# Do VLMs Need Vision Transformers? Evaluating State Space Models as Vision Encoders

Shang-Jui Ray Kuo, Paola Cascante-Bonilla
{skuo, paola}@cs.stonybrook.edu

**Stony Brook University**

Project Page Repository

## Abstract

Large vision–language models (VLMs) often use a frozen vision backbone, whose image features are mapped into a large language model through a lightweight connector. While transformer-based encoders are the standard visual backbone, we ask whether state space model (SSM) vision backbones can be a strong alternative. We systematically evaluate SSM vision backbones for VLMs in a controlled setting. Under matched ImageNet-1K initialization, the SSM backbone achieves the strongest overall performance across both VQA and grounding/localization. We further adapt both SSM and ViT-family backbones with detection or segmentation training and find that dense-task tuning generally improves performance across families; after this adaptation, the SSM backbone remains competitive while operating at a substantially smaller model scale. We further observe that (i) higher ImageNet accuracy or larger backbones do not reliably translate into better VLM performance, and (ii) some visual backbones are unstable in localization. Based on these findings, we propose stabilization strategies that improve robustness for both backbone families and highlight SSM backbones as a strong alternative to transformer-based vision encoders in VLMs.

## 1. Introduction

Recent progress in vision–language models (VLMs) commonly follows a modular design: a pretrained vision encoder produces visual tokens, a lightweight connector maps them into the embedding space of a large language model (LLM), and the combined system is instruction-tuned for open-ended generation [22, 18, 37]. Typically, the vision encoder is kept fixed during instruction tuning, updating only the connector and the LLM [14, 39, 2]. While some systems do finetune the vision encoder, doing so reliably requires careful optimization choices and can be unstable under standard instruction-tuning recipes [35]; moreover, it obscures controlled comparisons of vision backbones by entangling architectural effects with training dynamics. Therefore, freezing the vision backbone enables different backbones to be evaluated under matched multimodal training without entangling architectural effects with joint vision-language optimization [6, 30, 14].

Despite extensive work on VLM training recipes, the vision encoder remains relatively narrow in architectural choice. Most systems still rely on ViT-family [8], or broadly transformer-based encoders, as the vision backbone [14]. At the same time, many comparisons change multiple ingredients together, including the vision pretraining objective, the multimodal training pipeline, resolution and tokenization settings, and connector design. This makes it difficult to isolate what is due to the vision architecture itself and whether the choice of backbone family limits how much useful evidence can be delivered from the vision encoder [30, 6].

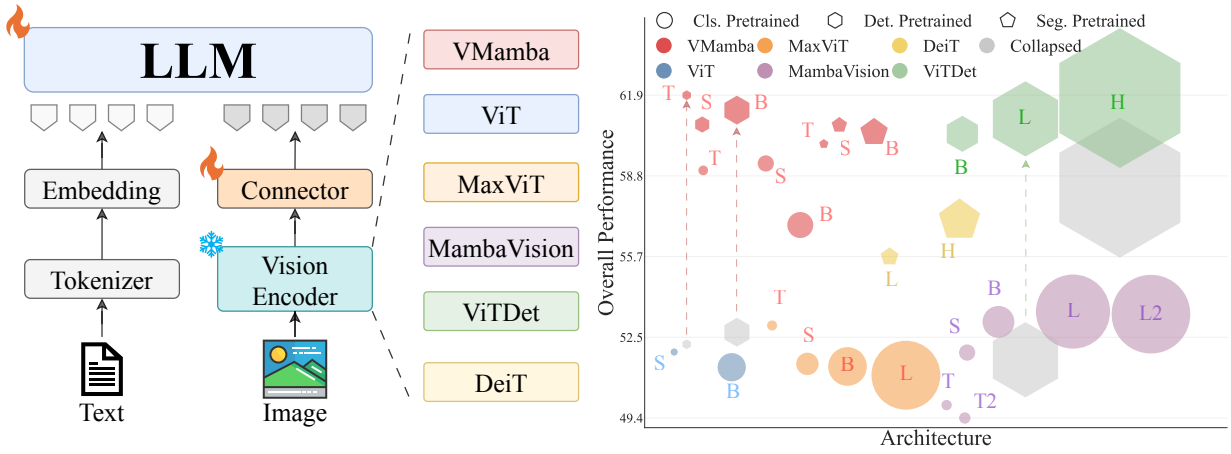


Figure 1 | **Overview of our controlled vision encoder study.** We follow a LLaVA-style VLM design where an input image is encoded by a frozen vision encoder into visual tokens, which are then processed by the LLM; this modular setup enables controlled swaps of vision encoders from different architecture families under a fixed training recipe. **Right:** Each marker summarizes one evaluated vision-backbone checkpoint plugged into the same VLM setting. Colors denote the backbone family; marker shapes denote the pretraining objective (classification, detection, or segmentation); marker size reflects encoder scale. Gray markers indicate configurations that exhibit collapse; arrows point to the corresponding stabilized variants after applying our stabilizations.

Another recurring challenge in VLMs is extracting spatially grounded evidence from images under a fixed multimodal token budget [11, 5]. To capture fine details, VLMs often increase image resolution or the number of visual tokens, but this quickly raises compute and memory costs in both the vision encoder and the LLM [19, 35]. This limitation raises an interesting open question: *is there a better visual representation that also encodes richer spatial information without increasing the number of vision tokens?*

State-space model (SSM) vision backbones have recently shown promising results in vision tasks, including competitive classification and particularly strong performance on dense prediction tasks [24, 40, 10]. Unlike ViTs, which rely on global self-attention over a flattened token sequence, many SSM vision models build representations through structured state-space updates implemented as multi-directional scans over the 2D grid [24, 38, 36]. These properties make SSM backbones plausible candidates for producing visual tokens that retain fine spatial information, which is important when the LLM must reason over localized details. To our knowledge, prior VLM work has not performed controlled backbone swaps that include SSM vision encoders while matching both the training recipe and the vision–language interface.

In this work, we use VMamba [24] as a strong pure-SSM backbone baseline, given its 2D-Selective-Scan (SS2D) design, which shows strong performance on dense vision tasks. Starting from ImageNet-1K (IN1K) [7] supervised initialization, we compare vision backbones in a controlled LLaVA-style setting [22], and then adapt SSM and ViT backbones with detection or segmentation pretraining. Building on this *backbone*-controlled swap, we further analyze two factors that affect outcomes: the vision pretraining *objective* (classification vs. dense objectives such as detection/segmentation), and the vision–language *interface* (input resolution/geometry and connector settings). Across open-ended VQA and localization benchmarks, we find that SSM-based vision encoders improve localization performance under matched settings while remaining competitive on VQA, and can match or surpass substantially

larger backbones on localization and grounding benchmarks. We further observe that standard vision metrics and naive backbone scaling can misrank VLM performance, and that some *backbone–interface* combinations are sensitive under certain resolution/geometry settings; these findings motivate simple stabilizations and practical guidance for selecting and deploying vision encoders in grounding-sensitive regimes. The overall framework of this paper is shown in Figure 1.

Our contributions are summarized as follows: (i) A controlled evaluation of frozen VLM vision encoders via backbone swaps across Transformer, SSM, and hybrid architectures, together with targeted analyses of pretraining objective and interface choices. (ii) Empirical evidence that SSM-based vision encoders (VMamba) improve localization under matched settings while remaining competitive on open-ended VQA. (iii) Systematic experiments reveal overlooked failure modes and show they are fixable (e.g., pretraining objective and visual model size do not always correlate with the overall VLM performance). (iv) We introduce a backbone–objective–interface exploration in the VLM design, and highlight SSM vision encoders as an underexplored, strong alternative.

## 2. Preliminaries

Our goal is to understand how the choice of vision backbone affects VLM behavior. To attribute differences to the vision encoder rather than to other confounding factors, we keep the rest of the VLM pipeline identical across experiments and swap only the vision backbone checkpoint. Our experimental framework largely follows [14]; unless we explicitly note a change or highlight a detail for clarity, we use the same setup. In this section, we summarize the settings shared by all experiments: (1) VLM architecture and notation, (2) optimization, (3) training data and preprocessing, (4) training implementation, and (5) evaluation suite.

### 2.1. Model Architecture and Notation.

We adopt a VLM architecture [22] consisting of a vision encoder, a lightweight connector, and a decoder-only language model (Vicuna-7B). A VLM takes an image $x_{\text{img}} \in \mathbb{R}^{H\times W\times 3}$ and a text prompt $p_{\text{prompt}}$ in natural language form.

*Vision Encoder.* First, the input image $x_{\text{img}}$ is fed into vision encoder $V_\omega$ which extract features from the image and output a sequence of tokens $f_{\text{img}} \in \mathbb{R}^{L\times d_{\text{vision}}}$ where $f_{\text{img}} = V_\omega(x_{\text{img}})$, $L$ is the number of visual tokens, and $d_{\text{vision}}$ is the dimension of a token. $L$ and $d_{\text{vision}}$ are defined by each vision backbones and depend on $H$ and $W$.

*Connector.* After we get the visual tokens $f_{\text{img}}$ from the vision encoder, we map these tokens into the LLM embedding space $d_{\text{text}}$ using a connector $C_\psi$. The output visual embeddings are $e_{\text{img}} \in \mathbb{R}^{L\times d_{\text{text}}}$, where $e_{\text{img}} = C_\psi(f_{\text{img}})$. Unless otherwise specified, the connector is defined as $C_\psi(x) = W_2\,\text{GELU}(W_1 x + b_1) + b_2$ where $W_1 \in \mathbb{R}^{d_{\text{text}}\times d_{\text{vis}}}$, $W_2 \in \mathbb{R}^{d_{\text{text}}\times d_{\text{text}}}$, and $b_1, b_2 \in \mathbb{R}^{d_{\text{text}}}$.

*Language Model.* Let $e_{\text{prompt}} \in \mathbb{R}^{N\times d_{\text{text}}}$ denote the prompt embeddings obtained by applying the LLM's tokenizer and embedding layer to the text prompt $u_{\text{prompt}}$. We concatenate the visual and text embeddings along the sequence dimension: $\mathbf{e}_{\text{input}} = [e_{\text{img}}; e_{\text{prompt}}] \in \mathbb{R}^{(L+N)\times d_{\text{text}}}$. The decoder-only language model $f_\theta$ then consumes $\mathbf{e}_{\text{input}}$ and autoregressively generates the output text $u_{\text{gen}} = f_\theta(\mathbf{e}_{\text{input}})$.

### 2.2. Optimization.

*Optimization.* Prior work finds that one-stage instruction tuning (i.e., jointly training a randomly initialized connector with the LLM) is more efficient and yields better performance than a two-stage pipeline that first aligns the connector and then performs joint tuning [14]. Following this recipe, we initialize the vision encoder $V_\omega$ and language model $f_\theta$ from pretrained checkpoints, and randomly initialize the connector $C_\psi$. During training, we freeze $\omega$ and update only $(\theta, \psi)$ via instruction tuning. We fix all optimization hyperparameters (optimizer, learning-rate schedule, batch size, number of steps, and precision) and the random seed across experiments; full details are provided in the Appendix A.

### 2.3. Training

*Data and Preprocessing.* We fine-tune on 665K multimodal instruction-tuning examples [14]; a detailed breakdown is provided in the appendix.

For image preprocessing, we apply letterbox resizing, which preserves the original aspect ratio by scaling the image and padding to the target resolution.

*Training Implementation.* We base our training on a verified codebase *, using Fully Sharded Data Parallel (FSDP) on 4× NVIDIA H200 GPUs with a fixed batch order.

### 2.4. Evaluation Suite.

Our evaluation for all the experiments covers two benchmark groups: **VQA** (VQA-v2 [9], GQA [12], VizWiz [3], TextVQA [29], POPE [21], TallyQA [1]) and **localization** (RefCOCO, RefCOCO+, RefCOCOg [15], OCID-Ref [34]). All pre-/post-processing and dataset-specific thresholds follow [14]. We also report the average VQA score, average localization score, and an overall average across all benchmarks weighted by the number of samples in each benchmark, to summarize VQA, localization, and overall performance for each checkpoint.

## 3. Investigating Different Vision Encoders

We report results in two regimes. First, we compare vision encoders under a strictly matched setting to isolate architectural effects in 3.1. Second, we evaluate detection- and segmentation-adapted checkpoints to study the impact of dense objectives in 3.2. Then, we summarize our observations in 3.3. Additional unmatched comparisons against larger data and larger-scale baselines are included in Appendix F.

### 3.1. Matched IN1K/224 Backbone Swaps

To compare IN1K-pretrained VMamba under a matched backbone-swap setting, we include three representative baselines from distinct architecture families. ViT [8] tokenizes an image into fixed-size patches and applies global self-attention over the patch sequence. MaxViT [32] is a hierarchical hybrid that combines convolutions with multi-axis attention (blocked local and dilated global attention) to capture local and global interactions. MambaVision [10] is a hybrid Mamba–Transformer backbone that adapts Mamba blocks for vision and retains self-attention in the final layers to capture long-range spatial dependencies. These backbones serve as strong reference points within their respective families.

*https://github.com/TRI-ML/prismatic-vlms

To enforce a strictly matched setup that isolates backbone architecture, we use checkpoints pretrained on IN1K at 224×224 resolution across all families. For multi-stage backbones (i.e., VMamba, MaxViT, and MambaVision), we extract features from the stage that yields the same number of visual tokens as ViT ($L$=196). In Appendix B, we further show that this choice also gives the best performance for these multi-stage backbones among plausible extraction stages.

