2a), it does not completely resolve the representational conflicts across modalities.

*Equal contribution. † Correspondence to *wangxinlong@baai.ac.cn*.

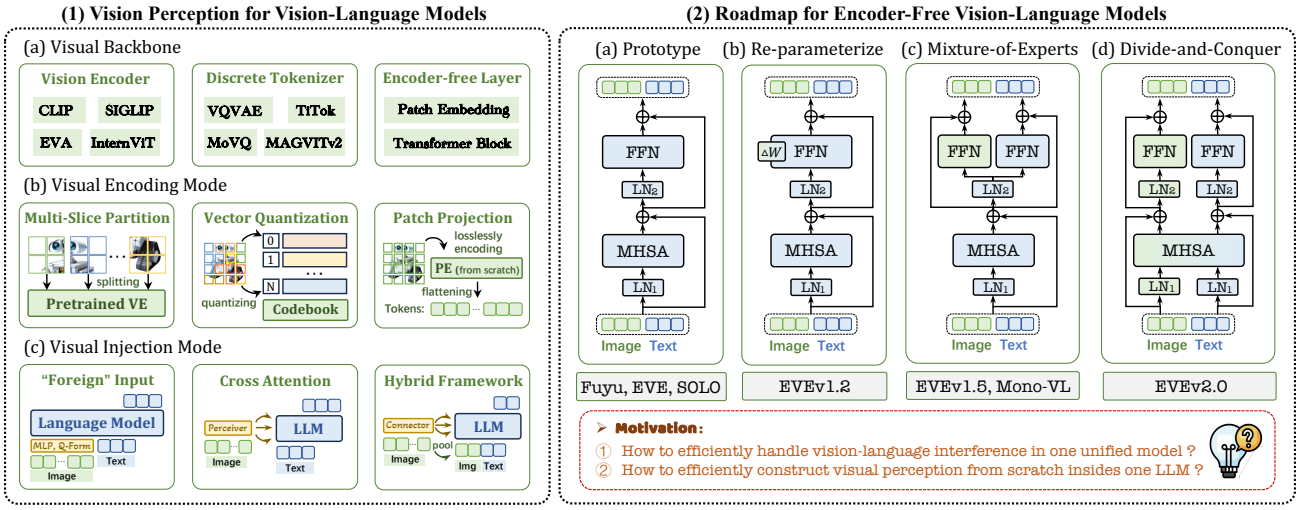



Figure 1. Overview of (1) diverse vision construction inside existing VLMs and (2) potential architecture variants of Encoder-Free VLMs.

From the above observations, we launch **EVEv2.0**, a new and improved baseline for encoder-free VLMs. **EVEv2.0** completely disentangles overall components and introduces modality-wise sparsity into one unified decoder-only backbone in Figure 1(2d). Such a Divide-and-Conquer architecture maximizes scaling efficiency in building visual perception while minimizing the negative influence on the LLM itself. Besides, using an enhanced caption engine, we construct an effective and practical route for monolithic VLM research that facilitates data-scaling efficiency and stably transfers increasingly stronger LLMs into the encoder-free VLMs. With 100M publicly available data, **EVEv2.0** outperforms encoder-free counterparts, and continually approaches encoder-based competitors of similar capacity across diverse vision-language benchmarks. Our **EVEv2.0** unveils valuable insights for developing scalable, native, and next-gen VLMs, paving a transparent roadmap for future research supported by larger training data and computational resources.

## 2. Related Work

**Encoder-based VLMs.** Encoder-based methods have become the dominant approach in vision-language models, widely adopted in commercial products, *e.g.* GPT-4V [59], Claude 3.5 [2], and Gemini [70], as well as in the open-source projects like LLaVA series [40, 50–52], Qwen-VL series [4, 81], InternVL series [14, 15], BLIP series [42, 43, 91], and EMU series [65, 67]. They benefit from the pre-trained knowledge from visual encoders [10, 61, 66, 98] and LLMs [3, 9, 16, 71, 73, 76, 77, 92], successfully building modular VLMs for various real-world applications. Among them, most studies [12, 46, 48, 90, 94, 95] directly translate vision representations into the input space of LLMs. In contrast, another type of research [1, 27, 30, 72, 101] introduces the cross-attention module for integrating visual and language information layer-by-layer. Despite achieving strong performance gains, it may be insufficient to simply map visual information into the input space of LLMs [50, 95] or connect the same visual features across different representational levels of the LLM [30, 33], given the heterogeneous characteristics between vision and language. Besides, these modular VLMs face challenges in further development due to the strong inductive biases in pre-training visual encoding patterns, complex infrastructure requirements, and scaling laws necessary to balance various separate components.

**Encoder-free VLMs.** Another visual processing strategy is discrete visual tokenization [23, 64, 78], which translates the image into a sequence of discrete tokens and is widely used in various multi-modal understanding and generation approaches [69, 83, 84, 99]. However, the discretization inevitably results in lossy visual information and weakens in extracting semantic contents, which in turn hinders fine-grained visual understanding and reasoning [85, 88]. Therefore, recent studies [45, 60, 86, 89] introduce semantic constraints in the visual tokenizer for both high-level semantic and fine-grained visual representations. Compared to their highly-compressed features in the discrete-value space, encoder-free VLMs [6, 13, 19] have emerged as a promising architecture for lossless visual encoding, efficient data scaling, and end-to-end multimodal processing. Specially, EVE [19] pioneers an efficient and transparent path for monolithic VLMs, with its data-scaling efficiency preliminarily validated by PaliGemma [7]. Impressively, Mono-InternVL [57] bridges the performance gap with its modular counterpart [15] of the same LLM capacity, using adequate data. We emphasize that limited by current training data and device resources, EVEv2.0 does not aim for state-of-the-art performance but instead focuses on revealing the most efficient route for encoder-free VLMs from scratch.

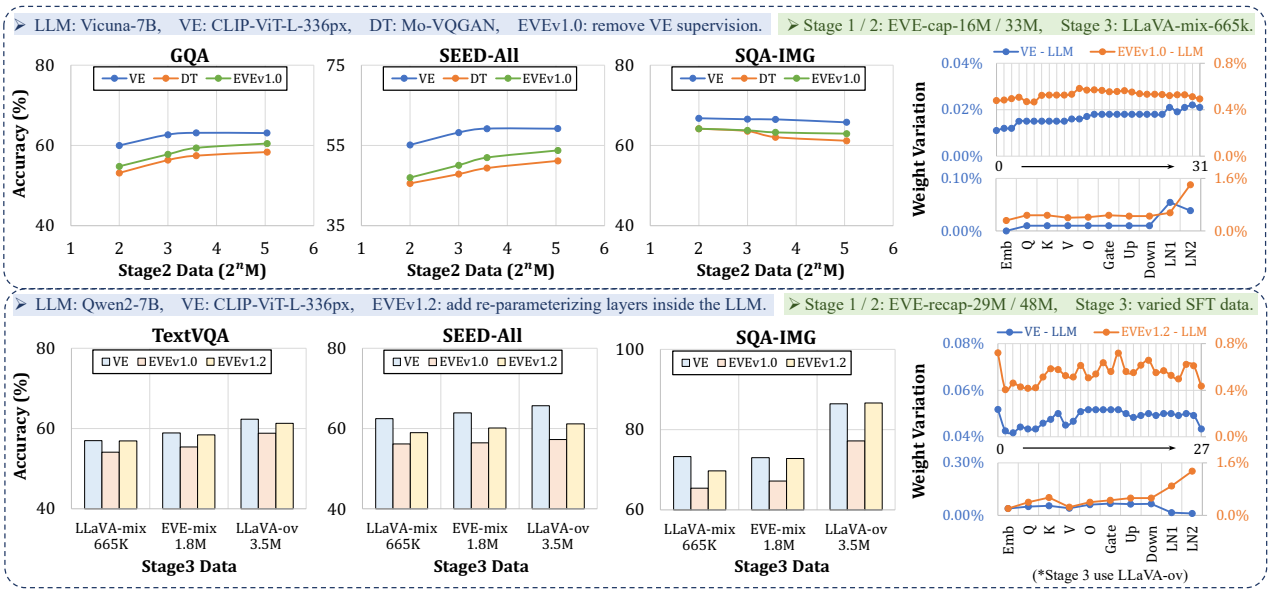



Figure 2. Preliminary scaling efficiency analyses during pre-training or fine-tuning across various VLMs. (More details in the Appendix). Notably, VE / DT / EVE apply varying image downsampling rates ($14^2$ / $8^2$ / $32^2$). For fairness, we choose different resolutions that yield relatively balanced token counts of 576 / 1024 / 625 tokens per image. Besides, we quantify weight changes between LLMs and VLMs by averaging absolute value variation within specific layer number or type. We report GQA [32], SEED [28], and TextVQA [63] for in-domain, open-domain, and OCR-related validation. *Note that SQA [56] involves text-related knowledge tasks susceptible to LLM's forgetting issue.*

# 3. Methodology

## 3.1. Preliminary

**Settings.** Building on [19, 50, 83], we conduct two pilot experiments in Figure 2. **Exp.(i):** we combine Vicuna-7B [16] with vision encoder (*VE*: CLIP-ViT-L-336px [61]), discrete tokenizer (*DT*: Mo-VQGAN [102]), or one single transformer block [79] (EVEv1.0: w/o VE supervision). We first train the projector, vision vocabulary, or vision block using EVE-cap-16M [19] caption data, followed by EVE-cap-33M [19] to update only vision encoder for *VE* (work best), or extra LLM weights for *DT* and EVEv1.0. Finally, we update all layers except for discrete tokenizers during fine-tuning. **Exp.(ii):** we use stronger Qwen2-7B [3] as the LLM for *VE*, EVEv1.0, and EVEv1.2. Using 29M image-text pairs, we initially train the projector for *VE* or all visual-related layers for EVE, with further updates to EVE's LLM layers via an extra 48M data. We then import varying-scale instruction data [19, 40, 50] to train the entire network.

**Finding (1):** *Performance gap between various vision encoding modes.* Exp.(i) shows that initially, *VE* performs better than *DT* and EVEv1.0 in visual understanding due to much larger image-text pretraining datasets (400M) and already alignment space between vision-language embeddings. Despite building visual recognition from scratch, EVEv1.0 demonstrates strong scaling properties, progressively closing the performance gap with *VE* as the data scale increases. Subsequent studies [7, 57] have proved that, with sufficient data, they can achieve comparable performance. Notably, *DT* maps visual information into a discrete space through quantization, hampering effective visual perception and weakening vision-language association via an image reconstruction objective. This ultimately results in subpar performance, even at larger data scales, leaving *DT* less competitive overall.

**Finding (2):** *Challenges towards multimodal interference and smooth transition.* While mixing text-only and multi-modal data can alleviate this issue, it slows down the development of multi-modal understanding [55]. Exp.(ii) shows that compared to EVEv1.0, EVEv1.2 approaches *VE* in knowledge-related SQA-IMG metric, slightly mitigating catastrophic linguistic forgetting in LLMs. Hence, we explore potential architectures and training strategies targeting LLM adaptation and multimodal interference. EVEv1.2 with re-parameterization design for each feed-forward weight helps smooth the transition from LLMs to VLMs, while EVEv1.5 with MoE design helps decouple vision-language encoding heterogeneity. However, after comparing LLMs and VLMs, we observe that *VE* only requires minor adjustments to the LLM, whereas EVE necessitates extensive updates for similar capabilities in Figure 2. Besides, Layer normalization emerges as the most impacted module, indicating that EVEv1.2 and EVEv1.5 struggle to efficiently construct visual representations and multi-modal alignments when attention and normalization layers in LLM are frozen. This dependency on LLM's pre-training paradigm limits the full potential for learning visual perception from scratch.

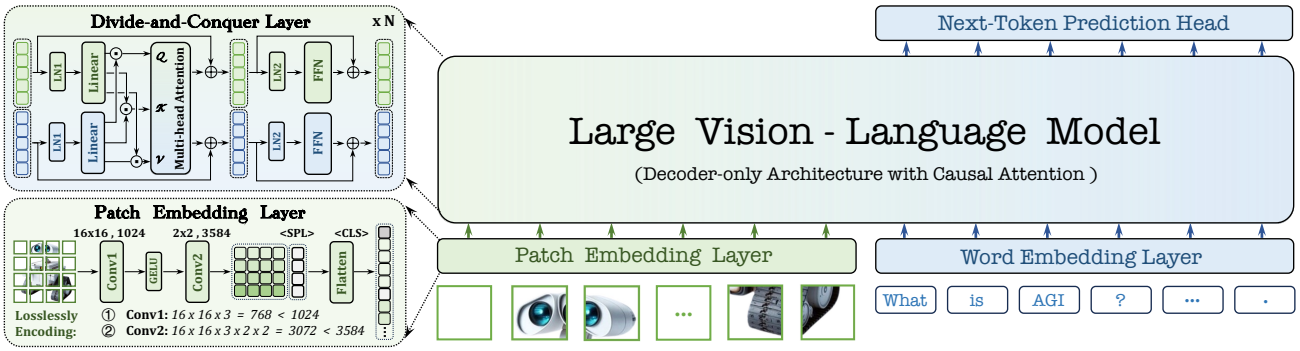



Figure 3. Overview of our proposed EVEv2.0 framework. We first adopt a patch embedding layer to encode images losslessly, and then concatenate visual and textual tokens into a unified decoder-only vision-language model. Here, it extends the standard autoregressive transformer by incorporating modality-specific weights for each multi-head self-attention layer, feed-forward layer, and layer normalization.

### 3.2. Model Architecture

Preliminary studies indicate that earlier EVE variants struggle to fully harness visual potential due to cross-modal interference within the pre-trained LLM distribution. To overcome this, we transform a dense transformer into a fully sparse, decoder-only architecture, guided by modality-aware routers in Figure 3. This approach yields a heterogeneous, modality-mixing model that retains the computational structure and FLOP count of its dense transformer counterpart.

**Visual and Textual Encoding**. For visual embeddings, we construct a minimalist patch embedding layer from scratch, eliminating strong inductive bias from pre-trained vision encoders in abstracting visual content. Given an image input $I \in \mathbb{R}^{H \times W \times 3}$, we first employ a Convolution layer (Conv1), followed by a Gaussian Error Linear Unit (GELU) activation function. After obtaining the resulting 2-D feature map, we then adopt another Convolution layer (Conv2) to flexibly control computational complexity as follows:

$$x_v = \text{Conv2}(\text{GELU}(\text{Conv1}(I))), \tag{1}$$

where Conv1 and Conv2 denote two convolutional layers with strides of 16 and 2, and output dimensions of 1024 and 3584, respectively. Thus, each pixel area (16×16×3×2×2 = 3072) is encoded into a 3584-dimensional feature, ensuring uncompressed and lossless information entropy. Besides, large-kernel linear projections (*e.g.*, 30×30) are hard to optimize, while 16×16 kernel quadruples visual tokens and increases overhead. Hence, two convolutional layers balance efficiency and accuracy. Additionally, two special learnable tokens serve as the prompts for the image start and line feed. The class token $<\text{CLS}>$ is appended at the beginning of the image sequence, while the split tokens $<\text{SPL}>$ are inserted after each row of image tokens for indicators. This patch embedding layer can support arbitrary-ratio images with up to about 2.5M pixels, *i.e.*, 2.5K patch tokens. Afterward, we adopt the text tokenizer from Qwen2.5 [73] to encode text $T$ into token embeddings $x_t$ with a dimension of 3584.

**Divide-and-Conquer Design.** Building on prior analyses, we propose explicitly decoupling key modules by introducing modality-aware components, including separate attention matrices (query, key, and value), normalization layers, and feed-forward modules, each with distinct parameters. Given the token sequence $x = (x_1, ..., x_n)$ where $x_i$ belongs to one specific modality $u_i \in \{v, t\}$, we perform a multi-head self-attention (ATTN) across all modalities, modeling cross-modal relationships in a unified feature space:

$$\begin{aligned} \text{ATTN}(x; \{\theta^u_{\text{attn}}\}) &= \left(\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V\right)W^{u_i}_O, \\ Q_i = x_i W^{u_i}_Q, K_i &= x_i W^{u_i}_K, V_i = x_i W^{u_i}_V, \end{aligned} \tag{2}$$

where modality-specific query, key, and value are derived from their respective attention weight matrices $W^{u_i}, u_i \in \{v, t\}$. The interaction process is performed across modalities, *i.e.*, visual $x_v$ and textual $x_t$ sets. Inspired by Figure 2, the significant quantified weight changes highlight the importance of decoupling the LayerNorm (LN) and Feed-Forward (FFN) layers; otherwise, they cause representational interference, limiting mutual capacities and capabilities. After fully decoupling the architecture, the overall operations within the Transformer block are defined as follows:

$$\begin{aligned} h &= x + \text{ATTN}(\text{LN1}(x; \theta^u_{\text{ln1}}); \{\theta^u_{\text{attn}}\}), \\ x' &= h + \text{FFN}(\text{LN2}(x; \theta^u_{\text{ln2}}); \theta^u_{\text{ffn}}). \end{aligned} \tag{3}$$

Compared with earlier EVE variants, EVEv2.0 employs a comprehensive decomposition with modality-specific components. This minimizes interference in the representation space by fully unbinding each layer and processing token sets separately for each modality. The structural decomposition supports efficient vision-perception training from scratch while retaining pre-trained knowledge by freezing off-the-shelf LLMs during pretraining. This also allows independent single-modality encoding and cross-modality correspondence across different layers simultaneously, enabling flexible modeling patterns for understanding and reasoning.

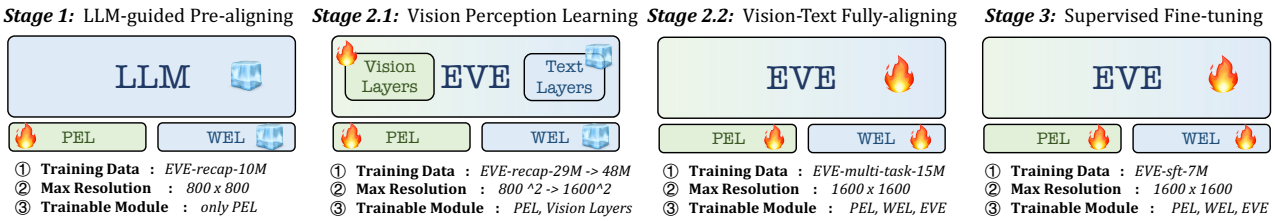



Figure 4. Overview of training procedure. PEL/WEL denotes patch/word embedding layer. We begin by training the patch embedding layer to establish initial alignment across modalities. Afterward, we only update vision layers within the LLM to enhance visual perception progressively. Notably, we gradually increase the image resolutions from 800×800 to 1600×1600 and keep the original image aspect ratio. Finally, we train the entire model via QA and instruction data to strengthen cross-modality correspondence and complex understanding.

### 3.3. Training Procedure

We divide the training process into four sequential stages in Figure 4. The training data consists of publicly available image datasets, along with diverse question-answering (QA) datasets and multimodal dialogue data in Table 1.

**DenseFusion++.** Developing strong visual perception requires high-quality, hyper-detailed image-text training data. Hence, we propose an efficient and low-budget engine based on LLaVA-1.6 (7B) [52] for scaling up synthetic data. Specifically, it leverages multiple vision experts (*e.g.*, Tag, Detection, OCR, ...) and learns the fusion strategy from GPT-4V, greatly facilitating data-scaling efficiency for native VLMs with high-quality annotations, not just distilling LLaVA. We empirically validate the superior training efficiency achieved by using intensive annotations from DenseFusion++ during pre-training. It consistently outperforms original web-sourced captions or those generated by lower-quality caption engines. We will release the code and weight of the caption engine for better interpretation and further exploration.