Table 1 | **Matched IN1K/224 backbone swaps (VQA).** ImageNet-1K supervised vision encoders plugged into the same VLM under a strictly matched setting with 224×224 inputs and $L$=196 visual tokens. We report per-benchmark VQA scores and the weighted average VQA score. Across all tables, the best results are shown in bold, and the second best are underlined.

| Visual Encoder | Encoder Size | IN1K Acc | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|
| ViT-S | 22M | 78.8 | 59.86 | 50.48 | 46.69 | 44.02 | **12.30** | 77.76 | 48.96 | 57.25 |
| ViT-B | 87M | 81.1 | 59.63 | 49.19 | 45.63 | 44.51 | 12.07 | 75.90 | 50.57 | 57.17 |
| MaxViT-T | 31M | 83.41 | 61.08 | 49.43 | 47.58 | 44.31 | 12.02 | 76.77 | 53.85 | 58.75 |
| MaxViT-S | 69M | 84.43 | 59.90 | 48.94 | 41.50 | 44.51 | 11.82 | 76.29 | 51.42 | 57.42 |
| MaxViT-B | 119M | 84.85 | 60.10 | 49.72 | 45.00 | <u>44.59</u> | <u>12.29</u> | 76.26 | 50.94 | 57.60 |
| MaxViT-L | 212M | 84.93 | 60.07 | 49.55 | 46.42 | 43.67 | 12.23 | 76.46 | 48.86 | 57.30 |
| MambaVision-T | 32M | 82.3 | 56.85 | 47.43 | 46.16 | 43.74 | 11.14 | 71.34 | 46.89 | 54.38 |
| MambaVision-T2 | 35M | 82.7 | 56.69 | 47.14 | 43.17 | 43.31 | 10.82 | 69.58 | 43.74 | 53.71 |
| MambaVision-S | 50M | 83.3 | 58.10 | 48.05 | 40.91 | 43.08 | 11.36 | 72.03 | 49.48 | 55.61 |
| MambaVision-B | 98M | 84.2 | 59.14 | 49.23 | 44.21 | 43.33 | 11.43 | 73.05 | 49.53 | 56.53 |
| MambaVision-L | 228M | 85 | 59.26 | 49.56 | 42.13 | 43.75 | 11.41 | 74.46 | 51.70 | 56.94 |
| MambaVision-L2 | 242M | 85.3 | 59.27 | 48.83 | **48.91** | 43.30 | 11.16 | 74.15 | 50.70 | 56.86 |
| VMamba-T | 30M | 82.6 | <u>64.99</u> | <u>54.02</u> | 44.96 | **44.62** | 12.22 | <u>81.70</u> | <u>54.58</u> | <u>62.07</u> |
| VMamba-S | 50M | 83.6 | **65.24** | **54.08** | 44.95 | 44.06 | 12.24 | **82.20** | **55.54** | **62.39** |
| VMamba-B | 80M | 83.9 | 64.20 | 53.25 | <u>47.97</u> | 43.81 | 12.08 | 80.75 | 54.05 | 61.38 |

Table 2 | **Matched IN1K/224 backbone swaps (localization and overall).** Same setting as 1, reporting per-benchmark localization/grounding scores, the weighted average localization score, and the weighted overall average across all benchmarks.

| Visual Encoder | Encoder Size | IN1K Acc | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|
| ViT-S | 22M | 78.8 | 32.32 | 21.77 | 24.80 | 5.08 | 17.82 | 51.95 |
| ViT-B | 87M | 81.1 | 26.66 | 15.41 | 19.12 | 4.14 | 13.92 | 51.36 |
| MaxViT-T | 31M | 83.41 | 29.44 | 17.29 | 22.10 | 5.17 | 15.79 | 52.98 |
| MaxViT-S | 69M | 84.43 | 25.57 | 14.41 | 17.28 | 4.38 | 13.32 | 51.49 |
| MaxViT-B | 119M | 84.85 | 22.02 | 12.67 | 15.28 | 3.36 | 11.41 | 51.39 |
| MaxViT-L | 212M | 84.93 | 21.15 | 12.01 | 14.77 | 2.94 | 10.81 | 51.05 |
| MambaVision-T | 32M | 82.3 | 34.59 | 24.01 | 27.14 | 9.52 | 20.98 | 49.89 |
| MambaVision-T2 | 35M | 82.7 | 35.78 | 24.69 | 27.61 | 9.72 | 21.56 | 49.39 |
| MambaVision-S | 50M | 83.3 | 44.65 | 33.06 | 37.48 | 13.21 | 28.22 | 51.93 |
| MambaVision-B | 98M | 84.2 | 48.52 | 35.57 | 40.58 | 15.84 | 31.17 | 53.12 |
| MambaVision-L | 228M | 85 | 48.19 | 35.82 | 40.26 | 16.79 | 31.51 | 53.52 |
| MambaVision-L2 | 242M | 85.3 | 47.91 | 35.41 | 40.28 | 16.48 | 31.22 | 53.42 |
| VMamba-T | 30M | 82.6 | **58.25** | **46.64** | **51.74** | <u>20.24</u> | **39.20** | <u>59.00</u> |
| VMamba-S | 50M | 83.6 | <u>56.48</u> | <u>44.27</u> | <u>49.88</u> | **23.09** | <u>39.17</u> | **59.27** |
| VMamba-B | 80M | 83.9 | 42.06 | 31.43 | 36.15 | 15.23 | 27.89 | 56.88 |

*Observations.* Under the matched IN1K/224 backbone setting in Table 1 and Table 2, VMamba is the strongest across its T/S/B variants, showing better overall performance. In addition, VMamba-T/S consistently outperforms other methods on grounding across all localization benchmarks. For VLMs with ViT and MaxViT backbones, higher IN1K accuracy consistently corresponds to lower VLM performance. In contrast, VLMs with VMamba and MambaVision backbones improve with scaling at small sizes, but show the same degradation at larger scales.

### 3.2. Dense Objectives Pretrained Backbone Comparisons

We next evaluate dense-objective checkpoints with higher resolutions. Alongside VMamba, we include two dense-task baselines: ViTDet [20], which adapts a plain ViT backbone for object detection and shows that a simple feature pyramid from a single-scale feature map can suffice for detection fine-tuning, and DeiT [31] checkpoints adapted with the ViT-Adapter framework [4], which adds a pre-training-free adapter to a plain ViT to introduce image-specific inductive biases for dense prediction. Concretely, we use VMamba and ViTDet pretrained on IN1K and then fine-tuned for detection, and VMamba and DeiT pretrained on IN1K and then fine-tuned for segmentation. Because these checkpoints differ in input geometry and feature extraction stages, they generally produce different output token lengths $L$. We therefore treat these results as evidence about dense pretraining objective effects, rather than as perfectly matched architectural comparisons.

Table 3 | **Dense-objective checkpoints at pretraining resolution (VQA).** We compare vision encoders adapted with dense objectives, including detection-pretrained (IN1K→COCO) ViTDet as a Transformer baseline and VMamba as the SSM backbone, and segmentation-pretrained (IN1K→ADE20K) DeiT baselines and VMamba. All checkpoints are evaluated at their pretraining input geometries and we report per-benchmark VQA scores and the average VQA score. Because input geometry (and thus token length) differs across entries, these results reflect the impact of the dense pretraining objective rather than perfectly matched architectural swaps.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN1K → COCO | ViTDet-B | 111M | 1024x1024 | 4096 | **65.83** | 53.61 | **50.58** | 43.75 | 11.60 | 84.46 | 55.96 | **63.00** |
| IN1K → COCO | ViTDet-L | 331M | 1024x1024 | 4096 | 60.00 | 48.59 | 44.78 | 43.88 | 11.78 | 78.14 | 51.94 | 57.64 |
| IN1K → COCO | ViTDet-H | 662M | 1024x1024 | 4096 | 64.43 | 52.62 | 48.99 | 43.93 | 11.50 | 84.29 | 55.95 | 61.89 |
| IN1K → COCO | VMamba-T | 30M | 1333x800 | 4150 | 60.24 | 49.18 | 49.04 | 44.06 | 11.81 | 79.48 | 52.65 | 58.05 |
| IN1K → COCO | VMamba-S | 50M | 1333x800 | 4150 | 64.87 | **53.98** | 47.73 | **44.24** | **11.93** | **86.47** | **59.21** | 62.78 |
| IN1K → COCO | VMamba-B | 89M | 1333x800 | 4150 | 60.36 | 49.21 | 46.46 | 43.84 | 11.73 | 81.61 | 55.69 | 58.57 |
| IN1K → ADE20K | DeiT-S | 58M | 512x512 | 256 | 61.56 | 52.37 | **52.16** | 44.25 | 11.74 | 79.82 | 53.60 | 59.36 |
| IN1K → ADE20K | DeiT-B | 134M | 512x512 | 256 | 63.37 | 53.32 | 49.40 | 43.98 | 11.97 | 81.64 | 53.11 | 60.69 |
| IN1K → ADE20K | VMamba-T | 30M | 512x512 | 1024 | 65.45 | 55.26 | 40.91 | 44.18 | 11.85 | 83.65 | **55.66** | 62.60 |
| IN1K → ADE20K | VMamba-S | 50M | 512x512 | 1024 | **66.42** | 55.68 | 47.44 | 44.42 | 11.86 | 84.01 | 53.91 | **63.21** |
| IN1K → ADE20K | VMamba-B | 89M | 512x512 | 1024 | 66.12 | **55.89** | 40.19 | **44.50** | **12.33** | **84.39** | 53.61 | 62.87 |

*Observations.* In Table 3 and Table 4, detection adaptation can improve both VQA and localization when fine-tuning remains stable (e.g., ViTDet-B and VMamba-S), but it can also exhibit sharp localization degradation (notably ViTDet-L/H and VMamba-T/B), which we refer to as *localization collapse*. In contrast, segmentation adaptation yields more consistently strong localization across scales: segmentation-adapted VMamba remains strong across sizes and generally outperforms the DeiT (ViT-Adapter) baselines.

Table 4 | **Dense-objective checkpoints at pretraining resolution (localization and averages).** Same settings as the VQA table, reporting localization/grounding benchmarks, the average localization score, and the overall average across all benchmarks. Checkpoints are evaluated at their pretraining input geometries; since geometry (and token length) differs across entries, the comparisons primarily isolate the effect of dense objectives rather than matched backbone architecture.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN1K → COCO | ViTDet-B | 111M | 1024x1024 | 4096 | 66.03 | 51.17 | 58.17 | 22.37 | 43.74 | 60.42 |
| IN1K → COCO | ViTDet-L | 331M | 1024x1024 | 4096 | 24.62 | 13.79 | 17.44 | 4.62 | 13.05 | 51.65 |
| IN1K → COCO | ViTDet-H | 662M | 1024x1024 | 4096 | 56.41 | 40.48 | 50.25 | 16.01 | 35.38 | 58.33 |
| IN1K → COCO | VMamba-T | 30M | 1333x800 | 4150 | 29.10 | 16.83 | 19.89 | 3.95 | 14.86 | 52.25 |
| IN1K → COCO | VMamba-S | 50M | 1333x800 | 4150 | **69.52** | **52.87** | **63.50** | **28.15** | **47.94** | **60.78** |
| IN1K → COCO | VMamba-B | 89M | 1333x800 | 4150 | 28.43 | 16.44 | 19.87 | 4.97 | 15.02 | 52.72 |
| IN1K → ADE20K | DeiT-S | 58M | 512x512 | 256 | 49.52 | 37.07 | 42.46 | 15.35 | 31.78 | 55.65 |
| IN1K → ADE20K | DeiT-B | 134M | 512x512 | 256 | 52.17 | 39.78 | 45.61 | 16.22 | 33.77 | 57.07 |
| IN1K → ADE20K | VMamba-T | 30M | 512x512 | 1024 | 62.65 | 51.10 | 57.15 | 24.01 | 43.47 | 60.03 |
| IN1K → ADE20K | VMamba-S | 50M | 512x512 | 1024 | **64.17** | **53.98** | **59.13** | 24.58 | 44.98 | **60.76** |
| IN1K → ADE20K | VMamba-B | 89M | 512x512 | 1024 | 63.99 | 52.52 | 58.68 | **25.91** | **45.08** | 60.48 |

### 3.3. Summary of Observations

Across the matched backbone-swap and dense-objective regimes, we observe: (1) Under strictly matched IN1K/224, $L$=196 swaps, VMamba achieves the strongest overall performance, with VMamba-T/S consistently leading localization across all grounding benchmarks, and VMamba variants achieve top aggregate VQA. (2) Dense pretraining objectives (detection/segmentation) can improve both VQA and localization for SSM- and Transformer-based vision encoders. (3) ImageNet accuracy and naive backbone scaling might not directly improve downstream VLM performance, and can even degrade. (4) Some detection-pretrained configurations exhibit sharp localization degradation (*localization collapse*). Section 4 analyzes these patterns and their causes, and presents practical stabilizations.

## 4. Analysis, Diagnosis, and Stabilizations

We first establish the relationship between VQA and localization metrics in 4.1, which provides the basis for the analyses in this section. We now analyze the empirical patterns from Sec. 3: (i) why VMamba-T/S consistently lead localization and why VMamba achieves the strongest overall performance under matched IN1K/224 swaps in Sec. 4.2; (ii) how dense pretraining objectives (detection/segmentation) affect VQA and localization across backbone families in Sec. 4.3; and (iii) two failure modes—ImageNet/scale being an unreliable predictor of downstream VLM behavior, and sharp localization degradation (*localization collapse*) in some high-resolution detection-adapted settings in Sec. 4.4. We close with practical implications that motivate the stabilizations in Sec. 4.5.

### 4.1. Localization Matters for General VQA

General open-ended VQA benchmarks implicitly require models to localize relevant objects and regions. For example, GQA is constructed around scene graphs with explicit grounding and evaluates grounding quality, and several works show that improving visual grounding or spatial localization leads to better VQA performance on both GQA and VQA-v2 [12, 16, 27].

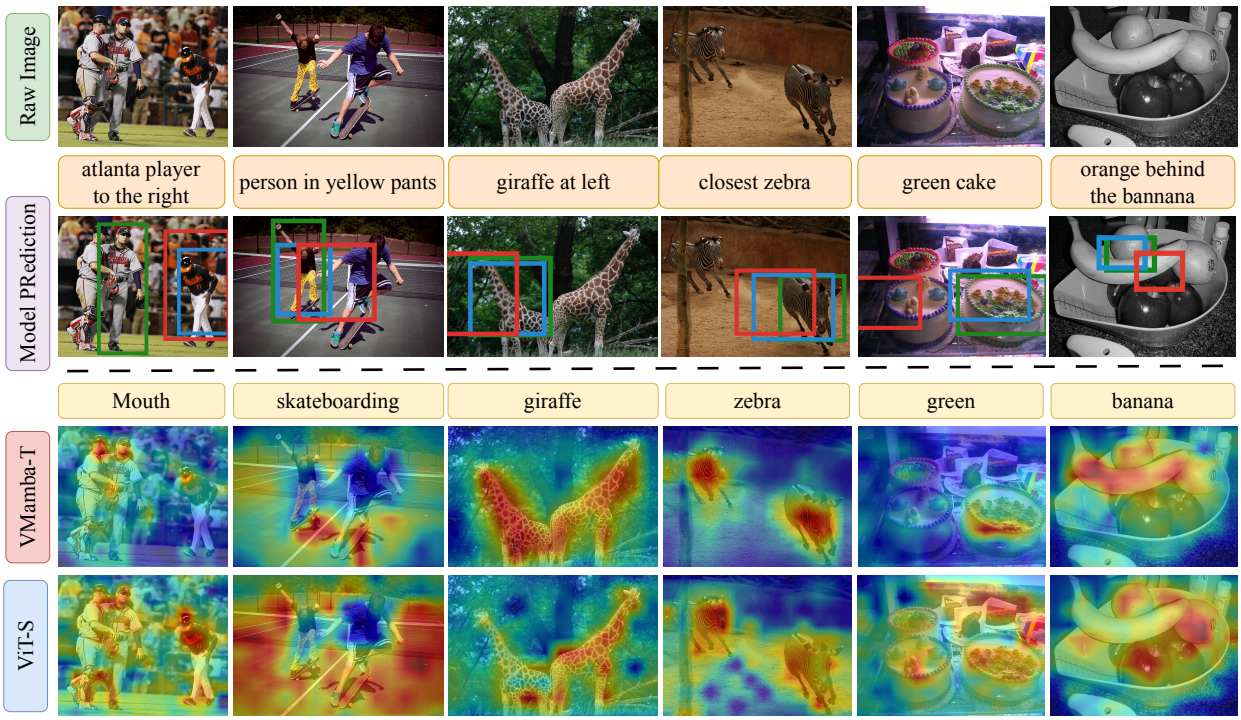


Figure 2 | **(Top) Grounding examples.** Under the matched IN1K/224 setting, VMamba-T (blue) predicts boxes closer to ground truth (green) than ViT-S (red). **(Bottom) Token–region similarity.** Similarity maps between the visual feature map and the corresponding text tokens from an intermediate LLM layer show VMamba-T yields sharper, more spatially localized text–vision alignment than ViT-S, indicating better preservation and utilization of spatial information.