**LLM-guided Pre-aligning.** Following [19], we freeze the LLM weights and train only the patch embedding layer to prevent model collapse and accelerate convergence in subsequent stages. Using publicly available web data, we filter 44M image samples from Datacomp [26] recaptioned with our captioning engine. For training, we utilize a subset of 10M image-text pairs, dubbed *EVE-recap-10M*, and optimize with cross-entropy (CE) loss. Our experiments suggest that more extensive training at this stage is beneficial for training stability, especially considering stronger LLMs.

**Vision Perception Learning.** In this stage, we initialize the vision layers inside the LLM by loading LLM weights, where only the patch embedding and vision layers are trainable, while the Qwen2.5 [73] model remains frozen during training. This strategy enables efficient learning of visual representations through pre-training on large-scale synthetic data, without compromising the knowledge encoded in the pre-trained LLM. Besides, we carefully partition the training data into a progressive, coarse-to-fine visual learning process. For training, we first introduce 29M re-captioning data from Datacomp [26] supervised by CE loss with a maximum of 640K image pixels (*i.e.*, 625 patch tokens). Afterward, we increase the maximum image resolution to 2.5M pixels (*i.e.*, 2.5K patch tokens) on an expanded dataset comprising 15M Datacomp [26], 15M LAION [62], 11M SA-1B [36], and 7M OpenImages [37], dubbed *EVE-recap-48M*.

Table 1. Details of training datasets across all stages. Note that we construct DenseFusion++ to re-caption web-scale image-text data.

| Stage | Dataset | #Num | Total |
|---|---|---|---|
| 1 / 2.1 | Datacomp [26]<br>LAION [62]<br>SA-1B [36]<br>OpenImages [37] | 44M<br>15M<br>11M<br>7M | 77M |
| 2.2 | Infinity-MM-GeneralQA [29] | 15M | 15M |
| 3 | LLaVA-onevision [40]<br>Infinity-MM-instruct [29] | 3.5M<br>3.8M | 7.3M |

**Vision-Text Fully-aligning.** After establishing an initial alignment across modalities, we update the entire architecture to further improve image-text associations via the same loss functions. To facilitate this, we curate a diverse dataset of 15M samples from Infinity-MM general visual instruction [29], including chart comprehension, OCR recognition, mathematical reasoning, and *etc*. This dataset, named *EVE-multi-task-15M*, enhances visual perception and vision-language alignment, equipping EVE with the foundational capabilities to handle various multimodal tasks.

**Supervised Fine-tuning.** During the SFT stage, we further enhance EVE's ability to understand complex linguistic instructions and multifarious dialogue patterns, which are crucial for real-world applications. Here, we optimize the overall network layers on a diverse set of high-quality, multi-source instruction datasets, namely *EVE-sft-7M*, including LLaVA-onevision [40] and partial Infinity-MM-instruct [29]. Note that Stages 2.2 and 3 can be merged if large, balanced, and high-quality instruction data is available. We separate them to handle diverse but uneven (Stage 2.2) and balanced (Stage 3) data to achieve consistent performance.

Table 2. Comparison with existing vision-language models on various vision-language benchmarks, including MMMU [97]; $MMB^{en}$: MMBench-EN [53]; $SEED^{I}$: SEEDBench-IMG [39]; MMV: MMVet [96]; MME [25]; POPE [47]; GQA [32]; $SQA^{I}$: ScienceQA-IMG [56]; TVQA: TextVQA [63]; CQA: ChartQA [58]; AI2D [34]; RWQA: RealWorldQA [87]; OCRB: OCRBench [54]. Note that #A-Param denotes the number of activated parameters; #Data represents the pre-training / fine-tuning data volume; #Vtoken indicates the maximum image patch tokens. For MME, we sum up the perception and cognition scores. The best two results are marked in **bold** and underline.

| Method | #A-Param | #Data | #Vtoken | MMMU | $MMB^{en}$ | $SEED^{I}$ | MMV | MME | POPE | GQA | $SQA^{I}$ | TQA | CQA | AI2D | RWQA | OCRB |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Encoder-based Vision-Language Models:* | | | | | | | | | | | | | | | | |
| InternVL-1.5 | 2.2B | – / – | 3328 | 34.6 | 70.9 | 69.8 | 39.3 | 1902 | **88.3** | 61.6 | 84.9 | 70.5 | 74.8 | 69.8 | 57.9 | 654 |
| QwenVL-Chat | 7B | 7.2B / 50M | 256 | 35.9 | 60.6 | 58.2 | – | 1848 | – | 57.5 | 68.2 | 61.5 | 49.8 | 45.9 | 49.3 | 488 |
| LLaVA-1.5 | 7B | 0.4B+ / 665K | 576 | 35.3 | 64.3 | 64.3 | 30.5 | 1859 | 85.9 | 62.0 | 66.8 | 46.1 | 18.2 | 54.8 | 54.8 | 318 |
| LLaVA-1.6 | 7B | 0.4B+ / 760K | 2880 | 35.1 | 67.4 | 64.7 | 43.9 | 1842 | 86.4 | 64.2 | 70.2 | 64.9 | 54.8 | 66.6 | 57.8 | 532 |
| Cambrian | 7B | 10B+ / 7M | 576 | 42.7 | 75.9 | 74.7 | – | – | – | **64.6** | 80.4 | **71.7** | 73.3 | 73.0 | 64.2 | 614 |
| LLaVA-OV | 7B | 10B+ / 3.2M | 7290 | **47.3** | **81.7** | **74.8** | **58.8** | **1998** | – | – | **96.6** | – | **78.8** | **81.6** | **65.5** | **697** |
| *Encoder-free Vision-Language Models:* | | | | | | | | | | | | | | | | |
| Fuyu | 8B | – / – | – | 27.9 | 10.7 | 59.3 | 21.4 | – | 84.0 | – | 56.8 | – | – | 64.5 | 43.7 | 366 |
| Chameleon | 7B | 1.4B+ / 1.8M | 1024 | 25.4 | 31.1 | 30.6 | 8.3 | 170 | 19.4 | – | 47.2 | 4.8 | 2.9 | 46.0 | 39.0 | 7.0 |
| EVE | 7B | 33M / 1.8M | 2304 | 32.6 | 52.3 | 64.6 | 25.7 | 1628 | 85.0 | 62.6 | 64.9 | 56.8 | 59.1 | 61.0 | – | 398 |
| SOLO | 8B | 43.7M / 2M | 1024 | – | **67.7** | 64.4 | 30.4 | 1260 | 78.6 | – | 73.3 | 25.0 | – | 61.4 | 44.7 | 126 |
| Mono-InternVL | 1.8B | 1.3B / 7M | 6400 | 33.7 | 65.5 | 67.4 | 40.1 | **1875** | – | 59.5 | 93.6 | **72.6** | 73.7 | 68.6 | – | **767** |
| Emu3 | 8B | – / – | 16K | 31.6 | 58.5 | 68.2 | 37.2 | – | 85.2 | 60.3 | 89.2 | 64.7 | 68.6 | 70.0 | 57.4 | 687 |
| **EVEv2.0** | 7B | 92M / 7.3M | 2500 | **39.3** | 66.3 | **71.4** | **45.0** | 1709 | **87.6** | **62.9** | **96.2** | 71.1 | **73.9** | **74.8** | **62.4** | 702 |

# 4. Experiments

## 4.1. Training Settings

**Data Preparation.** All the training data is collected from publicly accessible sources to ensure reproducibility. *(1) Image-Text Datasets.* We follow the pre-processing pipeline outlined in [19] to process SA-1B [36], OpenImages [37], and LAION [62], resulting in a total of about 33M samples. For Datacomp [26], we curate the images with resolutions greater than $512 \times 512$, using DenseFusion++ to obtain 44M high-quality image descriptions and abandon samples with repetitive text or incomplete sentences. *(2) Question-answering and Instruction-following Datasets.* We clean out 15M QA data from Infinity-MM-GeneralQA [29] in its Stage-2. Meanwhile, we collect a blended set of the LLaVA-onevision [40] and partial Infinity-MM-instruct [29] from its original Stage-3/4 for complicated conversation patterns.

**Implementation Details.** We use sixteen 8-A100 (40G) nodes to train EVEv2.0 using AdamW optimizer [35]. For Stage 1, 2.1, 2.2, and 3, the batch sizes are 1024, 1024, 512, and 512, while the maximum learning rates are set to $2 \times 10^{-4}$, $1 \times 10^{-4}$, $2 \times 10^{-5}$, and $1 \times 10^{-5}$. We adopt warm-up strategy with the ratio of 0.03 and cosine decay scheduler across all stages. Unless otherwise stated, we set image resolutions as $800^2$ and report fine-tuned results by LLaVA-mix-665K [50] for Stage 1/2.1/2.2 in Section 4.3.

## 4.2. Main Results

We conduct standard evaluations using the LMMs-Eval [100] across various vision-language benchmarks, including (1) Chart, Diagram, and Document Understanding tasks: OCR-Bench [54], ChartQA [58], and AI2D [34]; (2) Visual Perception and Challenging Reasoning tasks: MMMU [97], MMBench-EN [53], SEEDBench-IMG [39], MMVet [96], MME [25], POPE [47], GQA [32], ScienceQA-IMG [56], and TextVQA [63]. All the results are reported with greedy decoding and zero-shot settings, unless otherwise stated.

In Table 2, EVEv2.0 surpasses encoder-free counterparts, *e.g.* Fuyu [6], EVE [19], SOLO [13], *etc.* across various vision-language benchmarks. Besides, Mono-InternVL [57], despite its smaller model size, achieves strong performance by leveraging 13x more pretraining data and large-scale, high-quality SFT data. Notably, preliminary experiments in Figure 2 confirm that merely decoupling the feed-forward module is insufficient to address vision-language conflicts and module compatibility within a single unified network. Furthermore, we clarify and validate that our Divide-and-Conquer architecture can achieve better data-scaling efficiency over the Mixture-of-Experts approach employed in Mono-InternVL [57], as demonstrated in Figure 5.

Besides, EVEv2.0 displays superior performance against the VLMs using discrete tokenizers, *i.e.* Chameleon [69] and Emu3 [83], despite being trained on significantly fewer data or utilizing fewer visual tokens. This further validates the efficiency and effectiveness of encoder-free VLMs with lossless visual encoding mode, even using a lightweight patch embedding layer from scratch. Notably, EVEv2.0 competes with popular and mainstream encoder-based VLMs, *e.g.* LLaVA-1.6 [52] and Cambrian [74]. The performance gap with advanced modular VLMs stems largely from the significant discrepancy in in data scale. EVEv2.0 addresses this through modality-aware decomposition with high-quality annotations, offering an efficient and practical pathway for native VLMs to compete with encoder-based models.

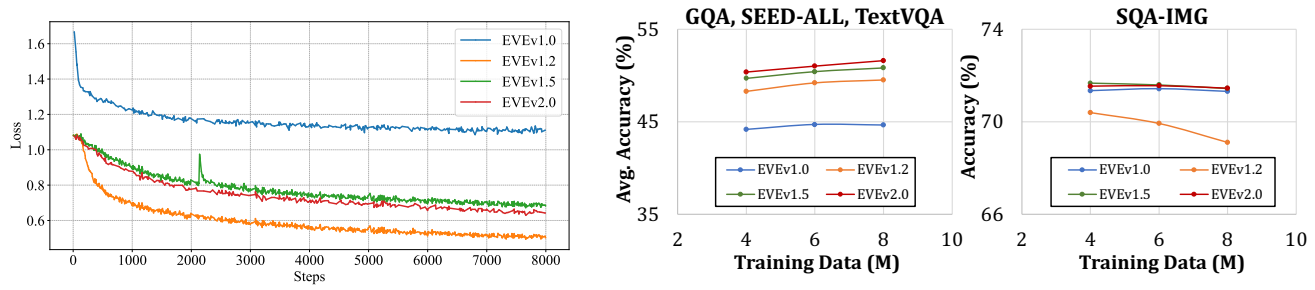



Figure 5. Training loss curve and evaluation results in Stage 2. We adopt various EVE variants based on Qwen-2.5 [73] as the baseline. We first train the patch embedding layer using *EVE-recap-10M* in Stage 1, and further unfreeze vision layers except LLM layers in Stage 2.

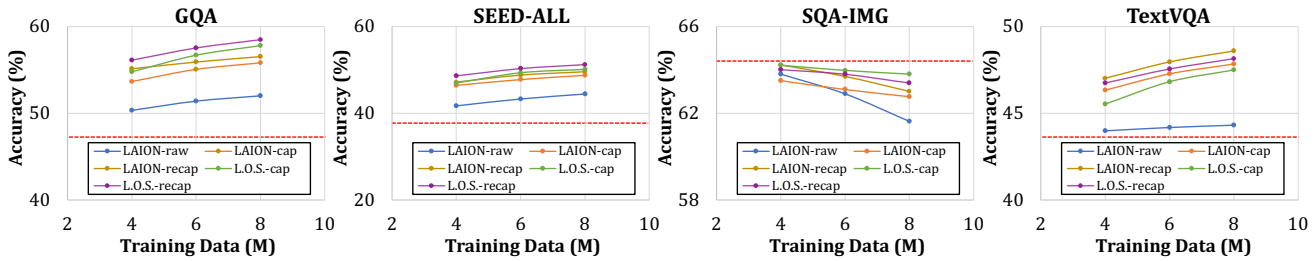



Figure 6. Evaluation results of different data sources and caption engines. We utilize EVEv1.0 based on Vicuna-7B [16] as the baseline. Here "*-raw", "*-cap", or "*-recap" denote noisy web image captions, the samples annotated by both LLaVA-1.5 (13B) and Emu2 (17B), or modified DenseFusion++ (7B), respectively. Note that "L.O.S." represents the mixture of LAION [62], OpenImages [37], and SAM [36].

### 4.3. Ablation Studies

**Divide-and-conquer (DaC) design outperforms the re-parameterization (ReP) and mixture-of-experts (MoE).** As shown in Figure 5, **(1)** the training process of EVEv1.0 is the slowest, and training only the patch embedding layer proves insufficient, resulting in minimal performance gains. **(2)** EVEv1.2 (ReP) shows a rapid loss decrease, which can be attributed to the gradual transfer of LLMs into the initial VLMs by updating the pre-trained LLM weights. However, this approach leads to a noticeable performance drop on the SQA-IMG task requiring abundant text-related knowledge. **(3)** In contrast, EVEv1.5 (MoE) only updates visual parameters inside frozen LLMs to effectively mitigate catastrophic forgetting issues during pre-training. However, solely decoupling FFN modules restricts distinct feature distributions across modalities, resulting in less-efficient improvements in visual perception and multi-modality alignment. **(4)** With prior validation support, EVEv2.0 (DaC) achieves optimal improvements across all multi-modal benchmarks, highlighting its superior data-scaling efficiency during large-scale pre-training. This success can be attributed to its modality-wise sparsity, which effectively preserves linguistic knowledge while providing greater flexibility for visual learning. This philosophy is further evidenced by the loss curve in Figure 5 with faster convergence and better training stability than EVEv1.5 during pre-training. **(5)** Notably, their Avg. accuracy gap rises from 0.8% to 1.4% as the training data grows from 8M to 24M, a trend likely to hold for other model sizes and data sources. Besides, only decoupling LayerNorm yields the Avg. accuracy of 48.8% vs. 51.6% for EVEv2.0 using 8M data, necessitating the complete decomposition.

**Fully-upgraded captioning engine facilitates training efficiency and model capabilities than prior competitors.** As illustrated in Figure 6, **(1)** web-scale image-text data often suffers from excessive noise and overly brief descriptions, which results in slow progress in visual content understanding and significantly pollutes LLM's pre-training knowledge. In contrast, using a powerful captioning engine to build high-quality, hyper-detailed image annotations proves essential for efficiently developing visual perception from scratch. Our modified DenseFusion++ (7B) outperforms previously adopted models like LLaVA-1.5 (13B) and Emu2 (17B) in this regard. Moreover, our model offers an additional advantage: its efficient and low-budget nature, capable of generating 700K descriptions per day with just a single 8-A100 (40G) node, accelerated by SGlang [103]. **(2)** A multi-source data mixture can significantly facilitate the visual training process. Our filtered LAION [62], OpenImages [37], SAM [36] provide OCR-related images, real-world scenarios, and abundant image content, respectively. Together, these data mixtures can enhance the capability of VLMs to handle diverse image inputs, promoting the development of a more robust and versatile visual perception.

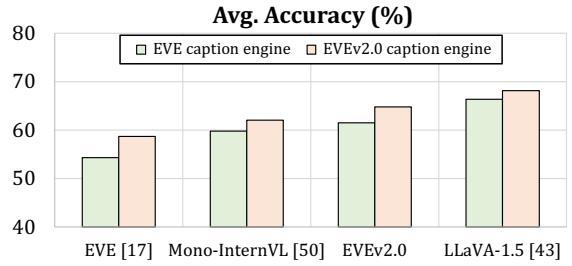



Figure 7. Evaluation results of different methods. We report results on GQA, SEED-ALL, TextQA, SQA-IMG, MMB, and MMVet.

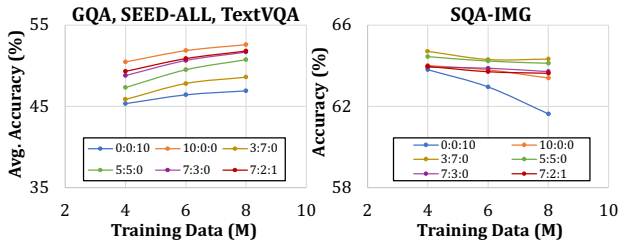



Figure 8. Evaluation results of mixed data ratio. We adopt EVEv1.0 with Vicuna-7B [16] for validation. Note that x:y:z denote the proportion of synthesized data : language-only data : web data.

**EVEv2.0 achieves notable performance, rivaling the modular backbone equipped with strong visual encoders.** In Figure 7, we utilize Qwen-2.5 [73] as the LLM with *EVE-recap-48M* for pre-training and *EVE-sft-7M* for fine-tuning. **(1)** EVEv2.0, with its modality-specific components and enhanced caption engine, effectively approaches the performance of encoder-based LLaVA-1.5 [50]. It surpasses the prototype EVE [19], which relies on visual supervision and a unified dense framework, as well as Mono-InternVL [57], which suffers from insufficient architecture decomposition. **(2)** The DenseFusion++ delivers consistent and significant gains across various types of VLMs, leveraging diverse visual experts to boost generalist VLMs' perception and data-scaling efficiency for native VLM construction. These findings further clarify the philosophy behind EVEv2.0.

**Meticulously adjusting data proportional distribution facilitates the balanced improvements across modalities.** We explore maintaining pre-trained LLM knowledge from a data mixture perspective in Figure 8. Striking the right balance is crucial for achieving robust multimodal capabilities without significantly sacrificing language performance. However, this balance is delicate and often influenced by various factors, *e.g.* image resolution, dataset composition, text sources, and the type of language model. In this paper, we address multi-modality compatibility from the model structure perspective, leveraging overall multi-modal synthesized data during pre-training. Nonetheless, we believe that combining these two could yield even greater benefits.

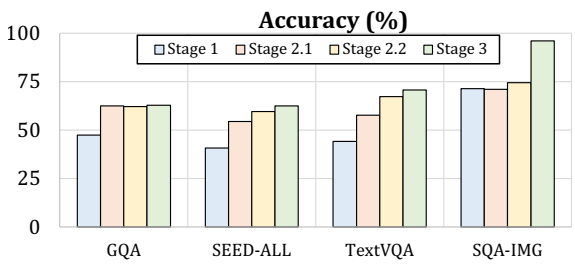



Figure 9. Evaluation results across progressive training procedures. We adopt standard EVEv2.0 based on Qwen-2.5 [73] as the LLM.