To quantify the relationship between VQA and localization in our setting, we compute a Pearson correlation matrix across the metrics reported for all the checkpoint runs. For each run, we take its per-benchmark scores and compute Pearson $r$ between every pair of metrics. The correlation matrix shows that the localization benchmarks are highly consistent with each other, and that VQA-v2, GQA, POPE, and TallyQA correlate moderately to strongly with localization (from 0.65 to 0.80), while TextVQA and VizWiz correlate weakly (from 0.23 to 0.50). The detailed correlation matrix numbers are provided in D. Overall, these trends support treating localization as a primary axis when comparing frozen vision encoders, and suggest that differences that are subtle on VQA can be amplified on localization benchmarks.

### 4.2. VMamba is Strong Under the Matched IN1K/224 Setting

We next analyze why VMamba performs strongly on localization in the matched IN1K/224 setting. VMamba's SS4D layer applies state-space aggregation along the 2D token grid in four scan directions; each patch receives four directional state updates that propagate information across rows and columns. This four-directional aggregation is repeated throughout the network, so spatial interactions are baked into the architecture and can preserve spatial structure even under standard classification pretraining without an explicit localization objective. In the ViT's case, in contrast, the Transformer is permutation-invariant with respect to token order, so spatial structure is carried mainly by positional encodings. Without an objective that explicitly rewards spatial fidelity, this positional information can be underutilized or attenuated in deeper layers under standard ImageNet pretraining [33, 13].

To provide qualitative support, we visualize token–region similarity maps (Figure 2, bottom). In the baseball example (first column), VMamba-T assigns high similarity tightly to the small, discriminative region referenced by the text (Mouth), whereas ViT-S produces a more diffuse response over larger body regions. Across the other examples, VMamba-T consistently yields sharper, more concentrated peaks on the target object, while ViT-S tends to spread high responses across multiple regions and sometimes across multiple objects, suggesting weaker spatial selectivity.

This pattern is consistent with the grounding predictions in Figure 2 (top): VMamba-T's boxes more closely match the ground-truth regions than ViT-S. Even in the failure case in the first column (selecting the wrong player), VMamba-T still produces a tighter box around the predicted referent, matching the more localized similarity structure observed in the corresponding similarity map. These qualitative results align with the quantitative localization gains in Table 2.

### 4.3. Dense-Task Objectives Help

Since VLM performance depends on spatially grounded visual representations, adapting ImageNet-classification backbones with dense objectives such as detection and segmentation is a natural way to strengthen spatial fidelity. These dense objectives directly supervise spatial layouts and region-level discrimination, and in our experiments this often translates into gains on both localization and VQA, provided the resulting VLM interface remains stable (Sec. 3.2).

This effect is most pronounced for ViT-family baselines that lack built-in spatial inductive bias: detection-adapted ViTDet and segmentation-adapted DeiT substantially improve over their classification-pretrained ViT counterparts. VMamba also benefits from dense-objective adaptation, but the improvements are typically more moderate, which is consistent with VMamba already preserving spatial structure under IN1K pretraining through its architectural bias.

Finally, we observe a similar pattern for non-classification pretraining, where combining contrastive-pretrained backbones (e.g., CLIP, SigLIP) with SSL pretrained backbones (e.g., DINOv2) yields the strongest results in our additional experiments, matching the trend reported by prior work [14]. However, comparable contrastive/SSL SSM-based backbones are not currently available; thus, we treat this direction as out of scope for the main paper and report details in Appendix F.

### 4.4. Diagnosing Failures

We now diagnose two failure modes surfaced in Sec. 3. First, ImageNet top-1 accuracy and naive backbone scaling are unreliable predictors for downstream VLM quality. Second, some dense-objective checkpoints exhibit *localization collapse*, where grounding performance drops abruptly after detection-style adaptation.

#### *4.4.1. Model Size and ImageNet Analysis*

Table 1 and Table 2 contain clear counterexamples where larger models with higher ImageNet accuracy fail to predict VLM performance. MaxViT-L has higher IN1K accuracy than MaxViT-T/S/B but achieves much worse localization. Similarly, scaling from VMamba-T/S to VMamba-B does not improve overall performance.

At first glance, overfitting on the ImageNet might be the reason why larger models reduce transferability at scale; we evaluate linear probing across diverse vision classification datasets and find evidence of degraded transfer for some large supervised models, but there exist models that transfer

well and still have poor VLM performance. Thus, probing alone does not fully explain the VLM ranking. We provide the detailed probing results in Appendix E.

Inspired by the dense-tasks pretraining analysis, instead of overfitting on ImageNet-1K, the larger models are more likely to overfit on the *classification objective*, which means that their features only retain the information for classifying the salient object and abandon most of the spatial information. This can explain why even when a visual backbone shows strong accuracy on ImageNet-1K and generalizes well on other classification datasets, it can still perform poorly on the VQA and localization benchmarks.

As observed in Table 1 and Table 2, visual backbones with state-space layers, such as VMamba and MambaVision, are more resistant to this *objective overfitting* as their 2D inductive can retain part of the spatial ability. However, as the model size increases, the parameters could still overpower the architectural induction bias, and the localization numbers still drop.

#### *4.4.2. Localization Collapse as an Interface Failure Mode*

Some detection-adapted results reveal sharp localization collapses that are difficult to reconcile with naive scaling expectations. For example, ViTDet-L/H collapses despite ViTDet-B being strong, and VMamba-T/B collapse while VMamba-S is strong.

A natural question is whether these failures are simply caused by poor visual feature extraction. However, this explanation is unconvincing: closely related variants in the same backbone family and objective setting can produce strong results, indicating that the underlying vision features can encode rich spatial information. This suggests that collapse is more likely driven by a failure in *vision–language transfer* interface, rather than a complete absence of spatial information in the vision encoder.

We consider two hypotheses. **(H1) Transmission bottleneck:** the spatial information is present in the features provided by the vision encoder but is not faithfully conveyed through the connector (e.g., the connector capacity is insufficient to preserve the relevant spatial structure when mapping into the LLM embedding space). **(H2) Utilization bottleneck:** even if the vision backbone encodes rich spatial information, and the connector faithfully conveys the spatial infromation form the vision encoder, the language model may not be able to reliably interpret and use these spatial cues for grounding. Both hypotheses exclude the vision encoder and implicate an interaction between the interface geometry (resolution and aspect ratio), the connector, and the LLM, which motivates the following stabilization strategies.

### 4.5. From Diagnosis to Stabilizations

#### *4.5.1. Stabilization Experiment Protocol*

Across all stabilization experiments, we keep the recipe identical as described in Sec. 2, and change only the factor under study. We evaluate stabilizations on representative collapse cases (i.e., detection-adapted DetViT-L/H and VMamba-T/B), and we include settings that are not collapsed as controls to verify that stabilizations do not degrade performance when collapse is not present.

Concretely, we test three strategies to separate transmission versus utilization bottlenecks: (i) increasing connector capacity to improve transmission of spatial structure into the LLM embedding space, (ii) modifying the interface geometry by changing image resolutions to reduce instability in dense-tasks adapted checkpoints, and (iii) combining the above two strategies to test whether the effects are complementary.

Table 5 | **Stabilizing localization collapse: VQA benchmarks.** We test two interface strategies on the collapse cases: increasing connector capacity with a stronger MLP connector (marked with (f)) and changing the input geometry to a square 512×512 setting while keeping the vision checkpoint fixed. Values in parentheses denote the change relative to the corresponding collapsed variant, which is highlighted in blue.

| Visual Encoder | Encoder Size | Image Size | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|
| ViTDet-B | 111M | 1024x1024 | 65.83 | 53.61 | **50.58** | 43.75 | 11.60 | 84.46 | 55.96 | 63.00 |
| ViTDet-L | 331M | 1024x1024 | 60.00 | 48.59 | 44.78 | 43.88 | 11.78 | 78.14 | 51.94 | 57.64 |
| ViTDet-L(f) | 331M | 1024x1024 | 66.43 (+6.43) | 53.54 (+4.95) | 45.52 (+0.74) | 43.76 (-0.12) | 11.58 (-0.20) | 84.82 (+6.68) | 57.24 (+5.30) | 63.55 (+5.91) |
| ViTDet-H | 662M | 1024x1024 | 64.43 | 52.62 | 48.99 | 43.93 | 11.50 | 84.29 | 55.95 | 61.89 |
| ViTDet-H(f) | 662M | 1024x1024 | 66.89 (+2.46) | 54.22 (+1.60) | 41.01 (-7.98) | 44.04 (+0.11) | 11.82 (+0.32) | 87.03 (+2.74) | 58.08 (+2.13) | 64.05 (+2.16) |
| VMamba-T | 30M | 1333x800 | 60.24 | 49.18 | 49.04 | 44.06 | 11.81 | 79.48 | 52.65 | 58.05 |
| VMamba-T(f) | 30M | 1333x800 | 64.46 (+4.22) | 53.33 (+4.15) | 45.21 (-3.83) | 43.68 (-0.38) | 11.90 (+0.09) | 84.14 (+4.66) | 54.27 (+1.62) | 61.66 (+3.61) |
| VMamba-T | 30M | 512x512 | **66.91** (+6.67) | 55.30 (+6.12) | 45.05 (-3.99) | 44.43 (+0.37) | 11.73 (-0.08) | 84.86 (+5.38) | 58.94 (+6.29) | 64.22 (+6.17) |
| VMamba-T(f) | 30M | 512x512 | 66.77 (+6.53) | **55.41** (+6.23) | 42.75 (-6.29) | 44.28 (+0.22) | 11.72 (-0.09) | 85.71 (+6.23) | 60.19 (+7.54) | **64.28** (+6.23) |
| VMamba-S | 50M | 1333x800 | 64.87 | 53.98 | 47.73 | 44.24 | 11.93 | 86.47 | 59.21 | 62.78 |
| VMamba-B | 89M | 1333x800 | 60.36 | 49.21 | 46.46 | 43.84 | 11.73 | 81.61 | 55.69 | 58.57 |
| VMamba-B(f) | 89M | 1333x800 | 60.85 (+0.49) | 50.25 (+1.04) | 40.44 (-6.02) | 43.99 (+0.15) | 11.77 (+0.04) | 82.71 (+1.10) | 55.04 (-0.65) | 58.84 (+0.27) |
| VMamba-B | 89M | 512x512 | 65.70 (+5.34) | 55.01 (+5.80) | 46.07 (-0.39) | 44.66 (+0.82) | 12.05 (+0.32) | **87.99** (+6.38) | 60.07 (+4.38) | 63.58 (+5.01) |
| VMamba-B(f) | 89M | 512x512 | 65.63 (+5.27) | 55.11 (+5.90) | 44.13 (-2.33) | **44.86** (+1.02) | **12.15** (+0.42) | 87.57 (+5.96) | **60.69** (+5.00) | 63.58 (+5.01) |

### *4.5.2. Transmission Bottleneck Test*

We first test whether localization collapse is caused by insufficient transmission capacity at the vision–language interface by increasing connector capacity while keeping all other settings fixed. Specifically, we replace the default 2-layer MLP connector with a 3-layer MLP, using the same initialization policy and training hyperparameters. Results are reported in Table 5 and Table 6; models using the stronger connector are marked with (f) in the visual encoder name.

Increasing connector capacity can substantially recover localization in some collapse cases, and the recovered localization typically comes with improved VQA. For example, ViTDet-L collapses under the default connector but recovers with the 3-layer connector, suggesting that the bottleneck lies primarily in the vision–language interface rather than the vision encoder itself. However, a stronger connector does not resolve all failures. For detection-pretrained VMamba-T, the stronger connector improves localization but still falls short of the ImageNet-pretrained VMamba-T baseline. For VMamba-B, the gain is marginal, and the collapse largely persists. These cases indicate that interface instability is not solely due to limited connector expressivity, motivating the geometry-based stabilizations studied next.

### *4.5.3. Utilization Bottleneck Test*

We next test a geometry-based stabilization motivated by a consistent difference across VMamba checkpoints: ImageNet-pretrained VMamba (stable) and segmentation-pretrained VMamba (stable) use square inputs, whereas detection-pretrained VMamba uses high-resolution non-square inputs. We hypothesize that the non-square geometry makes it harder for the language model to reliably utilize spatial cues from the visual tokens, contributing to localization collapse.

Keeping the detection-pretrained VMamba-T/B checkpoints and the full training recipe fixed, we evaluate a square input geometry (512×512) and compare against the original non-square setting. The square geometry eliminates the collapse behavior and improves both localization and VQA for detection-pretrained VMamba-T/B, outperforming their ImageNet-pretrained counterparts (Table 5, Table 6). VMamba-S shows a small degradation, which is expected given that 512×512 deviates from its pretraining resolution.