**Incremental benefits across different training stages.** To thoroughly investigate progressive training recipes, we present the training dynamics with detailed recipes in Table 1. From Figure 9, we observe continual improvements with increasing pre-training data scales, which are further enhanced by carefully organized question-answering datasets. After meticulously curating instruction-tuning datasets, our EVEv2.0 ultimately achieves superior multi-modality capabilities to handle various real-world application scenarios.

## 5. Limitation and Discussion

EVEv2.0 has systematically explored network architectures and training strategies for efficiently constructing encoder-free VLMs. Due to limitations in extensive high-quality data and computational devices, its full potential remains unrealized, thereby restricting performance on specific tasks, *e.g.* knowledge- and document-oriented benchmarks. Besides, several promising directions remain for further exploration and improvement: Model Scaling, Data Scaling, and Modalities Expanding (*e.g.* audio and video). We hope EVEv2.0 inspires further research on scaling laws for encoder-free VLMs with much more computational resources.

## 6. Conclusion

In this paper, we propose EVEv2.0 with a transparent and powerful encoder-free architecture for vision-and-language understanding. Rather than focusing solely on state-of-the-art results, we systematically analyze the most efficient strategies for building visual perception from scratch. To address vision-language interference, we fully disentangle the model components and introduce modality-wise sparsity within a unified decoder-only backbone. Besides, we establish an efficient pathway for optimizing data-scaling efficiency in monolithic VLM research through a modified caption engine and a meticulous training recipe. Utilizing only 100M public data, EVEv2.0 outperforms existing encoder-free counterparts and steadily approaches mainstream encoder-based approaches of similar model capacity across various vision-language benchmarks. This provides valuable insights for developing scalable, native VLMs for the next generation.

**Acknowledgements.** The paper is supported in part by the National Natural Science Foundation of China under grant No.62441231, 62293542, Liao Ning Province Science and Technology Plan No.2023JH26/10200016, Dalian City Science and Technology Innovation Fund No.2023JJ11CG001, and Natural Science Foundation of Zhejiang Province under Grant No.LD25F020001, Ningbo Key R&D project under Grant No.2025Z039, and National Key R&D Program of China (2022ZD0116302).

## References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. 1, 2

[2] AI Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 2024. 1, 2

[3] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report. *arXiv: 2309.16609*, 2023. 2, 3

[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*, 1(2):3, 2023. 1, 2

[5] Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhojit Som, Songhao Piao, and Furu Wei. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *Advances in Neural Information Processing Systems*, 35:32897–32912, 2022. 1

[6] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. Introducing our multimodal models, 2023. 1, 2, 6

[7] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024. 1, 2, 3

[8] Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. Deepseek llm: Scaling open-source language models with longtermism. *arXiv: 2401.02954*, 2024. 1

[9] Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, Xiaoyi Dong, Haodong Duan, Qi Fan, Zhaoye Fei, Yang Gao, Jiaye Ge, Chenya Gu, Yuzhe Gu, Tao Gui, Aijia Guo, Qipeng Guo, Conghui He, Yingfan Hu, Ting Huang, Tao Jiang, Penglong Jiao, Zhenjiang Jin, Zhikai Lei, Jiaxing Li, Jingwen Li, Linyang Li, Shuaibin Li, Wei Li, Yining Li, Hongwei Liu, Jiangning Liu, Jiawei Hong, Kaiwen Liu, Kuikun Liu, Xiaoran Liu, Chengqi Lv, Haijun Lv, Kai Lv, Li Ma, Runyuan Ma, Zerun Ma, Wenchang Ning, Linke Ouyang, Jiantao Qiu, Yuan Qu, Fukai Shang, Yunfan Shao, Demin Song, Zifan Song, Zhihao Sui, Peng Sun, Yu Sun, Huanze Tang, Bin Wang, Guoteng Wang, Jiaqi Wang, Jiayu Wang, Rui Wang, Yudong Wang, Ziyi Wang, Xingjian Wei, Qizhen Weng, Fan Wu, Yingtong Xiong, and et al. Internlm2 technical report. *arXiv: 2403.17297*, 2024. 1, 2

[10] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pages 9630–9640, 2021. 2

[11] Guiming Hardy Chen, Shunian Chen, Ruifei Zhang, Junying Chen, Xiangbo Wu, Zhiyi Zhang, Zhihong Chen, Jianquan Li, Xiang Wan, and Benyou Wang. Allava: Harnessing gpt4v-synthesized data for a lite vision-language model. *arXiv preprint arXiv:2402.11684*, 2024. 14

[12] 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: 2311.12793*, 2023. 2, 14

[13] Yangyi Chen, Xingyao Wang, Hao Peng, and Heng Ji. A single transformer for scalable vision-language modeling. *arXiv preprint arXiv:2407.06438*, 2024. 1, 2, 6

[14] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv: 2312.14238*, 2023. 2

[15] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv:2404.16821*, 2024. 1, 2

[16] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 2, 3, 7, 8, 14, 15

[17] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven C. H. Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *NeurIPS*, 2023. 14

[18] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter

Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, Rodolphe Jenatton, Lucas Beyer, Michael Tschannen, Anurag Arnab, Xiao Wang, Carlos Riquelme Ruiz, Matthias Minderer, Joan Puigcerver, Utku Evci, Manoj Kumar, Sjoerd van Steenkiste, Gamaleldin Fathy Elsayed, Aravindh Mahendran, Fisher Yu, Avital Oliver, Fantine Huot, Jasmijn Bastings, Mark Collier, Alexey A. Gritsenko, Vighnesh Birodkar, Cristina Nader Vasconcelos, Yi Tay, Thomas Mensink, Alexander Kolesnikov, Filip Pavetic, Dustin Tran, Thomas Kipf, Mario Lucic, Xiaohua Zhai, Daniel Keysers, Jeremiah J. Harmsen, and Neil Houlsby. Scaling vision transformers to 22 billion parameters. In *ICML*, pages 7480–7512, 2023. 1

[19] Haiwen Diao, Yufeng Cui, Xiaotong Li, Yueze Wang, Huchuan Lu, and Xinlong Wang. Unveiling encoder-free vision-language models. *arXiv preprint arXiv:2406.11832*, 2024. 1, 2, 3, 5, 6, 8, 14

[20] Haiwen Diao, Bo Wan, Ying Zhang, Xu Jia, Huchuan Lu, and Long Chen. Unipt: Universal parallel tuning for transfer learning with efficient parameter and memory. In *CVPR*, 2024. 1

[21] Haiwen Diao, Bo Wan, Xu Jia, Yunzhi Zhuge, Ying Zhang, Huchuan Lu, and Long Chen. Sherl: Synthesizing high accuracy and efficient memory for resource-limited transfer learning. In *ECCV*, pages 75–95, 2025. 1

[22] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 1

[23] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, pages 12873–12883, 2021. 2

[24] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. EVA: exploring the limits of masked visual representation learning at scale. In *CVPR*, pages 19358–19369, 2023. 1

[25] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, and Rongrong Ji. MME: A comprehensive evaluation benchmark for multimodal large language models. *arXiv: 2306.13394*, 2023. 6

[26] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *NeurIPS*, 36, 2024. 5, 6, 14

[27] 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: 2304.15010*, 2023. 2

[28] Yuying Ge, Sijie Zhao, Ziyun Zeng, Yixiao Ge, Chen Li, Xintao Wang, and Ying Shan. Making LLaMA SEE and draw with SEED tokenizer. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[29] Shuhao Gu, Jialing Zhang, Siyuan Zhou, Kevin Yu, Zhaohu Xing, Liangdong Wang, Zhou Cao, Jintao Jia, Zhuoyi Zhang, Yixuan Wang, et al. Infinity-mm: Scaling multimodal performance with large-scale and high-quality instruction data. *arXiv preprint arXiv:2410.18558*, 2024. 5, 6, 14

[30] Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, et al. Cogagent: A visual language model for gui agents. *arXiv:2312.08914*, 2023. 2

[31] Anwen Hu, Haiyang Xu, Liang Zhang, Jiabo Ye, Ming Yan, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. mplug-docowl2: High-resolution compressing for ocr-free multi-page document understanding. *arXiv preprint arXiv:2409.03420*, 2024. 14

[32] Drew A. Hudson and Christopher D. Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, pages 6700–6709, 2019. 3, 6

[33] IDEFICS Research Team. Introducing idefics: An open reproduction of state-of-the-art visual language model. https://huggingface.co/blog/idefics, 2023. 2

[34] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *ECCV*, pages 235–251, 2016. 6

[35] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[36] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. *arXiv: 2304.02643*, 2023. 5, 6, 7, 14

[37] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, and Vittorio Ferrari. The open images dataset V4: unified image classification, object detection, and visual relationship detection at scale. *arXiv: 1811.00982*, 2018. 5, 6, 7, 14

[38] Hugo Laurençon, Andrés Marafioti, Victor Sanh, and Léo Tronchon. Building and better understanding vision-language models: insights and future directions. *arXiv preprint arXiv:2408.12637*, 2024. 14

[39] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv: 2307.16125*, 2023. 6

[40] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 2, 3, 5, 6, 14

[41] Dongxu Li, Yudong Liu, Haoning Wu, Yue Wang, Zhiqi Shen, Bowen Qu, Xinyao Niu, Guoyin Wang, Bei Chen, and Junnan Li. Aria: An open multimodal native mixture-of-experts model. *arXiv preprint arXiv:2410.05993*, 2024. 1

[42] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICLR*, pages 12888–12900, 2022. 2

[43] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: bootstrapping language-image pre-training

with frozen image encoders and large language models. In *ICML*, pages 19730–19742, 2023. 1, 2

[44] Xiaotong Li, Yixiao Ge, Kun Yi, Zixuan Hu, Ying Shan, and Ling-Yu Duan. mc-beit: Multi-choice discretization for image bert pre-training. In *ECCV*, pages 231–246, 2022. 1

[45] Xiang Li, Kai Qiu, Hao Chen, Jason Kuen, Jiuxiang Gu, Bhiksha Raj, and Zhe Lin. Imagefolder: Autoregressive image generation with folded tokens. *arXiv preprint arXiv:2410.01756*, 2024. 2

[46] Xiaotong Li, Fan Zhang, Haiwen Diao, Yueze Wang, Xinlong Wang, and Ling-Yu Duan. Densefusion-1m: Merging vision experts for comprehensive multimodal perception. *arXiv preprint arXiv:2407.08303*, 2024. 2, 14

[47] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *EMNLP*, pages 292–305, 2023. 6

[48] Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-modal models. *arXiv: 2311.06607*, 2023. 2

[49] Xi Victoria Lin, Akshat Shrivastava, Liang Luo, Srinivasan Iyer, Mike Lewis, Gargi Gosh, Luke Zettlemoyer, and Armen Aghajanyan. Moma: Efficient early-fusion pre-training with mixture of modality-aware experts. *arXiv preprint arXiv:2407.21770*, 2024. 1

[50] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv: 2310.03744*, 2023. 2, 3, 6, 8, 14

[51] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 1

[52] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 2, 5, 6

[53] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player? *arXiv: 2307.06281*, 2023. 6

[54] Yuliang Liu, Zhang Li, Biao Yang, Chunyuan Li, Xucheng Yin, Cheng-lin Liu, Lianwen Jin, and Xiang Bai. On the hidden mystery of ocr in large multimodal models. *arXiv preprint arXiv:2305.07895*, 2023. 6

[55] Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. Deepseek-vl: Towards real-world vision-language understanding. *arXiv: 2403.05525*, 2024. 1, 3

[56] Pan Lu, Swaroop Mishra, Tony 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 *NeurIPS*, 2022. 3, 6

[57] Gen Luo, Xue Yang, Wenhan Dou, Zhaokai Wang, Jifeng Dai, Yu Qiao, and Xizhou Zhu. Mono-internvl: Pushing the boundaries of monolithic multimodal large language models with endogenous visual pre-training. *arXiv preprint arXiv:2410.08202*, 2024. 1, 2, 3, 6, 8

[58] Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *ACL*, pages 2263–2279, 2022. 6

[59] OpenAI. GPT-4 technical report. *arXiv: 2303.08774*, 2023. 1, 2

[60] Liao Qu, Huichao Zhang, Yiheng Liu, Xu Wang, Yi Jiang, Yiming Gao, Hu Ye, Daniel K Du, Zehuan Yuan, and Xinglong Wu. Tokenflow: Unified image tokenizer for multimodal understanding and generation. *arXiv preprint arXiv:2412.03069*, 2024. 2

[61] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 1, 2, 3

[62] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 35:25278–25294, 2022. 5, 6, 7, 14

[63] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *CVPR*, 2019. 3, 6

[64] Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024. 2

[65] Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Zhengxiong Luo, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative multimodal models are in-context learners. *arXiv: 2312.13286*, 2023. 2

[66] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. EVA-CLIP: improved training techniques for CLIP at scale. *arXiv: 2303.15389*, 2023. 1, 2

[67] Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative pretraining in multimodality. *arXiv: 2307.05222*, 2023. 2

[68] Quan Sun, Jinsheng Wang, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, and Xinlong Wang. EVA-CLIP-18B: scaling CLIP to 18 billion parameters. *arXiv: 2402.04252*, 2024. 1

[69] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. 2, 6

[70] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv: 2312.11805*, 2023. 2

[71] InternLM Team. Internlm: A multilingual language model with progressively enhanced capabilities. https://github.com/InternLM/InternLM, 2023. 2

[72] Meta Team. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models, 2024. 1, 2

[73] Qwen Team. Qwen2.5: A party of foundation models, 2024. 2, 4, 5, 7, 8

[74] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*, 2024. 1, 6, 14

[75] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *CVPR*, pages 9568–9578, 2024. 1

[76] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *arXiv: 2302.13971*, 2023. 2

[77] 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: 2307.09288*, 2023. 1, 2

[78] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *NeurIPS*, 30, 2017. 2

[79] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 3

[80] 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: 2311.07574*, 2023. 14

[81] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 2

[82] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, and Furu Wei. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv: 2208.10442*, 2022. 1

[83] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024. 2, 3, 6

[84] Zekun Wang, King Zhu, Chunpu Xu, Wangchunshu Zhou, Jiaheng Liu, Yibo Zhang, Jiashuo Wang, Ning Shi, Siyu Li, Yizhi Li, et al. Mio: A foundation model on multimodal tokens. *arXiv preprint arXiv:2409.17692*, 2024. 2

[85] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. *arXiv preprint arXiv:2410.13848*, 2024. 2

[86] Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, et al. Vila-u: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2024. 2

[87] x.ai. Grok-1.5 vision preview, 2024. 6

[88] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024. 2

[89] Rongchang Xie, Chen Du, Ping Song, and Chang Liu. Muse-vl: Modeling unified vlm through semantic discrete encoding. *arXiv preprint arXiv:2411.17762*, 2024. 2

[90] Ruyi Xu, Yuan Yao, Zonghao Guo, Junbo Cui, Zanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, Maosong Sun, and Gao Huang. Llava-uhd: an LMM perceiving any aspect ratio and high-resolution images. *arXiv: 2403.11703*, 2024. 1, 2

[91] Le Xue, Manli Shu, Anas Awadalla, Jun Wang, An Yan, Senthil Purushwalkam, Honglu Zhou, Viraj Prabhu, Yutong Dai, Michael S Ryoo, et al. xgen-mm (blip-3): A family of open large multimodal models. *arXiv preprint arXiv:2408.08872*, 2024. 2

[92] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024. 1, 2, 14

[93] Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv: 2309.17421*, 9, 2023. 1

[94] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, Chenliang Li, Yuanhong Xu, Hehong Chen, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. mplug-owl: Modularization empowers large language models with multimodality. *arXiv: 2304.14178*, 2023. 2

[95] Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. *arXiv: 2311.04257*, 2023. 2

[96] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv: 2308.02490*, 2023. 6

[97] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. *arXiv: 2311.16502*, 2023. 6

[98] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, pages 11975–11986, 2023. 1, 2

[99] Jun Zhan, Junqi Dai, Jiasheng Ye, Yunhua Zhou, Dong Zhang, Zhigeng Liu, Xin Zhang, Ruibin Yuan, Ge Zhang, Linyang Li, Hang Yan, Jie Fu, Tao Gui, Tianxiang Sun, Yugang Jiang, and Xipeng Qiu. Anygpt: Unified multimodal LLM with discrete sequence modeling. *arXiv: 2402.12226*, 2024. 2

[100] Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, et al. Lmms-eval: Reality check on the evaluation of large multimodal models. *arXiv preprint arXiv:2407.12772*, 2024. 6

[101] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. In *ICLR*, 2024. 2

[102] Chuanxia Zheng, Tung-Long Vuong, Jianfei Cai, and Dinh Phung. Movq: Modulating quantized vectors for high-fidelity image generation. *NeurIPS*, 35:23412–23425, 2022. 3

[103] Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Jeff Huang, Chuyue Sun, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark W. Barrett, and Ying Sheng. Efficiently programming large language models using sglang. *arXiv: 2312.07104*, 2023. 7

Table 3. Experiment Details in the main body. Note that T.M. denotes trainable modules in each stage. PEL, PAL, and VLayers represent patch embedding, alignment, and newly added vision layers within the LLM. EVE-recap-8/29M indicates a subset 8M of 29M training data.

| Exp. | Model | LLM | Stage 1 | | Stage 2 | | Stage 3 | |
|---|---|---|---|---|---|---|---|---|
| | | | Data | T.M. | Training Data | Trainable Module | Data | T.M. |
| Fig.2 (i) | EVEv1.0 | Vicuna-7B | EVE-cap-16M | PEL | EVE-cap-33M | PEL, LLM | LLaVA-mix-665k | PEL, LLM |
| Fig.5 | EVEv1.0 | Qwen2.5-7B | EVE-recap-10M | PEL | EVE-recap-8/29M | PEL | LLaVA-mix-665k | PEL, LLM |
| | EVEv1.2 | Qwen2.5-7B | EVE-recap-10M | PEL | EVE-recap-8/29M | PEL, VLayers | LLaVA-mix-665k | PEL, LLM |
| | EVEv1.5 | Qwen2.5-7B | EVE-recap-10M | PEL | EVE-recap-8/29M | PEL, VLayers | LLaVA-mix-665k | PEL, LLM |
| | EVEv2.0 | Qwen2.5-7B | EVE-recap-10M | PEL | EVE-recap-8/29M | PEL, VLayers | LLaVA-mix-665k | PEL, LLM |
| Fig.6 | EVEv1.0 | Vicuna-7B | 10M varied data | PEL | 8M same data from Stage 1 | PEL, LLM | LLaVA-mix-665k | PEL, LLM |
| Fig.7 | EVE [17] | Qwen2.5-7B | EVE-cap/recap-10/48M | PEL, PAL | EVE-cap/recap-48M | PEL, PAL, LLM | EVE-sft-7M | PEL, PAL, LLM |
| | Mono-InternVL [50] | Qwen2.5-7B | EVE-cap/recap-10/48M | PEL | EVE-cap/recap-48M | PEL, VLayers | EVE-sft-7M | PEL, LLM |
| | EVEv2.0 | Qwen2.5-7B | EVE-cap/recap-10/48M | PEL | EVE-cap/recap-48M | PEL, VLayers | EVE-sft-7M | PEL, LLM |
| | LLaVA-1.5 [43] | Qwen2.5-7B | EVE-cap/recap-10/48M | Projector | EVE-cap/recap-48M | Vision Encoder, Projector | EVE-sft-7M | All Layers |
| Fig.8 | EVEv1.0 | Vicuna-7B | 10M varied data | PEL | 8M same data from Stage 1 | PEL, LLM | LLaVA-mix-665k | PEL, LLM |
| Supp.Fig.1 | EVEv1.0 | Vicuna-7B | EVE-recap-10M | PEL | EVE-recap-8/29M | PEL, LLM | LLaVA-mix-665k | PEL, LLM |

| Exp. | Model | LLM | Stage 1 | | Stage 2.1 | | Stage 2.2 | | Stage 3 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Data | T.M. | Data | T.M. | Data | T.M. | Data | T.M. |
| Fig.2 (ii) | EVEv1.0 | Qwen2-7B | EVE-recap-10M | PEL | EVE-recap-29M | PEL | EVE-recap-48M | PEL, LLM | Various SFT data | PEL, LLM |
| | EVEv1.2 | Qwen2-7B | EVE-recap-10M | PEL | EVE-recap-29M | PEL, VLayers | EVE-recap-48M | PEL, LLM | Various SFT data | PEL, LLM |
| Tab.2 | EVEv2.0 | Qwen2.5-7B | EVE-recap-10M | PEL | EVE-recap-77M | PEL, VLayers | EVE-multi-task-5M | PEL, LLM | EVE-sft-7M | PEL, LLM |
| Fig.9 | EVEv2.0 | Qwen2.5-7B | EVE-recap-10M | PEL | EVE-recap-77M | PEL, VLayers | EVE-multi-task-5M | PEL, LLM | EVE-sft-7M | PEL, LLM |

# A. Experiment Details

All experiment details in the main body are listed in Table 3. "EVE-cap-16M" denotes the data mixture of LAION [62], OpenImages [37], and SAM [36] annotated by LLaVA-1.5 (13B) and Emu2 (17B). "EVE-recap-16M" denotes the data mixture of Datacomp [26], LAION [62], OpenImages [37], and SAM [36] annotated by DenseFusion++ (7B).