Table 6 | **Stabilizing localization collapse: localization benchmarks.** Same settings as Table 5, reporting grounding metrics and aggregate scores.

| Visual Encoder | Encoder Size | Image Size | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|
| ViTDet-B | 111M | 1024x1024 | 66.03 | 51.17 | 58.17 | 22.37 | 43.74 | 60.42 |
| ViTDet-L | 331M | 1024x1024 | 24.62 | 13.79 | 17.44 | 4.62 | 13.05 | 51.65 |
| ViTDet-L(f) | 331M | 1024x1024 | 65.73 (+41.11) | 52.09 (+38.30) | 58.86 (+41.42) | 23.87 (+19.25) | 44.58 (+31.53) | 61.00 (+9.35) |
| ViTDet-H | 662M | 1024x1024 | 56.41 | 40.48 | 50.25 | 16.01 | 35.38 | 58.33 |
| ViTDet-H(f) | 662M | 1024x1024 | 69.37 (+12.96) | **54.27** (+13.79) | 61.97 (+11.72) | 26.51 (+10.50) | 47.40 (+12.02) | 61.81 (+3.48) |
| VMamba-T | 30M | 1333x800 | 29.10 | 16.83 | 19.89 | 3.95 | 14.86 | 52.25 |
| VMamba-T(f) | 30M | 1333x800 | 57.10 (+28.00) | 42.55 (+25.72) | 50.10 (+30.21) | 20.64 (+16.69) | 37.93 (+23.07) | 58.47 (+6.22) |
| VMamba-T | 30M | 512x512 | 61.51 (+32.41) | 50.50 (+33.67) | 53.76 (+33.87) | 28.50 (+24.55) | 44.52 (+29.66) | 61.57 (+9.32) |
| VMamba-T(f) | 30M | 512x512 | 63.69 (+34.59) | 52.70 (+35.87) | 55.68 (+35.79) | 30.87 (+26.92) | 46.75 (+31.89) | **61.92** (+9.67) |
| VMamba-S | 50M | 1333x800 | **69.52** | 52.87 | **63.50** | 28.15 | **47.94** | 60.78 |
| VMamba-B | 89M | 1333x800 | 28.43 | 16.44 | 19.87 | 4.97 | 15.02 | 52.72 |
| VMamba-B(f) | 89M | 1333x800 | 31.47 (+3.04) | 19.10 (+2.66) | 22.24 (+2.37) | 3.95 (-1.02) | 16.23 (+1.21) | 53.11 (+0.39) |
| VMamba-B | 89M | 512x512 | 62.77 (+34.34) | 50.14 (+33.70) | 56.33 (+36.46) | 30.00 (+25.03) | 45.63 (+30.61) | 61.17 (+8.45) |
| VMamba-B(f) | 89M | 512x512 | 64.57 (+36.14) | 50.99 (+34.55) | 57.07 (+37.20) | **31.35** (+26.38) | 46.90 (+31.88) | 61.34 (+8.62) |

This sensitivity to geometry is consistent with recent findings that VLM behavior can be brittle to input geometry at high resolution [25, 17]. We leave a deeper mechanistic analysis of why square geometry improves utilization to future work.

#### *4.5.4. Applying Connector & Square Geometry Together*

Finally, we combine the two strategies above to test whether connector capacity and geometry address complementary parts of the failure. We apply the stronger connector and square 512x512 resolution settings on VMamba-T and VMamba-B. Table 5 and Table 6 show that the combined stabilization yields the greatest and most consistent improvements across the collapse cases. Empirically, this suggests that both transmission capacity and interface geometry contribute to stable vision–language transfer.

### 4.6. Summary: Backbone, Objective, and Interface

Taken together, our results suggest a factorized view where VLM performance is jointly determined by (i) backbone architecture, (ii) pretraining objective, and (iii) the vision–language interface (connector capacity and input geometry). Architectures with stronger spatial inductive bias and dense-task objectives both tend to improve grounding and VQA; however, these gains only translate into downstream VLM quality when the interface can stably transmit and utilize the spatial signal. When the interface is unstable, even strong backbone–objective combinations can exhibit localization collapse and become ineffective. Notably, factors (ii) and (iii) are largely architecture-agnostic and can benefit across different backbones, while factor (i) remains important for robustness under matched classification pretraining.

## 5. Related Works

**SSMs in Multimodal Models.** Recent work applies SSMs to vision–language systems primarily on the language and fusion side, where Mamba-style layers replace transformer-based text backbones or multimodal blocks to improve sequence modeling efficiency, while the vision encoder often remains

a ViT- or CNN-style backbone (e.g., Cobra [39], VL-Mamba [26]). More recent work such as CLIMP [28] explores fully Mamba-based contrastive vision–language pretraining by replacing both vision and text encoders with SSM architectures, focusing on representation learning rather than downstream generative VLM behavior. These approaches highlight that SSMs can serve as competitive and efficient components for multimodal modeling, but they do not study the effect of different vision backbone architectures under controlled VLM training settings. In parallel, MambaVLT [23] uses Mamba blocks for multimodal fusion and evaluates on classification tasks. In contrast, we evaluate SSM vision backbones (VMamba) as frozen vision towers in a generative VLM, isolating their effect on grounding and VQA under matched training conditions.

## 6. Conclusion

In a controlled LLaVA-style setting with frozen vision encoders, we show that SSM-based VMamba is a strong alternative to ViT-family encoders. First, we show that, under strictly matched IN1K/224 swaps, VMamba variants achieve the strongest overall performances, with VMamba-T/S consistently leading grounding benchmarks, while dense-task pretraining objectives (i.e., detection and segmentation) further improve both VQA and localization across backbone families. At the same time, we find that ImageNet accuracy and naive scaling are unreliable predictors for downstream VLM quality, and that some high-resolution detection-pretrained configurations can suffer sharp localization collapse. We diagnose collapse as a vision–language interface failure mode and show that simple, architecture-agnostic stabilizations such as increasing connector capacity and adjusting interface geometry can be complementary and recover both localization and overall performance. Taken together, the results support a backbone, objective, and interface view of VLM design and suggest that SSM-based vision encoders are a promising, size-efficient backbone choice for VLMs.

**Acknowledgments.**
This material is based upon work supported by the SUNY AI Platform on Google Cloud (Phase 2).

## References

[1] Acharya, M., Kafle, K., Kanan, C.: Tallyqa: Answering complex counting questions. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 8076–8084 (2019)

[2] Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)

[3] Bigham, J.P., Jayant, C., Ji, H., Little, G., Miller, A., Miller, R.C., Miller, R., Tatarowicz, A., White, B., White, S., et al.: Vizwiz: nearly real-time answers to visual questions. In: Proceedings of the 23nd annual ACM symposium on User interface software and technology. pp. 333–342 (2010)

[4] Chen, Z., Duan, Y., Wang, W., He, J., Lu, T., Dai, J., Qiao, Y.: Vision transformer adapter for dense predictions. arXiv preprint arXiv:2205.08534 (2022)

[5] Cheng, A.C., Yin, H., Fu, Y., Guo, Q., Yang, R., Kautz, J., Wang, X., Liu, S.: Spatialrgpt: Grounded spatial reasoning in vision-language models. Advances in Neural Information Processing Systems **37**, 135062–135093 (2024)

[6] Cocchi, F., Moratelli, N., Caffagni, D., Sarto, S., Baraldi, L., Cornia, M., Cucchiara, R.: Llava-more: A comparative study of llms and visual backbones for enhanced visual instruction tuning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4278–4288 (2025)

[7] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009). https://doi.org/10.1109/CVPR.2009.5206848

[8] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)

[9] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D.: 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)

[10] Hatamizadeh, A., Kautz, J.: Mambavision: A hybrid mamba-transformer vision backbone. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 25261–25270 (2025)

[11] Huang, Y., Ma, F., Shao, Y., Guo, J., Yu, Z., Cui, L., Tian, Q.: N\" uwa: Mending the spatial integrity torn by vlm token pruning. arXiv preprint arXiv:2602.02951 (2026)

[12] Hudson, D.A., Manning, C.D.: 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)

[13] Jelassi, S., Sander, M., Li, Y.: Vision transformers provably learn spatial structure. Advances in Neural Information Processing Systems **35**, 37822–37836 (2022)

[14] Karamcheti, S., Nair, S., Balakrishna, A., Liang, P., Kollar, T., Sadigh, D.: Prismatic vlms: Investigating the design space of visually-conditioned language models. In: International Conference on Machine Learning (ICML) (2024)

[15] Kazemzadeh, S., Ordonez, V., Matten, M., Berg, T.: 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)

[16] Khan, A.U., Kuehne, H., Gan, C., Lobo, N.D.V., Shah, M.: Weakly supervised grounding for vqa in vision-language transformers. In: European Conference on Computer Vision. pp. 652–670. Springer (2022)

[17] Leroy, V., Revaud, J., Lucas, T., Weinzaepfel, P.: Win-win: Training high-resolution vision transformers from two windows. arXiv preprint arXiv:2310.00632 (2023)

[18] Li, J., Li, D., Savarese, S., Hoi, S.: 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 (2023)

[19] Li, K., Goyal, S., Semedo, J.D., Kolter, J.Z.: Inference optimal VLMs need fewer visual tokens and more parameters. In: The Thirteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=6VhDQP7WGX

[20] Li, Y., Mao, H., Girshick, R.B., He, K.: Exploring plain vision transformer backbones for object detection. ArXiv **abs/2203.16527** (2022)

[21] Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W.X., Wen, J.R.: Evaluating object hallucination in large vision-language models. In: Proceedings of the 2023 conference on empirical methods in natural language processing. pp. 292–305 (2023)

[22] Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. Advances in neural information processing systems **36**, 34892–34916 (2023)

[23] Liu, X., Zhou, L., Zhou, Z., Chen, J., He, Z.: Mambavlt: Time-evolving multimodal state space model for vision-language tracking. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 8731–8741 (2025)

[24] Liu, Y., Tian, Y., Zhao, Y., Yu, H., Xie, L., Wang, Y., Ye, Q., Jiao, J., Liu, Y.: Vmamba: Visual state space model. Advances in neural information processing systems **37**, 103031–103063 (2024)

[25] Niu, J., Zheng, Y., Miao, Z., Dong, H., Ge, C., Liang, H., Lu, M., Zeng, B., Zheng, Q., He, C., et al.: Native visual understanding: Resolving resolution dilemmas in vision-language models. arXiv preprint arXiv:2506.12776 (2025)

[26] Qiao, Y., Yu, Z., Guo, L., Chen, S., Zhao, Z., Sun, M., Wu, Q., Liu, J.: Vl-mamba: Exploring state space models for multimodal learning. arXiv preprint arXiv:2403.13600 (2024)

[27] Ranasinghe, K., Shukla, S.N., Poursaeed, O., Ryoo, M.S., Lin, T.Y.: Learning to localize objects improves spatial reasoning in visual-llms. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12977–12987 (2024)

[28] Shabtay, N., Zimerman, I., Schwartz, E., Giryes, R.: Climp: Contrastive language-image mamba pretraining. arXiv preprint arXiv:2601.06891 (2026)

[29] Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., Rohrbach, M.: Towards vqa models that can read. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8317–8326 (2019)

[30] Tong, P., Brown, E., Wu, P., Woo, S., Iyer, A.J.V., Akula, S.C., Yang, S., Yang, J., Middepogu, M., Wang, Z., et al.: Cambrian-1: A fully open, vision-centric exploration of multimodal llms. Advances in Neural Information Processing Systems **37**, 87310–87356 (2024)

[31] Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. arxiv 2020. arXiv preprint arXiv:2012.12877 **2**(3) (2020)

[32] Tu, Z., Talebi, H., Zhang, H., Yang, F., Milanfar, P., Bovik, A., Li, Y.: Maxvit: Multi-axis vision transformer. In: European conference on computer vision. pp. 459–479. Springer (2022)

[33] Wang, H., Fan, J., Wang, Y., Song, K., Wang, T., ZHANG, Z.X.: Droppos: Pre-training vision transformers by reconstructing dropped positions. Advances in Neural Information Processing Systems **36**, 46134–46151 (2023)

[34] Wang, K.J., Liu, Y.H., Su, H.T., Wang, J.W., Wang, Y.S., Hsu, W., Chen, W.C.: Ocid-ref: A 3d robotic dataset with embodied language for clutter scene grounding. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 5333–5338 (2021)

[35] Wang, W., Gao, Z., Gu, L., Pu, H., Cui, L., Wei, X., Liu, Z., Jing, L., Ye, S., Shao, J., et al.: Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265 (2025)

[36] Xiao, C., Li, M., Zhang, Z., Meng, D., Zhang, L.: Spatial-mamba: Effective visual state space models via structure-aware state fusion. arXiv preprint arXiv:2410.15091 (2024)

[37] Yin, S., Fu, C., Zhao, S., Li, K., Sun, X., Xu, T., Chen, E.: A survey on multimodal large language models. National Science Review **11**(12), nwae403 (2024)

[38] Zhang, J., Nguyen, A.T., Han, X., Trinh, V.Q.H., Qin, H., Samaras, D., Hosseini, M.S.: 2dmamba: Efficient state space model for image representation with applications on giga-pixel whole slide image classification. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 3583–3592 (2025)

[39] Zhao, H., Zhang, M., Zhao, W., Ding, P., Huang, S., Wang, D.: Cobra: Extending mamba to multi-modal large language model for efficient inference. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 10421–10429 (2025)

[40] Zhu, L., Liao, B., Zhang, Q., Wang, X., Liu, W., Wang, X.: Vision mamba: Efficient visual representation learning with bidirectional state space model. In: Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., Berkenkamp, F. (eds.) Proceedings of the 41st International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 235, pp. 62429–62442. PMLR (21–27 Jul 2024), https://proceedings.mlr.press/v235/zhu24f.html

## A. Training Setup and Hyperparameters

*Shared training recipe.* Unless otherwise specified, all models are trained under the same one-stage instruction-tuning recipe to ensure a controlled comparison across vision backbones. Following Sec. 2, we freeze the vision encoder and train only the language model and connector.

*Optimization.* We use Fully Sharded Data Parallel (FSDP) training on 4 GPUs with BF16 mixed precision and activation checkpointing for the LLM. Training is run for 1 epoch with global batch size 128 and per-GPU batch size 16, corresponding to 2 gradient-accumulation steps. We use AdamW with learning rate $2 \times 10^{-5}$, weight decay 0.1, max gradient norm 1.0, and a linear-warmup cosine-decay schedule with warmup ratio 0.03. These hyperparameters are fixed across all experiments for fairness.

*Data and preprocessing.* We fine-tune on the LLaVA-v1.5 665K instruction-tuning mixture described in Sec. 2.3. Images are processed with letterbox resizing, which preserves aspect ratio by resizing and padding to the target resolution. Incomplete batches are retained rather than dropped. For variable-length inputs, text is truncated to the tokenizer or model maximum length, and images are padded within each batch as needed.

*Implementation note.* Our implementation is based on a verified training codebase [†] and uses a fixed batch order across experiments. Additional low-level implementation details are provided in the released code. *The source code will be released upon acceptance.*

## B. Vision Encoder Feature Extraction Stage

*Hierarchical backbones.* VMamba, MaxViT, and MambaVision use a Swin-style four-stage hierarchical architecture. For an input image $x_{\text{img}} \in \mathbb{R}^{H\times W\times 3}$, the first stage has spatial resolution $(H/4) \times (W/4)$, and each subsequent stage downsamples the feature map by a factor of 2 along each spatial dimension. Thus, stage $n \in \{1, 2, 3, 4\}$ has spatial size

$$H_n \times W_n = \left(\frac{H}{2^{n+1}}\right) \times \left(\frac{W}{2^{n+1}}\right),$$

with token count

$$L_n = H_n W_n = \frac{HW}{2^{2(n+1)}}.$$

Accordingly, the four stages contain

$$L_1 = \frac{HW}{16}, \qquad L_2 = \frac{HW}{64}, \qquad L_3 = \frac{HW}{256}, \qquad L_4 = \frac{HW}{1024}$$

tokens.

*Correspondence to ViT features.* For comparison, a ViT with patch size $P$ produces

$$L_{\text{ViT}} = \left(\frac{H}{P}\right)\left(\frac{W}{P}\right) = \frac{HW}{P^2}$$

tokens. For the standard ViT/16 setting ($P = 16$), this gives

$$L_{\text{ViT}} = \frac{HW}{256} = L_3.$$

Therefore, stage-3 features of these hierarchical backbones have the same token count as ViT/16 features at the same input resolution. For example, at $224 \times 224$ resolution, stage 3 has spatial size $14 \times 14$ and hence 196 tokens, exactly matching ViT/16.

Table 7 | MaxViT feature-stage ablation on VQA benchmarks. Rows marked (S4) use stage-4 features, while unmarked rows use stage-3 features. Stage 3 matches the ViT/16 token count at the same input resolution, whereas stage 4 uses fewer vision tokens.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | MaxViT-T(S4) | 31M | 224x224 | 49 | 58.36 | 48.75 | 47.29 | 44.03 | 12.23 | 72.87 | 41.22 | 54.89 |
| IN1K | MaxViT-S(S4) | 69M | 224x224 | 49 | 59.05 | 49.68 | 41.13 | 44.45 | 12.16 | 75.25 | 49.23 | 56.50 |
| IN1K | MaxViT-B(S4) | 119M | 224x224 | 49 | 58.61 | 49.33 | 44.53 | 43.63 | 11.84 | 76.33 | 45.19 | 55.68 |
| IN1K | MaxViT-L(S4) | 212M | 224x224 | 49 | 57.66 | 48.86 | 45.95 | 43.91 | 11.93 | 75.34 | 47.76 | 55.29 |
| IN1K | MaxViT-T | 31M | 224x224 | 196 | **61.08** | 49.43 | **47.58** | 44.31 | 12.02 | **76.77** | **53.85** | **58.75** (+3.86) |
| IN1K | MaxViT-S | 69M | 224x224 | 196 | 59.90 | 48.94 | 41.50 | 44.51 | 11.82 | 76.29 | 51.42 | 57.42 (+0.92) |
| IN1K | MaxViT-B | 119M | 224x224 | 196 | 60.10 | **49.72** | 45.00 | **44.59** | **12.29** | 76.26 | 50.94 | 57.60 (+1.92) |
| IN1K | MaxViT-L | 212M | 224x224 | 196 | 60.07 | 49.55 | 46.42 | 43.67 | 12.23 | 76.46 | 48.86 | 57.30 (+2.01) |

Table 8 | MaxViT feature-stage ablation on localization benchmarks. Rows marked (S4) use stage-4 features, while unmarked rows use stage-3 features. Using the lower-token stage-4 features substantially degrades grounding and localization performance.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | MaxViT-T(S4) | 31M | 224x224 | 49 | 13.80 | 9.33 | 10.68 | 4.89 | 8.74 | 48.68 |
| IN1K | MaxViT-S(S4) | 69M | 224x224 | 49 | 12.63 | 8.60 | 10.07 | 4.27 | 7.96 | 49.97 |
| IN1K | MaxViT-B(S4) | 119M | 224x224 | 49 | 12.05 | 8.10 | 9.13 | 4.73 | 7.79 | 49.24 |
| IN1K | MaxViT-L(S4) | 212M | 224x224 | 49 | 12.26 | 7.98 | 8.82 | 4.11 | 7.52 | 48.87 |
| IN1K | MaxViT-T | 31M | 224x224 | 196 | **29.44** | **17.29** | **22.10** | **5.17** | **15.79** (+7.05) | **52.98** (+4.30) |
| IN1K | MaxViT-S | 69M | 224x224 | 196 | 25.57 | 14.41 | 17.28 | 4.38 | 13.32 (+5.36) | 51.49 (+1.52) |
| IN1K | MaxViT-B | 119M | 224x224 | 196 | 22.02 | 12.67 | 15.28 | 3.36 | 11.41 (+3.62) | 51.39 (+2.15) |
| IN1K | MaxViT-L | 212M | 224x224 | 196 | 21.15 | 12.01 | 14.77 | 2.94 | 10.81 (+3.29) | 51.05 (+2.18) |

*Stage choice in our experiments.* We also report results using stage-4 features for MaxViT and MambaVision. Across Tables 7, 8, 9, and 10, we observe that the choice of feature stage has a substantial impact on downstream performance, particularly on localization benchmarks. Notably, stage-4 features often underperform stage-3 features on grounding and localization tasks even when the stage-4 variant uses a higher input resolution and a comparable number of vision tokens, indicating that this effect is not explained by token count alone. A plausible explanation is that stage-4 features are more semantically abstract, whereas stage-3 features retain richer spatial detail. We therefore adopt stage 3 as the main setting, as it both matches the token count of ViT/16 and yields a better balance between VQA and localization performance.

## C. VMamba vs Vim

As shown in Table 11 and Table 12, VMamba consistently performs better than Vim in our VLM setting. We therefore adopt VMamba as the representative SSM-based vision backbone in the main paper.

We restrict this comparison to Vim and VMamba because they are among the most established SSM-based vision backbones with mature public implementations. Exploring newer variants built on top of these architectures in the VLM setting is an interesting direction for future work.

[†]https://github.com/TRI-ML/prismatic-vlms

Table 9 | MambaVision feature-stage ablation on VQA benchmarks. Rows marked (S4) use stage-4 features, while unmarked rows use stage-3 features. We report both settings to study the trade-off between token count and semantic abstraction.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | MambaVision-T(S4) | 32M | 224x224 | 49 | 60.84 | 51.77 | 45.76 | 43.86 | 12.06 | 79.70 | **53.51** | 58.68 |
| IN1K | MambaVision-T2(S4) | 35M | 224x224 | 49 | 60.85 | 51.26 | 44.02 | 44.11 | 12.54 | 79.18 | 49.56 | 58.11 |
| IN1K | MambaVision-S(S4) | 50M | 224x224 | 49 | 61.34 | 51.32 | 42.71 | **44.52** | 12.35 | 79.77 | 53.49 | 59.00 |
| IN1K | MambaVision-B(S4) | 98M | 224x224 | 49 | **61.87** | 52.16 | 45.17 | 43.43 | **12.63** | **80.47** | 52.45 | **59.34** |
| IN1K | MambaVision-L(S4) | 228M | 224x224 | 49 | 61.52 | **52.35** | 42.86 | 44.16 | 12.46 | 80.32 | 50.53 | 58.80 |
| IN1K | MambaVision-L2(S4) | 242M | 224x224 | 49 | 61.83 | 51.94 | 44.62 | 43.94 | 11.96 | 80.01 | 52.34 | 59.26 |
| IN1K | MambaVision-T | 32M | 224x224 | 196 | 56.85 | 47.43 | 46.16 | 43.74 | 11.14 | 71.34 | 46.89 | 54.38 (-4.30) |
| IN1K | MambaVision-T2 | 35M | 224x224 | 196 | 56.69 | 47.14 | 43.17 | 43.31 | 10.82 | 69.58 | 43.74 | 53.71 (-4.40) |
| IN1K | MambaVision-S | 50M | 224x224 | 196 | 58.10 | 48.05 | 40.91 | 43.08 | 11.36 | 72.03 | 49.48 | 55.61 (-3.39) |
| IN1K | MambaVision-B | 98M | 224x224 | 196 | 59.14 | 49.23 | 44.21 | 43.33 | 11.43 | 73.05 | 49.53 | 56.53 (-2.81) |
| IN1K | MambaVision-L | 228M | 224x224 | 196 | 59.26 | 49.56 | 42.13 | 43.75 | 11.41 | 74.46 | 51.70 | 56.94 (-1.86) |
| IN1K | MambaVision-L2 | 242M | 224x224 | 196 | 59.27 | 48.83 | **48.91** | 43.30 | 11.16 | 74.15 | 50.70 | 56.86 (-2.40) |

Table 10 | MambaVision feature-stage ablation on localization benchmarks. Rows marked (S4) use stage-4 features, while unmarked rows use stage-3 features. Stage 3 provides a more balanced trade-off for localization under a fair token-count comparison to ViT/16-style backbones.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | MambaVision-T(S4) | 32M | 224x224 | 49 | 33.88 | 23.69 | 28.72 | 7.40 | 20.04 | 53.49 |
| IN1K | MambaVision-T2(S4) | 35M | 224x224 | 49 | 30.11 | 20.31 | 23.84 | 7.20 | 17.70 | 52.68 |
| IN1K | MambaVision-S(S4) | 50M | 224x224 | 49 | 32.87 | 21.65 | 26.06 | 5.95 | 18.42 | 53.55 |
| IN1K | MambaVision-B(S4) | 98M | 224x224 | 49 | 36.33 | 25.48 | 31.58 | 9.01 | 22.03 | **54.33** |
| IN1K | MambaVision-L(S4) | 228M | 224x224 | 49 | 35.34 | 24.22 | 30.37 | 9.35 | 21.50 | 53.79 |
| IN1K | MambaVision-L2(S4) | 242M | 224x224 | 49 | 31.21 | 20.96 | 25.39 | 7.26 | 18.32 | 53.76 |
| IN1K | MambaVision-T | 32M | 224x224 | 196 | 34.59 | 24.01 | 27.14 | 9.52 | 20.98 (+0.94) | 49.89 (-3.60) |
| IN1K | MambaVision-T2 | 35M | 224x224 | 196 | 35.78 | 24.69 | 27.61 | 9.72 | 21.56 (+3.86) | 49.39 (-3.29) |
| IN1K | MambaVision-S | 50M | 224x224 | 196 | 44.65 | 33.06 | 37.48 | 13.21 | 28.22 (+9.80) | 51.93 (-1.62) |
| IN1K | MambaVision-B | 98M | 224x224 | 196 | **48.52** | 35.57 | **40.58** | 15.84 | 31.17 (+9.14) | 53.12 (-1.21) |
| IN1K | MambaVision-L | 228M | 224x224 | 196 | 48.19 | **35.82** | 40.26 | **16.79** | **31.51** (+10.01) | 53.52 (-0.27) |
| IN1K | MambaVision-L2 | 242M | 224x224 | 196 | 47.91 | 35.41 | 40.28 | 16.48 | 31.22 (+12.90) | 53.42 (-0.34) |

## D. Benchmark Correlation Martix

Fig. 3 shows the correlation matrix of benchmark scores across all evaluated models in this study. Each entry represents the Pearson correlation computed using the performance of the same set of models on the corresponding pair of benchmarks. The matrix provides a compact view of how different benchmarks relate to each other across models.

## E. Linear Probing Results

We evaluate the representation quality of the vision backbones using standard linear probing. For each backbone, we first extract and cache global image features from the frozen encoder for the train, validation, and test splits of each dataset. For each image, we extract a single global feature from the frozen encoder: we use the CLS token for backbones that provide one, and otherwise apply mean pooling over spatial tokens. A linear classifier (single fully connected layer) is then trained on top of the frozen features. We train the probe for 30 epochs using AdamW with learning rate $10^{-3}$ and batch size 512. Model selection is performed using the validation split, with weight decay selected from a

Table 11 | Vim vs. VMamba on VQA benchmarks under the shared VLM training recipe. VMamba achieves stronger overall performance and is therefore used as the representative SSM-based backbone in the main paper.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | Vim-T | 7M | 224x224 | 196 | 55.83 | 46.93 | 39.39 | 43.82 | 11.67 | 72.91 | 45.69 | 53.40 |
| IN1K | Vim-T-ft | 7M | 224x224 | 729 | 56.82 | 47.62 | 41.51 | 43.56 | 11.76 | 74.69 | 47.73 | 54.52 |
| IN1K | Vim-S | 26M | 224x224 | 196 | 56.84 | 47.73 | 40.47 | 43.81 | 11.58 | 73.58 | 49.58 | 54.74 |
| IN1K | Vim-S-ft | 26M | 224x224 | 729 | 57.68 | 48.55 | 45.68 | 43.78 | 11.89 | 74.31 | 45.96 | 55.02 |
| IN1K | Vim-B | 98M | 224x224 | 196 | 62.93 | 52.11 | 45.52 | **44.81** | **12.50** | 78.97 | 55.07 | 60.46 |
| IN1K | VMamba-T | 30M | 224x224 | 196 | 64.99 | 54.02 | 44.96 | 44.62 | 12.22 | 81.70 | 54.58 | 62.07 |
| IN1K | VMamba-S | 50M | 224x224 | 196 | **65.24** | **54.08** | 44.95 | 44.06 | 12.24 | **82.20** | **55.54** | **62.39** |
| IN1K | VMamba-B | 80M | 224x224 | 196 | 64.20 | 53.25 | **47.97** | 43.81 | 12.08 | 80.75 | 54.05 | 61.38 |

Table 12 | Vim vs. VMamba on localization benchmarks under the shared VLM training recipe. VMamba yields stronger grounding performance overall and is therefore used as the representative SSM-based backbone in the main paper.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | Vim-T | 7M | 224x224 | 196 | 21.35 | 12.42 | 14.99 | 3.51 | 11.21 | 47.73 |
| IN1K | Vim-T-ft | 7M | 224x224 | 729 | 21.28 | 12.30 | 15.13 | 3.96 | 11.37 | 48.72 |
| IN1K | Vim-S | 26M | 224x224 | 196 | 22.33 | 12.60 | 15.32 | 2.70 | 11.20 | 48.88 |
| IN1K | Vim-S-ft | 26M | 224x224 | 729 | 22.99 | 13.09 | 15.01 | 3.57 | 11.80 | 49.21 |
| IN1K | Vim-B | 98M | 224x224 | 196 | 44.62 | 34.10 | 38.54 | 13.17 | 28.56 | 56.17 |
| IN1K | VMamba-T | 30M | 224x224 | 196 | **58.25** | **46.64** | **51.74** | 20.24 | **39.20** | 59.00 |
| IN1K | VMamba-S | 50M | 224x224 | 196 | 56.48 | 44.27 | 49.88 | **23.09** | 39.17 | **59.27** |
| IN1K | VMamba-B | 80M | 224x224 | 196 | 42.06 | 31.43 | 36.15 | 15.23 | 27.89 | 56.88 |

grid $\{10^{-6}, \ldots, 10^{-1}\}$. The best model according to the validation metric is evaluated on the test split. For multiclass datasets, we report top-1 accuracy, while multilabel datasets use mean average precision (mAP). Table 13 and Table 14 report the resulting test performance. The *Weighted Avg* column is computed as the dataset-size-weighted average across all available benchmarks.