For Exp.(i), we first use EVE-cap-16M in Stage 1 to train the projector, vision vocabulary embeddings, and lightweight vision block for the vision encoder (*VE*), the discrete tokenizer (*DT*), and EVEv1.0. In Stage 2, we use EVE-cap-33M to train only the vision encoder and projector for *VE*, as unfreezing LLM weights at this stage leads to performance collapse [19]. For *DT* and EVEv1.0, we unfreeze all model parameters. In Stage 3, we train all model weights across all models. Finally, we quantify weight changes between Vicuna-7B [16] and *VE* / EVEv1.0 to analyze the architectural differences. To ensure fairness, we remove the original vision encoder supervision in EVEv1.0.

For Exp.(ii), all VLMs use stronger Qwen2-7B [92] and high-quality EVE-recap. In Stage 1, we train the projector for *VE*, patch embedding layer for EVEv1.0, patch embedding and extra vision layer inside the LLM for EVEv1.2. In Stage 2 and 3, we train all model weights for EVEv1.0-1.2. Note that we skip Stage 2 for *VE* as the baseline for comparison. Here, we compare the weights between Qwen2 and *VE* / EVEv1.2 trained by LLaVA-onevision [40].

# B. Dataset Details

From Table 4, we use Cambrian-FL [74], Infinity-Instruct-FL [29], LVIS-Instruct-FL [80], Sharegpt4v-FL [12], ALLaVA-laion-FL [11], ALLaVA-vflan-FL [11], LLaVA-Pretrain-FL [50], DocReason-FL [31], DocDownstream-FL [31], and DocStruct4M-FL [31] in Stage 2.2. In Stage 3, we adopt LLaVA-onevision [40] and Infinity-MM [29].

Table 4. Dataset details in Stage 2.2, and 3 for fine-tuning EVEv2.0. Note that ***-FL denotes the filtered training dataset.

| Stage | Dataset | #Data |
|---|---|---|
| 2.2 | Cambrian-FL [74], Infinity-Instruct-FL [29], LVIS-Instruct-FL [80], Sharegpt4v-FL [12], ALLaVA-laion-FL [11], ALLaVA-vflan-FL [11], LLaVA-Pretrain-FL [50], DocReason-FL [31], DocDownstream-FL [31], DocStruct4M-FL [31]. | 5M |
| 3 | LLaVA-onevision [40], Infinity-MM-Synthesis [29], Infinity-MM-Preference [29], Infinity-Instruct-FL [29], DenseFusion [46], Cambrian-FL [74], Docmatix-FL [38], LVIS-Instruct-FL [80], BLIP-OCR[17], LLaVA-mix [50]. | 7.3M |

Table 5. Hyper-parameter configurations in Stage 1-3 for training EVEv2.0. Note that we set the training epoch in each stage as 1.

| Configuration | Stage 1 | Stage 2.1 | Stage 2.2 | Stage 3 |
|---|---|---|---|---|
| Maximum Patch Token | 625 | 625 − 2500 | 2500 | 2500 |
| Optimizer | AdamW | | | |
| Hyperparameters | $\beta_1 = 0.9, \beta_2 = 0.999, eps = 1e^{-8}$ | | | |
| Peak learning rate | $2e^{-4}$ | $1e^{-4}$ | $2e^{-5}$ | $1e^{-5}$ |
| LR schedule | cosine decay with warm-up | | | |
| Warm-up steps | 0.03 | | | |
| Weight decay | 0.0 | | | |
| Global batch size | 1024 | 1024 | 512 | 512 |
| Numerical precision | `bfloat16` | | | |

# C. Hyper-parameter Configurations

The detailed implementation configurations in Stages 1, 2.1, 2.2, and 3 are summarized in Table 5.

# D. Additional Ablation Studies

**Decreasing SQA-Img metric with more training data.** We empirically discovered that the performance of the SQA-Img metric is strongly influenced by the knowledge of the pre-trained LLM, making it prone to catastrophic linguistic forgetting issues. As it primarily involves text-centric knowledge tasks, increasing image captioning data alone degrades performance when the LLM parameters are unfrozen.

**EVEv1.2: lowest loss, poor performance in Figure 5.** This is because EVEv1.2 progressively updates the LLM weights while quickly adapting to the simplified language structures within the VL caption data. Notably, the SFT losses—which better reflect task performance—exhibit a clear downward trend (EVEv1.2: 1.0306, EVEv1.5: 0.9513, EVEv2: 0.9327), indicating that EVEv1.2 is less effective at alleviating LLM forgetting and multimodal interference.

**Consistent benefits across different LLM capacities.** We train EVEv1, v1.5, and v2 using Qwen2.5-1.5B with EVE-recap-4M and LLaVA-mix-665k, achieving average scores of 35.1%, 37.3%, and 38.5% across six diverse vision-language benchmarks. This further confirms that sparse design consistently benefits models of different scales.

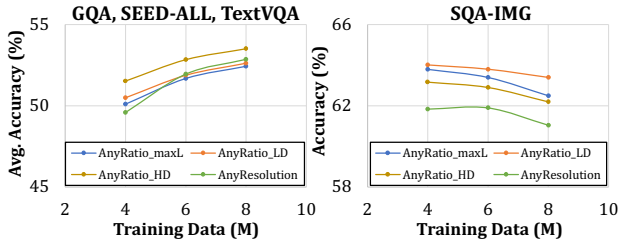



Figure 10. Evaluation results of image settings. We use EVEv1.0 with Vicuna-7B [16]. "AnyRatio_maxL": longest image edge as 800, "AnyRatio_LD": fixed image area as $800^2$, "AnyRatio_HD": fixed image area as $1600^2$, "AnyResolution": arbitrary resolution.

**Flexible image processing mode of encoder-free VLMs.** In Figure 10, we explore four different input formats. Among them, "AnyRatio_HD" (standard) provides the best performance gains, while "AnyResolution" performs poorly in the early stages but shows improved data-scaling efficiency over data scales. This early underperformance is likely due to limited pre-training data and the imbalance in image resolutions. We believe that with sufficient and well-balanced data, arbitrary-resolution inputs can offer better computational efficiency and greater flexibility for real-world images.

Table 6. Inference speed comparison of various EVE counterparts

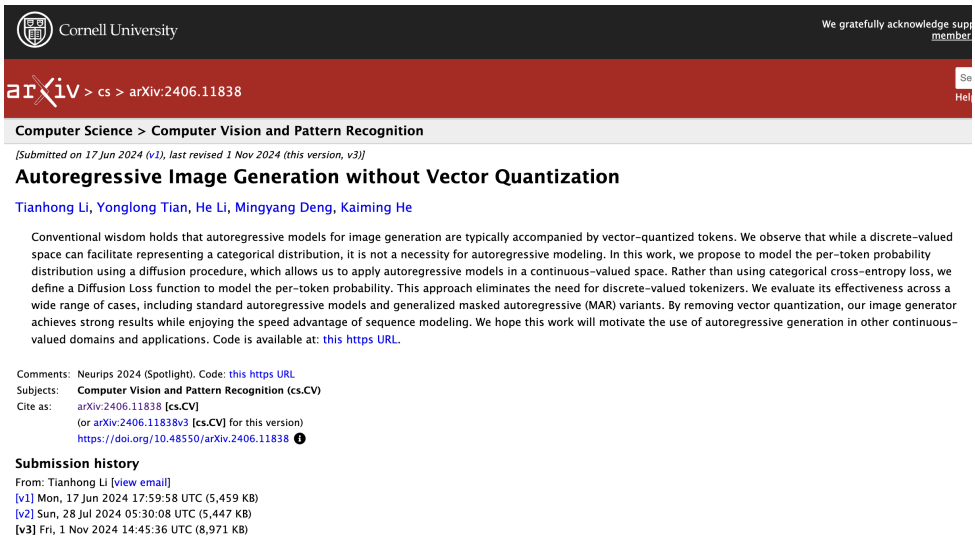
| Model | EVEv1.0 | EVEv1.2 | EVEv1.5 | EVEv2.0 |
|---|---|---|---|---|
| TTFT (s) | 3.77 | 3.77 | 4.01 | 4.25 |

**Inference speed comparisons across EVE variants.** EVEv2.0 decouples weights across modalities at each layer, totaling 2×7B parameters with active FLOPs per token matching a 7B dense model. Notably, we adopt an NVIDIA A100 with PyTorch, fixing image, input text, and output text tokens to 2500, 200, and 100, respectively. We report TTFT (time to first token) and TPS (tokens per second after the first token). Here, EVEv1.5 / EVEv2 bring marginal 6 / 13% TTFT overheads, while all EVEv1–v2 models sustain 35 TPS (tok/s) using KV-cache for visual and text prefixes.