*Observations.* Tables 13 and 14 show that linear probing only partially explains downstream VLM behavior. In some families, larger supervised models exhibit slightly worse probing performance, such as MaxViT-L relative to MaxViT-B, MambaVision-L2 relative to MambaVision-L, and VMamba-B relative to VMamba-S, which is consistent with reduced transferability at scale. However, higher probing performance still does not reliably predict stronger VLM performance: ViT-B probes better than ViT-S yet performs worse overall, and MaxViT-S/B/L all probe substantially better than MaxViT-T while yielding weaker localization and lower overall VLM performance. This supports the interpretation in Sec. 4.4 that the core issue is not merely overfitting to ImageNet-1K, but over-specialization to classification objectives, which favors global category information over spatially grounded representations. Conversely, dense-objective checkpoints can improve downstream VLM performance despite only modest, or even lower, probing scores relative to their classification-pretrained counterparts, indicating that standard global-feature probing does not fully capture the spatial fidelity that matters for grounded VLM behavior.

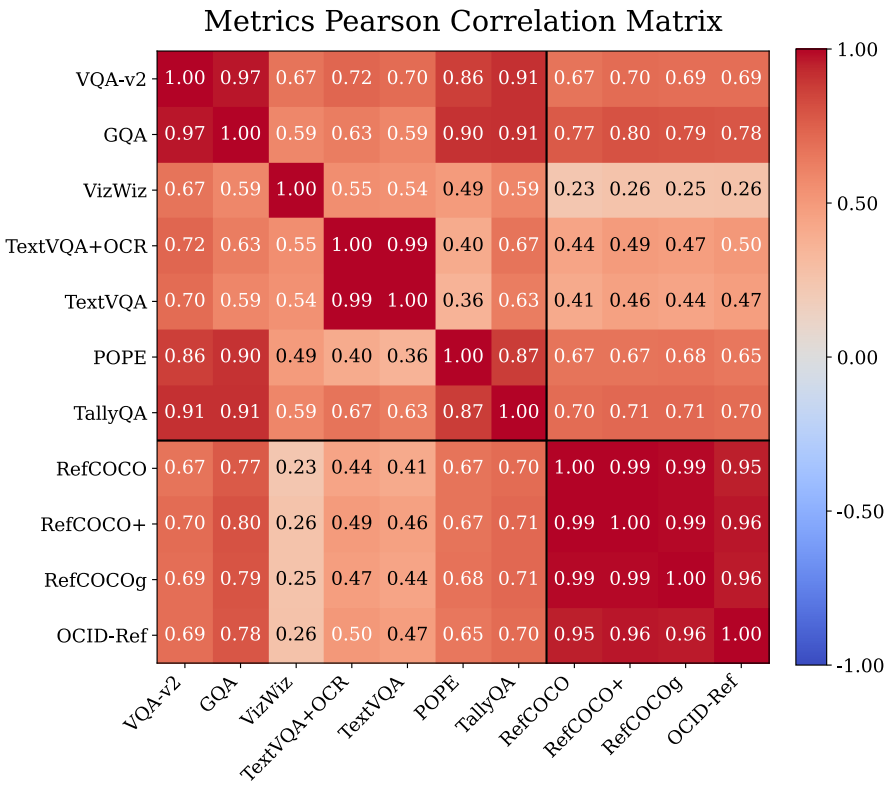


Figure 3 | Correlation matrix of benchmark scores across all evaluated models.

Table 13 | Weighted linear probing scores versus downstream weighted VQA, localization, and overall performance on the test split (%). The Weighted Probing score is the dataset-size-weighted average across probing benchmarks.

| Pretrained Dataset | Visual Encoder | Encoder Size | Resolution | Weighted Probing | Weighted VQA | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|
| IN1K | ViT-S | 22M | 224 | 39.19 | 57.25 | 17.82 | 51.95 |
| IN1K | ViT-B | 87M | 224 | 49.88 | 57.17 | 13.92 | 51.36 |
| IN1K | MaxViT-T | 31M | 224 | 69.22 | 58.75 | 15.79 | 52.98 |
| IN1K | MaxViT-S | 69M | 224 | 76.23 | 57.42 | 13.32 | 51.49 |
| IN1K | MaxViT-B | 119M | 224 | 78.13 | 57.60 | 11.41 | 51.39 |
| IN1K | MaxViT-L | 212M | 224 | 75.92 | 57.30 | 10.81 | 51.05 |
| IN1K | MambaVision-T | 32M | 224 | 37.03 | 54.38 | 20.98 | 49.89 |
| IN1K | MambaVision-T2 | 35M | 224 | 36.94 | 53.71 | 21.56 | 49.39 |
| IN1K | MambaVision-S | 50M | 224 | 41.74 | 55.61 | 28.22 | 51.93 |
| IN1K | MambaVision-B | 98M | 224 | 43.20 | 56.53 | 31.17 | 53.12 |
| IN1K | MambaVision-L | 228M | 224 | 49.31 | 56.94 | 31.51 | 53.52 |
| IN1K | MambaVision-L2 | 242M | 224 | 48.32 | 56.86 | 31.22 | 53.42 |
| IN1K | VMamba-T | 30M | 224 | 73.73 | 62.07 | 39.20 | 59.00 |
| IN1K | VMamba-S | 50M | 224 | **79.45** | 62.39 | 39.17 | 59.27 |
| IN1K | VMamba-B | 80M | 224 | <u>78.88</u> | 61.38 | 27.89 | 56.88 |
| IN1K → COCO | ViTDet-B | 111M | 1024 | 52.78 | <u>63.00</u> | 43.74 | 60.42 |
| IN1K → COCO | ViTDet-L | 331M | 1024 | 56.23 | 57.64 | 13.05 | 51.65 |
| IN1K → COCO | ViTDet-H | 662M | 1024 | 54.54 | 61.89 | 35.38 | 58.33 |
| IN1K → COCO | VMamba-T | 30M | 800 | 50.21 | 58.05 | 14.86 | 52.25 |
| IN1K → COCO | VMamba-S | 50M | 800 | 56.54 | 62.78 | **47.94** | **60.78** |
| IN1K → COCO | VMamba-B | 89M | 800 | 58.24 | 58.57 | 15.02 | 52.72 |
| IN1K → ADE20K | DeiT-S | 58M | 512 | 54.49 | 59.36 | 31.78 | 55.65 |
| IN1K → ADE20K | DeiT-B | 134M | 512 | 67.08 | 60.69 | 33.77 | 57.07 |
| IN1K → ADE20K | VMamba-T | 30M | 512 | 58.15 | 62.60 | 43.47 | 60.03 |
| IN1K → ADE20K | VMamba-S | 50M | 512 | 62.35 | **63.21** | 44.98 | <u>60.76</u> |
| IN1K → ADE20K | VMamba-B | 89M | 512 | 65.32 | 62.87 | <u>45.08</u> | 60.48 |

Table 14 | Per-dataset linear probing results on the test split (%). Results are obtained by training a linear classifier on frozen global image features.

| Visual Encoder | food101 | cifar10 | cifar100 | dtd | oxford pets | stanford cars | fgvc aircraft | sun397 | oxford flowers102 | caltech101 | birdsnap | voc2007 | coco multilabel |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ViT-S | 23.56 | 78.83 | 51.55 | 53.88 | 25.02 | 7.20 | 8.94 | 45.15 | 28.90 | 71.10 | 37.09 | 36.69 | 42.60 |
| ViT-B | 39.58 | 87.88 | 62.13 | 54.31 | 60.53 | 9.75 | 16.26 | 49.43 | 30.54 | 77.61 | 68.83 | 54.92 | 50.87 |
| MaxViT-T | 71.64 | 92.23 | 76.40 | 75.11 | 74.76 | 30.53 | 28.86 | 68.69 | 77.69 | 89.99 | 59.35 | 61.57 | 61.38 |
| MaxViT-S | 76.91 | 94.82 | 80.56 | **77.87** | 86.64 | 46.61 | 37.77 | 71.43 | **86.27** | 92.62 | 88.61 | 78.69 | 68.13 |
| MaxViT-B | 77.71 | **96.51** | 82.21 | 73.99 | 92.31 | 54.46 | 36.51 | 70.91 | 84.75 | 92.95 | 98.14 | 86.11 | 73.63 |
| MaxViT-L | 75.72 | 95.87 | 80.82 | 69.41 | 92.29 | 49.94 | 34.92 | 68.92 | 73.80 | 90.65 | **98.90** | 84.98 | 74.51 |
| MambaVision-T | 38.08 | 63.25 | 35.93 | 52.82 | 18.34 | 8.32 | 7.68 | 41.41 | 43.32 | 62.82 | 11.42 | 21.74 | 32.27 |
| MambaVision-T2 | 37.43 | 62.94 | 35.96 | 52.87 | 18.70 | 8.63 | 8.22 | 41.14 | 43.84 | 63.17 | 11.39 | 22.58 | 32.36 |
| MambaVision-S | 43.30 | 67.77 | 42.14 | 59.41 | 22.19 | 10.15 | 9.78 | 46.59 | 48.98 | 67.44 | 15.26 | 25.69 | 36.56 |
| MambaVision-B | 45.58 | 69.28 | 43.97 | 59.57 | 22.27 | 10.48 | 9.15 | 48.28 | 51.72 | 67.97 | 16.34 | 25.71 | 37.90 |
| MambaVision-L | 51.68 | 73.85 | 50.26 | 65.00 | 31.43 | 17.07 | 13.23 | 53.45 | 60.11 | 74.52 | 23.34 | 35.16 | 42.22 |
| MambaVision-L2 | 51.11 | 72.46 | 48.57 | 64.26 | 29.11 | 16.08 | 12.96 | 52.77 | 58.56 | 73.44 | 22.11 | 33.73 | 41.75 |
| VMamba-T | 76.09 | 94.29 | 80.12 | 76.01 | 88.14 | 32.22 | 30.18 | 71.85 | 81.12 | 92.08 | 79.10 | 70.26 | 68.11 |
| VMamba-S | **80.38** | 96.13 | **83.65** | 75.96 | 93.51 | **55.60** | **39.00** | **74.63** | 85.54 | **93.11** | 89.79 | 83.84 | 73.56 |
| VMamba-B | 80.13 | 95.96 | 82.25 | 75.27 | **94.06** | 53.51 | 32.55 | 74.14 | 80.94 | 93.10 | 90.54 | **88.04** | 76.01 |
| ViTDet-B | 59.25 | 66.14 | 27.26 | 59.89 | 36.17 | 10.01 | 18.12 | 59.07 | 53.37 | 76.92 | 43.64 | 74.81 | 74.15 |
| ViTDet-L | 66.78 | 69.22 | 26.38 | 65.16 | 38.51 | 10.73 | 20.82 | 61.37 | 54.37 | 82.08 | 43.66 | 79.46 | 78.86 |
| ViTDet-H | 66.54 | 74.12 | 29.20 | 62.98 | 26.55 | 10.10 | 16.44 | 57.83 | 44.59 | 77.70 | 35.88 | 78.54 | 80.89 |
| VMamba-T | 59.94 | 53.57 | 24.11 | 58.78 | 39.96 | 10.51 | 18.72 | 55.07 | 60.90 | 76.33 | 41.96 | 57.58 | 69.09 |
| VMamba-S | 69.50 | 52.74 | 22.55 | 60.80 | 59.63 | 14.14 | 21.18 | 62.10 | 63.12 | 81.67 | 47.18 | 76.01 | 80.10 |
| VMamba-B | 69.60 | 57.08 | 28.14 | 61.22 | 56.39 | 13.07 | 22.32 | 63.30 | 66.01 | 82.07 | 52.07 | 79.43 | **83.84** |
| DeiT-S | 63.56 | 50.26 | 22.22 | 65.21 | 63.89 | 13.69 | 14.64 | 68.33 | 56.07 | 79.83 | 56.51 | 61.49 | 61.56 |
| DeiT-B | 76.31 | 61.22 | 34.74 | 71.54 | 88.44 | 29.81 | 22.32 | 73.68 | 76.47 | 88.18 | 89.06 | 80.97 | 72.03 |
| VMamba-T | 67.47 | 60.79 | 30.58 | 69.79 | 70.37 | 13.92 | 14.55 | 67.62 | 66.95 | 77.71 | 63.01 | 61.17 | 65.16 |
| VMamba-S | 75.26 | 57.21 | 24.41 | 71.28 | 85.39 | 20.51 | 16.20 | 70.46 | 68.89 | 84.22 | 79.90 | 68.13 | 71.67 |
| VMamba-B | 76.40 | 63.21 | 31.82 | 72.13 | 88.50 | 21.07 | 19.98 | 72.72 | 68.55 | 86.59 | 82.53 | 76.90 | 75.34 |

# F. Unfair Comparisons & Contrastive/Self-Supervised Backbones

## F.1. Unfair Comparisons

We additionally report several comparisons that are informative but fall outside the controlled setting of the main paper. In these experiments, multiple factors change at once, including the pretraining dataset, input resolution, number of visual tokens, and model size. As a result, these tables should be interpreted as supplementary reference rather than evidence for the main controlled conclusions.

*Different classification pretraining settings.* Tables 15, 16, 17, and 18 report classification-pretrained checkpoints that differ from the available VMamba setting in pretraining data scale and/or evaluation resolution. ViT and several MaxViT variants generally benefit on VQA from larger-scale classification pretraining or higher-resolution feature extraction, but these changes do not consistently improve localization. MambaVision does not show the same trend: increasing pretraining scale or resolution does not reliably improve either VQA or localization, and can even degrade performance relative to the smaller controlled setting. In contrast, VMamba remains competitive on VQA and continues to dominate localization despite using a much smaller classification pretraining dataset, lower input resolution, and smaller model sizes. This further supports the summary in Sec. 4 that the inductive bias is important when the pretraining objective is not encouraging the backbone to preserve the spatial information.

Table 15 | Representative VQA comparison at matched token count. All models are evaluated in the standard 224 × 224 / 196-token setting, or the stage-3 equivalent for hierarchical backbones, providing a compact subset of the main comparison.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN21k | ViT-T | 5.7M | 224x224 | 196 | 62.82 | 51.30 | 46.03 | 44.62 | 11.90 | 76.94 | 52.65 | 59.95 |
| IN21k | ViT-S | 22M | 224x224 | 196 | 61.28 | 50.34 | 47.95 | 44.18 | 12.30 | 78.59 | 53.62 | 58.97 |
| IN21k | ViT-B | 87M | 224x224 | 196 | **68.16** | **55.96** | 49.40 | **44.75** | **12.77** | 82.34 | 55.26 | **64.70** |
| IN21k | ViT-L | 304M | 224x224 | 196 | 67.13 | 54.85 | 49.15 | 44.28 | 12.64 | **82.38** | 53.32 | 63.61 |
| IN21k → IN1K | ViT-T | 5.7M | 224x224 | 196 | 61.34 | 50.00 | 45.30 | 44.02 | 11.84 | 77.90 | 53.82 | 58.96 |
| IN21k → IN1K | ViT-S | 22M | 224x224 | 196 | 60.39 | 49.46 | 44.54 | 44.58 | 12.23 | 78.49 | 51.60 | 57.95 |
| IN21k → IN1K | ViT-B | 87M | 224x224 | 196 | 67.24 | 54.96 | 47.76 | 43.92 | 12.34 | 82.16 | **57.16** | 64.17 |
| IN21k → IN1K | ViT-L | 304M | 224x224 | 196 | 62.17 | 50.82 | 42.10 | 44.40 | 12.71 | 78.42 | 53.47 | 59.55 |
| IN21K | MaxViT-B | 119M | 224x224 | 196 | 62.67 | 49.98 | 45.92 | 44.08 | 12.38 | 76.46 | 53.58 | 59.88 |
| IN21K | MaxViT-L | 212M | 224x224 | 196 | 65.50 | 52.03 | 47.31 | 44.30 | 12.49 | 80.59 | 53.62 | 62.23 |
| IN21K | MaxViT-XL | 475M | 224x224 | 196 | 65.55 | 51.61 | **50.14** | 44.13 | 12.58 | 80.07 | 54.78 | 62.43 |
| IN21K | MambaVision-B | 98M | 224x224 | 196 | 49.42 | 42.07 | 41.98 | 42.71 | 10.36 | 58.90 | 42.05 | 47.50 |
| IN21K | MambaVision-L | 228M | 224x224 | 196 | 52.06 | 43.27 | 39.10 | 43.12 | 10.64 | 65.22 | 43.49 | 49.87 |
| IN1K | VMamba-T | 30M | 224x224 | 196 | 64.99 | 54.02 | 44.96 | 44.62 | 12.22 | 81.70 | 54.58 | 62.07 |
| IN1K | VMamba-S | 50M | 224x224 | 196 | 65.24 | 54.08 | 44.95 | 44.06 | 12.24 | 82.20 | 55.54 | 62.39 |
| IN1K | VMamba-B | 80M | 224x224 | 196 | 64.20 | 53.25 | 47.97 | 43.81 | 12.08 | 80.75 | 54.05 | 61.38 |

Table 16 | Representative localization comparison at matched token count. All models are evaluated in the standard 224 × 224 / 196-token setting, or the stage-3 equivalent for hierarchical backbones, providing a compact subset of the main comparison.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN21k | ViT-T | 5.7M | 224x224 | 196 | 34.52 | 21.72 | 26.47 | 7.30 | 19.43 | 54.50 |
| IN21k | ViT-S | 22M | 224x224 | 196 | 24.40 | 13.79 | 17.42 | 4.73 | 13.04 | 52.80 |
| IN21k | ViT-B | 87M | 224x224 | 196 | 55.00 | 44.79 | 48.35 | 19.90 | 37.46 | **61.04** |
| IN21k | ViT-L | 304M | 224x224 | 196 | 40.64 | 31.38 | 35.21 | 11.40 | 25.86 | 58.54 |
| IN21k → IN1K | ViT-T | 5.7M | 224x224 | 196 | 31.56 | 19.84 | 23.79 | 8.35 | 18.40 | 53.51 |
| IN21k → IN1K | ViT-S | 22M | 224x224 | 196 | 24.29 | 14.05 | 16.99 | 4.05 | 12.75 | 51.88 |
| IN21k → IN1K | ViT-B | 87M | 224x224 | 196 | 54.82 | 45.00 | 48.30 | 18.44 | 36.87 | 60.50 |
| IN21k → IN1K | ViT-L | 304M | 224x224 | 196 | 22.92 | 13.04 | 15.77 | 3.18 | 11.69 | 53.12 |
| IN21K | MaxViT-B | 119M | 224x224 | 196 | 23.04 | 12.72 | 16.44 | 2.56 | 11.46 | 53.38 |
| IN21K | MaxViT-L | 212M | 224x224 | 196 | 26.27 | 15.83 | 19.53 | 4.94 | 14.30 | 55.79 |
| IN21K | MaxViT-XL | 475M | 224x224 | 196 | 24.43 | 13.70 | 18.34 | 4.46 | 13.02 | 55.79 |
| IN21K | MambaVision-B | 98M | 224x224 | 196 | 19.44 | 10.68 | 12.99 | 3.52 | 10.12 | 42.48 |
| IN21K | MambaVision-L | 228M | 224x224 | 196 | 22.52 | 11.91 | 14.69 | 3.91 | 11.50 | 44.72 |
| IN1K | VMamba-T | 30M | 224x224 | 196 | **58.25** | **46.64** | **51.74** | 20.24 | **39.20** | 59.00 |
| IN1K | VMamba-S | 50M | 224x224 | 196 | 56.48 | 44.27 | 49.88 | **23.09** | 39.17 | 59.27 |
| IN1K | VMamba-B | 80M | 224x224 | 196 | 42.06 | 31.43 | 36.15 | 15.23 | 27.89 | 56.88 |

Table 17 | Additional VQA results for selected MaxViT and MambaVision configurations beyond the matched 196-token setting. We report larger-resolution and larger-token variants for hierarchical backbones and compare them with the standard VMamba models.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | MaxViT-T | 31M | 384x384 | 576 | 64.04 | 53.04 | 48.00 | 44.69 | 11.71 | 79.86 | 53.30 | 61.13 |
| IN1K | MaxViT-S | 69M | 384x384 | 576 | 61.31 | 49.85 | 46.73 | 44.19 | 11.79 | 77.57 | 52.05 | 58.70 |
| IN1K | MaxViT-B | 119M | 384x384 | 576 | 63.90 | 52.76 | 46.61 | 43.95 | 12.27 | 81.23 | 52.83 | 60.97 |
| IN1K | MaxViT-L | 212M | 384x384 | 576 | 62.90 | 51.26 | 49.94 | 43.80 | 12.41 | 81.00 | 49.25 | 59.73 |
| IN1K | MaxViT-T | 31M | 512x512 | 1024 | 64.25 | 52.84 | 48.14 | 44.40 | 11.84 | 79.24 | 51.93 | 61.08 |
| IN1K | MaxViT-S | 69M | 512x512 | 1024 | 63.51 | 51.26 | 46.29 | 44.25 | 11.92 | 79.36 | 54.01 | 60.71 |
| IN1K | MaxViT-B | 119M | 512x512 | 1024 | 64.53 | 53.14 | 47.92 | 44.10 | 12.09 | 82.90 | 53.06 | 61.56 |
| IN1K | MaxViT-L | 212M | 512x512 | 1024 | 63.19 | 51.17 | 46.09 | 43.63 | 12.41 | 81.30 | 48.87 | 59.84 |
| IN21K → IN1K | MaxViT-B | 119M | 384x384 | 576 | 63.44 | 50.92 | 50.41 | 44.38 | 12.71 | 76.72 | 54.53 | 60.71 |
| IN21K → IN1K | MaxViT-L | 212M | 384x384 | 576 | 67.15 | 52.82 | **53.89** | 43.72 | **13.28** | 82.30 | 55.45 | **63.89** |
| IN21K → IN1K | MaxViT-XL | 475M | 384x384 | 576 | 66.37 | 52.53 | 49.11 | 44.16 | 12.89 | 81.46 | 53.58 | 62.95 |
| IN21K → IN1K | MaxViT-B | 119M | 512x512 | 1024 | 63.42 | 51.07 | 49.87 | 44.10 | 12.53 | 77.67 | 54.35 | 60.69 |
| IN21K → IN1K | MaxViT-L | 212M | 512x512 | 1024 | 67.27 | 52.70 | 51.91 | **44.71** | 13.09 | 83.11 | 54.38 | 63.84 |
| IN21K → IN1K | MaxViT-XL | 475M | 512x512 | 1024 | **67.34** | 52.72 | 51.08 | 44.16 | 12.95 | **83.15** | 52.83 | 63.67 |
| IN21K | MambaVision-L2 | 242M | 512x512 | 1024 | 50.80 | 41.95 | 41.57 | 42.79 | 10.50 | 62.97 | 41.93 | 48.63 |
| IN21K | MambaVision-L3 | 740M | 256x256 | 256 | 52.18 | 43.58 | 36.27 | 42.94 | 10.66 | 65.13 | 38.51 | 49.26 |
| IN21K | MambaVision-L3 | 740M | 512x512 | 1024 | 51.50 | 42.62 | 40.91 | 43.40 | 10.45 | 61.98 | 43.19 | 49.32 |
| IN1K | VMamba-T | 30M | 224x224 | 196 | 64.99 | 54.02 | 44.96 | 44.62 | 12.22 | 81.70 | 54.58 | 62.07 |
| IN1K | VMamba-S | 50M | 224x224 | 196 | 65.24 | **54.08** | 44.95 | 44.06 | 12.24 | 82.20 | **55.54** | 62.39 |
| IN1K | VMamba-B | 80M | 224x224 | 196 | 64.20 | 53.25 | 47.97 | 43.81 | 12.08 | 80.75 | 54.05 | 61.38 |

Table 18 | Additional localization results for selected MaxViT and MambaVision configurations beyond the matched 196-token setting. We report larger-resolution and larger-token variants for hierarchical backbones and compare them with the standard VMamba models.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | MaxViT-T | 31M | 384x384 | 576 | 55.01 | 43.06 | 47.75 | 22.30 | 37.97 | 58.02 |
| IN1K | MaxViT-S | 69M | 384x384 | 576 | 25.68 | 14.97 | 18.22 | 4.08 | 13.46 | 52.62 |
| IN1K | MaxViT-B | 119M | 384x384 | 576 | 28.18 | 18.45 | 20.61 | 6.08 | 15.98 | 54.92 |
| IN1K | MaxViT-L | 212M | 384x384 | 576 | 23.92 | 13.65 | 17.12 | 5.36 | 13.12 | 53.46 |
| IN1K | MaxViT-T | 31M | 512x512 | 1024 | 54.54 | 43.28 | 48.24 | **23.42** | 38.42 | 58.03 |
| IN1K | MaxViT-S | 69M | 512x512 | 1024 | 35.07 | 23.51 | 28.10 | 9.29 | 20.99 | 55.37 |
| IN1K | MaxViT-B | 119M | 512x512 | 1024 | 33.23 | 22.73 | 25.14 | 8.34 | 19.64 | 55.92 |
| IN1K | MaxViT-L | 212M | 512x512 | 1024 | 25.60 | 15.34 | 18.83 | 5.90 | 14.34 | 53.72 |
| IN21K → IN1K | MaxViT-B | 119M | 384x384 | 576 | 21.22 | 12.01 | 14.56 | 2.87 | 10.77 | 54.00 |
| IN21K → IN1K | MaxViT-L | 212M | 384x384 | 576 | 26.46 | 15.92 | 19.36 | 5.79 | 14.70 | 57.28 |
| IN21K → IN1K | MaxViT-XL | 475M | 384x384 | 576 | 24.67 | 14.46 | 18.32 | 3.67 | 12.93 | 56.23 |
| IN21K → IN1K | MaxViT-B | 119M | 512x512 | 1024 | 21.72 | 12.49 | 15.26 | 2.58 | 10.97 | 54.01 |
| IN21K → IN1K | MaxViT-L | 212M | 512x512 | 1024 | 26.45 | 15.50 | 20.26 | 5.22 | 14.46 | 57.21 |
| IN21K → IN1K | MaxViT-XL | 475M | 512x512 | 1024 | 26.67 | 15.66 | 20.20 | 5.33 | 14.59 | 57.07 |
| IN21K | MambaVision-L2 | 242M | 512x512 | 1024 | 21.80 | 12.46 | 14.17 | 3.45 | 11.22 | 43.60 |
| IN21K | MambaVision-L3 | 740M | 256x256 | 256 | 21.39 | 11.89 | 13.93 | 3.71 | 11.06 | 44.13 |
| IN21K | MambaVision-L3 | 740M | 512x512 | 1024 | 21.27 | 11.61 | 13.46 | 3.66 | 10.89 | 44.15 |
| IN1K | VMamba-T | 30M | 224x224 | 196 | **58.25** | **46.64** | **51.74** | 20.24 | **39.20** | 59.00 |
| IN1K | VMamba-S | 50M | 224x224 | 196 | 56.48 | 44.27 | 49.88 | 23.09 | 39.17 | **59.27** |
| IN1K | VMamba-B | 80M | 224x224 | 196 | 42.06 | 31.43 | 36.15 | 15.23 | 27.89 | 56.88 |

*Dense-objective pretraining beyond the controlled setup.* Tables 19 and 20 report additional segmentation-pretrained backbones. Under these much larger pretraining setups, very large models can achieve the strongest VQA results in this group. Nevertheless, VMamba variants remain consistently stronger on localization. These results reinforce the main paper's observation that dense objectives can improve downstream VLM performance, while also showing that better spatial transfer does not simply come from scaling model size, pretraining data, or resolution.