# E. Visual Understanding Demonstration

We investigate several vision perception and reasoning capabilities of EVEv2.0, including OCR capability in Tables 7 and 8, mixed information in Table 9, real-world scenes in Tables 10 and 11, and set-of-mark prompting task in Table 12.

Example 1: Understanding OCR Information in Webpage



Cornell University

We gratefully acknowledge supp
member

arXiv > cs > arXiv:2406.11838

Se
Hel

Computer Science > Computer Vision and Pattern Recognition

*[Submitted on 17 Jun 2024 (v1), last revised 1 Nov 2024 (this version, v3)]*

**Autoregressive Image Generation without Vector Quantization**

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, Kaiming He

Conventional wisdom holds that autoregressive models for image generation are typically accompanied by vector-quantized tokens. We observe that while a discrete-valued space can facilitate representing a categorical distribution, it is not a necessity for autoregressive modeling. In this work, we propose to model the per-token probability distribution using a diffusion procedure, which allows us to apply autoregressive models in a continuous-valued space. Rather than using categorical cross-entropy loss, we define a Diffusion Loss function to model the per-token probability. This approach eliminates the need for discrete-valued tokenizers. We evaluate its effectiveness across a wide range of cases, including standard autoregressive models and generalized masked autoregressive (MAR) variants. By removing vector quantization, our image generator achieves strong results while enjoying the speed advantage of sequence modeling. We hope this work will motivate the use of autoregressive generation in other continuous-valued domains and applications. Code is available at: this https URL.

Comments: Neurips 2024 (Spotlight). Code: this https URL
Subjects: **Computer Vision and Pattern Recognition (cs.CV)**
Cite as: arXiv:2406.11838 **[cs.CV]**
(or arXiv:2406.11838v3 **[cs.CV]** for this version)
https://doi.org/10.48550/arXiv.2406.11838

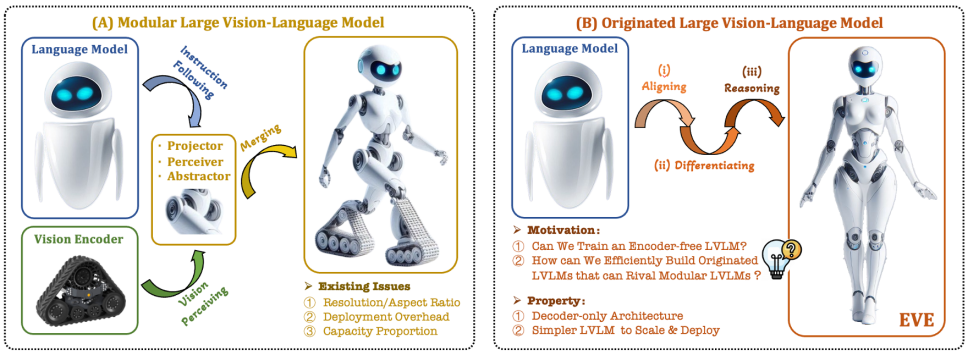
**Submission history**
From: Tianhong Li [view email]
[v1] Mon, 17 Jun 2024 17:59:58 UTC (5,459 KB)
[v2] Sun, 28 Jul 2024 05:30:08 UTC (5,447 KB)
**[v3]** Fri, 1 Nov 2024 14:45:36 UTC (8,971 KB)

**User** Please output all the text information.

**EVEv2.0** Cornell University We gratefully acknowledge support member. arXiv:2406.11838 Computer Science > Computer Vision and Pattern Recognition Submitted on 17 Jun 2024 (v1), last revised 1 Nov 2024 (this version, v3))
Autoregressive Image Generation without Vector Quantization
Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, Kaiming He
Conventional wisdom holds that autoregressive models for image generation are typically accompanied by vector-quantized tokens. We observe that while a discrete-valued space can facilitate representing a categorical distribution, it is not a necessity for autoregressive modeling. In this work, we propose to model the per-token probability distribution using a diffusion procedure, which allows us to apply autoregressive models in a continuous-valued space. Rather than using categorical cross-entropy loss, we define a Diffusion Loss function to model the per-token probability. This approach eliminates the need for discrete-valued tokenizers. We evaluate its effectiveness across a wide range of cases, including standard autoregressive models and generalized masked autoregressive (MAR) variants. By removing vector quantization, our image generator achieves strong results while enjoying the speed advantage of sequence modeling. We hope this work will motivate the use of autoregressive generation in other continuous-valued domains and applications. Code is available at: this https URL
Comments: Neurips 2024 (Spotlight); Code: this https URL Subjects: Computer Vision and Pattern Recognition (cs.CV) Cite as: arXiv:2406.11838 [cs.CV] (or arXiv:2406.11838v3 [cs.CV] for this version) Submission history From: Tianhong Li [view email] [v1] Mon, 17 Jun 2024 17:59:58 UTC (5,459 KB) [v2] Sun, 28 Jul 2024 05:30:08 UTC (5,447 KB) [v3] Fri, 1 Nov 2024 14:45:36 UTC (8,971 KB)

Table 7. EVEv2.0 exhibits excellent OCR recognition capabilities in understanding Webpage.

Example 2: Understanding OCR Information in PPT




| | |
|---|---|
| **User** | Please output all the text information. |
| **EVEv2.0** | (A) Modular Large Vision-Language Model (B) Originated Large Vision-Language Model Language Model Interaction Following (i) Aligning Reasoning (ii) Differentiating Vision Encoder Perceiver Merging Projector Abstractor Vision Encoder Vision Encoder Existing Issues (1) Resolution/Aspect Ratio (2) Deployment Overhead (3) Capacity Proportion Motivation: Can We Train an Encoder-free LVLM? How can We Efficiently Build Originated LVLMs that can Rival Modular LVLMs? Property: (1) Decoder-only Architecture (2) Simpler LVLM to Scale & Deploy EVE. |

Table 8. EVEv2.0 exhibits excellent OCR recognition capabilities in understanding PowerPoint.

Example 3: Understanding Poster Content

| | |
|---|---|
| **User** | Please describe the image in detail. |
| **EVEv2.0** | The image is a promotional poster for "Neanderthal," which appears to be a documentary or film about Neanderthals. The title "Neanderthal" is prominently displayed in large, bold yellow letters across the center of the image. Below the title, there is additional text that reads "Years 7 and 8 Welcome of the Day," suggesting that this event might be targeted towards students in those grade levels.<br>The background of the poster features a close-up photograph of a man with a Neanderthal-like appearance. He has a long beard, a prominent forehead, and a strong jawline, which are characteristic features often associated with Neanderthals. The man is holding a wooden stick or pole, which adds to the prehistoric theme of the image. The background behind him is blurred, drawing focus to his face and upper body.<br>At the bottom right corner of the poster, there is more text indicating the date of the event: "Friday 23 January 2015." This provides a specific time frame for when the event or release of the documentary/film would have taken place.<br>Overall, the poster uses visual elements such as the Neanderthal figure and the title to convey information about the subject matter, while also providing essential details like the target audience and the date of the event. The design effectively captures attention and communicates key information to viewers. |

Table 9. EVEv2.0 exhibits excellent visual recognition capabilities in understanding Poster.

Example 4: Understanding Real-world Scene

| | |
|---|---|
| **User** | Please describe the image in detail. |
| **EVEv2.0** | In the image, there is a man standing in front of a bathroom mirror, taking a selfie with his cell phone. He is wearing a purple shirt and leopard print pants. The man appears to be smiling as he captures the photo.<br>In the background, there is a potted plant with white flowers placed on the counter near the sink. The sink is white and has a modern design. There is also a poster hanging on the wall next to the mirror. The poster features a silhouette of a person's head with several circles on it. Below the silhouette, there is text that reads "COME SETS," followed by some additional information about Friday night events at Codame.com.<br>The bathroom appears to be well-lit, likely due to natural light coming from a window or another source not visible in the image. The overall atmosphere seems casual and relaxed, with the man enjoying his time in the bathroom while capturing a moment through his phone camera. |

Table 10. EVEv2.0 exhibits excellent recognition capabilities in the real-world scene.

Example 5: Understanding Real-world Scene

| | |
|---|---|
| **User** | Please describe the image in detail. |
| **EVEv2.0** | The image depicts a well-organized workspace featuring a wooden desk with various items on it. At the center of the desk is a closed silver laptop computer, which has an Apple logo on its lid. This laptop is accompanied by a white keyboard placed directly in front of it, suggesting that the user might prefer using an external keyboard for typing.<br>To the right of the laptop, there is a white wireless mouse, indicating that the user also utilizes a mouse for navigation and other tasks. On the left side of the desk, there is a glass containing a beverage, possibly water or another drink, which is positioned near the edge of the desk.<br>In addition to these items, there are two cylindrical objects resembling speakers, one on each side of the desk. These speakers likely provide audio output for the user while working or listening to music.<br>Behind the desk, there is a bookshelf filled with numerous books, suggesting that the user may have a keen interest in reading or research. The presence of multiple books indicates that this workspace could be used for studying, work, or leisure activities involving reading.<br>Overall, the image portrays a functional and organized workspace designed for productivity and comfort, with essential computing equipment and additional accessories such as speakers and a beverage. |

Table 11. EVEv2.0 exhibits excellent recognition capabilities in the real-world scene.

Example 6: Set-of-mark Prompting

| | |
|---|---|
| **User** | Here is an image with objects labeled in numbers. Please describe the objects that are marked as 9, 12 and 15. |
| **EVEv2.0** | In the image, there are three objects marked with numbers: 9, 12, and 15.<br>Object 9: A laptop computer is placed on the desk. It appears to be open and ready for use.<br>Object 12: A lamp is situated near the desk, providing light for the workspace. It has a sleek design and complements the overall aesthetic of the room.<br>Object 15: A potted plant is located on the desk, adding a touch of greenery and life to the space. It contributes to creating a more pleasant and inviting atmosphere in the room. |

Table 12. EVEv2.0 exhibits excellent set-of-mark prompting capabilities, ie, referring to marks when answering questions.