Table 19 | VQA benchmarks for additional backbones with non-classification pretraining or task-adapted checkpoints, alongside VMamba transfer variants. Because these settings differ from the main controlled comparison in pretraining data and objective, they are reported separately.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Multi-Modal → ADE20K | Perceiver | 364M | 512x512 | 256 | 65.62 | 54.63 | 46.25 | 44.26 | 12.25 | 83.79 | 56.62 | 62.92 |
| IN22K → ADE20K | BeiT-L | 451M | 640x640 | 400 | 68.78 | 56.52 | 48.06 | 44.75 | 13.01 | 86.88 | 58.24 | 65.70 |
| IN22K → ADE20K | BeiT-L | 568M | 640x640 | 400 | 69.57 | 57.12 | **51.95** | 44.03 | 13.14 | 86.57 | 58.64 | 66.41 |
| IN22K+COCO → ADE20K | BeiT-L | 571M | 896x896 | 784 | 69.57 | 57.47 | 49.73 | 44.58 | 12.89 | 88.09 | 59.03 | 66.49 |
| IN22K+COCO → ADE20K | BeiTv2-L | 571M | 896x896 | 784 | **70.84** | **58.58** | 49.84 | 44.28 | **13.39** | **88.24** | **61.32** | **67.80** |
| IN22K → ADE20K | AugReg-T | 36M | 512x512 | 256 | 58.52 | 50.25 | 47.98 | 43.92 | 11.53 | 76.08 | 52.32 | 56.65 |
| IN22K → ADE20K | AugReg-B | 134M | 512x512 | 256 | 61.51 | 52.68 | 46.41 | 44.65 | 11.99 | 78.28 | 54.20 | 59.29 |
| IN22K → ADE20K | AugReg-L | 364M | 512x512 | 256 | 65.92 | 55.60 | 50.37 | 44.42 | 12.32 | 83.23 | 54.98 | 63.01 |
| IN1K → COCO | VMamba-T | 30M | 512x512 | 1024 | 66.91 | 55.30 | 45.05 | 44.43 | 11.73 | 84.86 | 58.94 | 64.22 |
| IN1K → COCO | VMamba-S | 50M | 512x512 | 1024 | 66.42 | 55.38 | 46.14 | **44.99** | 11.98 | 86.78 | 58.20 | 63.85 |
| IN1K → COCO | VMamba-B | 89M | 512x512 | 1024 | 65.70 | 55.01 | 46.07 | 44.66 | 12.05 | 87.99 | 60.07 | 63.58 |
| IN1K → COCO | VMamba-T(f) | 30M | 512x512 | 1024 | 66.77 | 55.41 | 42.75 | 44.28 | 11.72 | 85.71 | 60.19 | 64.28 |
| IN1K → COCO | VMamba-S(f) | 50M | 512x512 | 1024 | 65.98 | 55.18 | 45.69 | 44.44 | 12.02 | 87.28 | 60.42 | 63.81 |
| IN1K → COCO | VMamba-B(f) | 89M | 512x512 | 1024 | 65.63 | 55.11 | 44.13 | 44.86 | 12.15 | 87.57 | 60.69 | 63.58 |
| IN1K → ADE20K | VMamba-T | 30M | 512x512 | 1024 | 65.45 | 55.26 | 40.91 | 44.18 | 11.85 | 83.65 | 55.66 | 62.60 |
| IN1K → ADE20K | VMamba-S | 50M | 512x512 | 1024 | 66.42 | 55.68 | 47.44 | 44.42 | 11.86 | 84.01 | 53.91 | 63.21 |
| IN1K → ADE20K | VMamba-B | 89M | 512x512 | 1024 | 66.12 | 55.89 | 40.19 | 44.50 | 12.33 | 84.39 | 53.61 | 62.87 |

Table 20 | Localization benchmarks for additional backbones with non-classification pretraining or task-adapted checkpoints, alongside VMamba transfer variants. Because these settings differ from the main controlled comparison in pretraining data and objective, they are reported separately.

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| Multi-Modal → ADE20K | Perceiver | 364M | 512x512 | 256 | 63.31 | 51.55 | 55.88 | 20.81 | 42.29 | 60.14 |
| IN22K → ADE20K | BeiT-L | 451M | 640x640 | 400 | 52.99 | 42.75 | 47.81 | 16.94 | 35.22 | 61.61 |
| IN22K → ADE20K | BeiT-L | 568M | 640x640 | 400 | 56.09 | 46.49 | 51.45 | 18.60 | 37.94 | 62.58 |
| IN22K+COCO → ADE20K | BeiT-L | 571M | 896x896 | 784 | 55.53 | 46.07 | 49.43 | 18.51 | 37.45 | 62.59 |
| IN22K+COCO → ADE20K | BeiTv2-L | 571M | 896x896 | 784 | 60.34 | 49.29 | 53.33 | 24.69 | 42.34 | **64.38** |
| IN22K → ADE20K | AugReg-T | 36M | 512x512 | 256 | 41.19 | 29.14 | 34.17 | 11.32 | 25.31 | 52.44 |
| IN22K → ADE20K | AugReg-B | 134M | 512x512 | 256 | 48.32 | 36.36 | 41.44 | 15.54 | 31.29 | 55.53 |
| IN22K → ADE20K | AugReg-L | 364M | 512x512 | 256 | 56.58 | 45.16 | 50.47 | 20.28 | 38.32 | 59.69 |
| IN1K → COCO | VMamba-T | 30M | 512x512 | 1024 | 61.51 | 50.50 | 53.76 | 28.50 | 44.52 | 61.57 |
| IN1K → COCO | VMamba-S | 50M | 512x512 | 1024 | 64.64 | 52.22 | 56.35 | **32.50** | **47.60** | 61.67 |
| IN1K → COCO | VMamba-B | 89M | 512x512 | 1024 | 62.77 | 50.14 | 56.33 | 30.00 | 45.63 | 61.17 |
| IN1K → COCO | VMamba-T(f) | 30M | 512x512 | 1024 | 63.69 | 52.70 | 55.68 | 30.87 | 46.75 | 61.92 |
| IN1K → COCO | VMamba-S(f) | 50M | 512x512 | 1024 | **65.09** | 52.81 | 57.13 | 31.14 | 47.38 | 61.60 |
| IN1K → COCO | VMamba-B(f) | 89M | 512x512 | 1024 | 64.57 | 50.99 | 57.07 | 31.35 | 46.90 | 61.34 |
| IN1K → ADE20K | VMamba-T | 30M | 512x512 | 1024 | 62.65 | 51.10 | 57.15 | 24.01 | 43.47 | 60.03 |
| IN1K → ADE20K | VMamba-S | 50M | 512x512 | 1024 | 64.17 | **53.98** | **59.13** | 24.58 | 44.98 | 60.76 |
| IN1K → ADE20K | VMamba-B | 89M | 512x512 | 1024 | 63.99 | 52.52 | 58.68 | 25.91 | 45.08 | 60.48 |

Table 21 | VQA benchmarks for backbones pretrained with contrastive or self-supervised objectives, along with fused DINOv2/CLIP and DINOv2/SigLIP variants. Rows marked (f) use a stronger connector for stabilization.

| Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CLIP-B | 86M | 224x224 | 196 | 72.25 | 58.41 | 52.92 | 51.01 | 31.98 | 85.04 | 60.65 | 69.14 |
| CLIP-L | 304M | 224x224 | 256 | 74.69 | 59.15 | 53.73 | 52.87 | 37.29 | 86.20 | 60.49 | 71.13 |
| DINOv2-L | 304M | 224x224 | 256 | 64.15 | 52.37 | 46.30 | 44.37 | 12.67 | 82.18 | 56.58 | 61.68 |
| SigLIP-B16 | 93M | 224x224 | 196 | 72.52 | 58.26 | 44.73 | 50.70 | 31.48 | 83.62 | 61.16 | 69.22 |
| SigLIP-SO400M | 428M | 224x224 | 256 | 75.88 | 59.76 | 52.79 | 55.09 | 42.13 | 84.93 | 64.50 | 72.65 |
| SigLIP-B16 | 93M | 256x256 | 256 | 73.30 | 58.67 | 46.70 | 52.39 | 34.98 | 83.84 | 62.11 | 70.07 |
| CLIP-L | 304M | 336x336 | 576 | 76.63 | 60.19 | 52.98 | 57.05 | 45.78 | 87.55 | 64.43 | 73.39 |
| SigLIP-B16 | 93M | 384x384 | 576 | 75.77 | 60.05 | 47.55 | 55.64 | 44.37 | 86.13 | 62.88 | 72.37 |
| SigLIP-SO400M | 428M | 384x384 | 729 | 78.42 | 61.09 | 56.04 | **60.79** | **55.46** | 87.69 | 67.80 | 75.49 |
| DINOv2-L/SigLIP-SO400M | 304M/428M | 224x224/224x224 | 256+256 | 69.03 | 54.49 | 52.62 | 49.64 | 27.30 | 83.97 | 60.79 | 66.45 (-6.58) |
| DINOv2-L/CLIP-L | 304M/304M | 336x336/336x336 | 576+576 | 65.08 | 52.50 | 47.65 | 45.29 | 12.93 | 81.98 | 58.03 | 62.60 (-8.34) |
| DINOv2-L/SigLIP-SO400M | 304M/428M | 384x384/384x384 | 729+729 | 67.83 | 52.95 | 51.29 | 50.15 | 28.31 | 83.39 | 60.30 | 65.42 (-10.33) |
| DINOv2-L/SigLIP-SO400M(f) | 304M/428M | 224x224/224x224 | 256+256 | 76.09 | 60.24 | 56.29 | 54.48 | 39.41 | 86.47 | 65.75 | 73.03 |
| DINOv2-L/CLIP-L(f) | 304M/304M | 336x336/336x336 | 576+576 | 74.34 | 60.96 | 52.69 | 45.96 | 15.22 | 88.09 | 63.84 | 70.94 |
| DINOv2-L/SigLIP-SO400M(f) | 304M/428M | 384x384/384x384 | 729+729 | **78.59** | **61.66** | **59.36** | 59.93 | 52.58 | **88.31** | **68.52** | **75.75** |

Table 22 | Localization benchmarks for backbones pretrained with contrastive or self-supervised objectives, along with fused DINOv2/CLIP and DINOv2/SigLIP variants. Rows marked (f) use a stronger connector for stabilization. Direct fusion can hurt localization, while stabilized fusion substantially improves grounding performance and yields the strongest results in this group.

| Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|
| CLIP-B | 86M | 224x224 | 196 | 64.85 | 55.57 | 60.05 | 28.40 | 47.19 | 66.19 |
| CLIP-L | 304M | 224x224 | 256 | 68.05 | 58.26 | 62.79 | 32.70 | 50.66 | 68.38 |
| DINOv2-L | 304M | 224x224 | 256 | 27.53 | 16.46 | 18.40 | 4.07 | 14.28 | 55.31 |
| SigLIP-B16 | 93M | 224x224 | 196 | 63.85 | 54.60 | 60.36 | 23.78 | 44.85 | 65.94 |
| SigLIP-SO400M | 428M | 224x224 | 256 | 63.26 | 54.39 | 57.60 | 23.61 | 44.29 | 68.84 |
| SigLIP-B16 | 93M | 256x256 | 256 | 64.77 | 55.78 | 60.09 | 25.62 | 46.08 | 66.85 |
| CLIP-L | 304M | 336x336 | 576 | 72.31 | 63.81 | 68.69 | 36.19 | 55.10 | 70.93 |
| SigLIP-B16 | 93M | 384x384 | 576 | 71.57 | 62.79 | 66.99 | 34.81 | 53.92 | 69.89 |
| SigLIP-SO400M | 428M | 384x384 | 729 | 69.17 | 60.98 | 66.89 | 34.46 | 52.75 | 72.44 |
| DINOv2-L/SigLIP-SO400M | 304M/428M | 224x224/224x224 | 256+256 | 33.07 | 22.33 | 26.51 | 4.04 | 17.90 (-35.66) | 59.93 (-10.49) |
| DINOv2-L/CLIP-L | 304M/304M | 336x336/336x336 | 576+576 | 27.99 | 15.85 | 18.85 | 3.16 | 13.92 (-45.48) | 56.06 (-13.33) |
| DINOv2-L/SigLIP-SO400M | 304M/428M | 384x384/384x384 | 729+729 | 29.45 | 17.24 | 21.81 | 4.29 | 15.39 (-48.99) | 58.69 (-15.53) |
| DINOv2-L/SigLIP-SO400M(f) | 304M/428M | 224x224/224x224 | 256+256 | 69.87 | 61.21 | 65.83 | 36.17 | 53.56 | 70.42 |
| DINOv2-L/CLIP-L(f) | 304M/304M | 336x336/336x336 | 576+576 | 75.94 | 66.57 | 72.18 | 42.01 | 59.40 | 69.39 |
| DINOv2-L/SigLIP-SO400M(f) | 304M/428M | 384x384/384x384 | 729+729 | **78.83** | **70.25** | **75.29** | **49.50** | **64.38** | **74.22** |

## F.2. Contrastive/Self-Supervised Backbones

*Alternative pretraining objectives.* Tables 21 and 22 report additional backbones pretrained with contrastive or self-supervised objectives, as well as fused encoders built from them. Large contrastive and self-supervised backbones already achieve strong performance in this setting, especially when paired with larger models and higher input resolutions. Direct fusion, however, is not always stable, particularly for localization. Once stabilized with a stronger connector, the fused DINOv2 and SigLIP models become the strongest results in this group, supporting our observation that combining complementary pretraining objectives can further improve VLM performance. At the same time, these models use substantially larger vision stacks and higher token budgets than the standard VMamba setting, so we report them separately from the main controlled comparison. Overall, these results suggest that bringing contrastive or self-supervised objectives to VMamba-style backbones is a promising direction for future work.

## G. Single-GPU Inference Profiling

### G.1. Profiling Setup

We profile inference efficiency for representative ViT, VMamba, and ViTDet backbones inside the same VLM wrapper on a single GPU. We choose these three families to cover different practical trade-offs. ViT provides a standard transformer baseline at a scale similar to VMamba, which allows a direct comparison between two backbone families with similar model size but different VLM performance. ViTDet is included as a larger and competitive alternative, allowing us to compare against a higher-capacity vision backbone with a different runtime and memory profile.

For each backbone and input resolution, we rebuild the vision encoder at the target square resolution, combine it with the same projector and LLM, and run a full multimodal forward pass under `torch.inference_mode()`. To focus on how inference cost changes with resolution, we use synthetic inputs instead of dataset samples. Unless otherwise stated, the batch size is 1 and the text length is fixed to 128 tokens. Each configuration is run for 200 forward iterations. We treat the first 100 iterations as warm-up and report the average of the final 100. For each model, we sweep resolutions in ascending order and stop at the first out-of-memory setting.

We report six metrics in Fig. 4. **Host-side vision latency** measures the wall-clock time of the vision stage, including the vision backbone and projector. **Host-side LLM latency** measures the wall-clock time of the language stage, including multimodal input preparation and the LLM forward pass. **GPU vision latency** and **GPU LLM latency** measure how much time the GPU spends on the vision and language stages, respectively. **End-to-end VLM latency** measures the total wall-clock time of one multimodal forward pass. **Peak allocated GPU memory** records the largest GPU memory allocation during the forward pass.

These metrics capture different views of inference cost. The host-side timings reflect the overall time seen from the CPU side, so they include both model execution and the overhead of preparing and launching the computation. The GPU timings focus on the computation time spent on the GPU. Because these two views measure different parts of the system, they do not need to match exactly, and their sums do not necessarily equal the end-to-end latency.

In practice, end-to-end VLM latency is the most direct measure of inference speed. The stagewise GPU timings show where the main computation cost lies, the host-side timings show the overall runtime overhead of each stage, and peak memory determines the largest resolution that can be run. Ignoring the first 100 iterations reduces startup effects and gives a more stable estimate of steady-state inference cost.

### G.2. Observations

Figure 4 shows a clear practical trade-off across the three backbone families. ViT and VMamba operate in a broadly similar efficiency regime, while ViTDet is substantially heavier and reaches the memory limit much earlier. This makes the comparison informative: ViT serves as a scale-matched baseline for VMamba, and ViTDet shows the cost of using a much larger vision backbone.

First, Fig. 4(f) shows that ViTDet runs out of memory at much lower resolutions than ViT and VMamba. This is consistent with its much larger model size and indicates that memory, rather than latency alone, quickly becomes the limiting factor for this family.

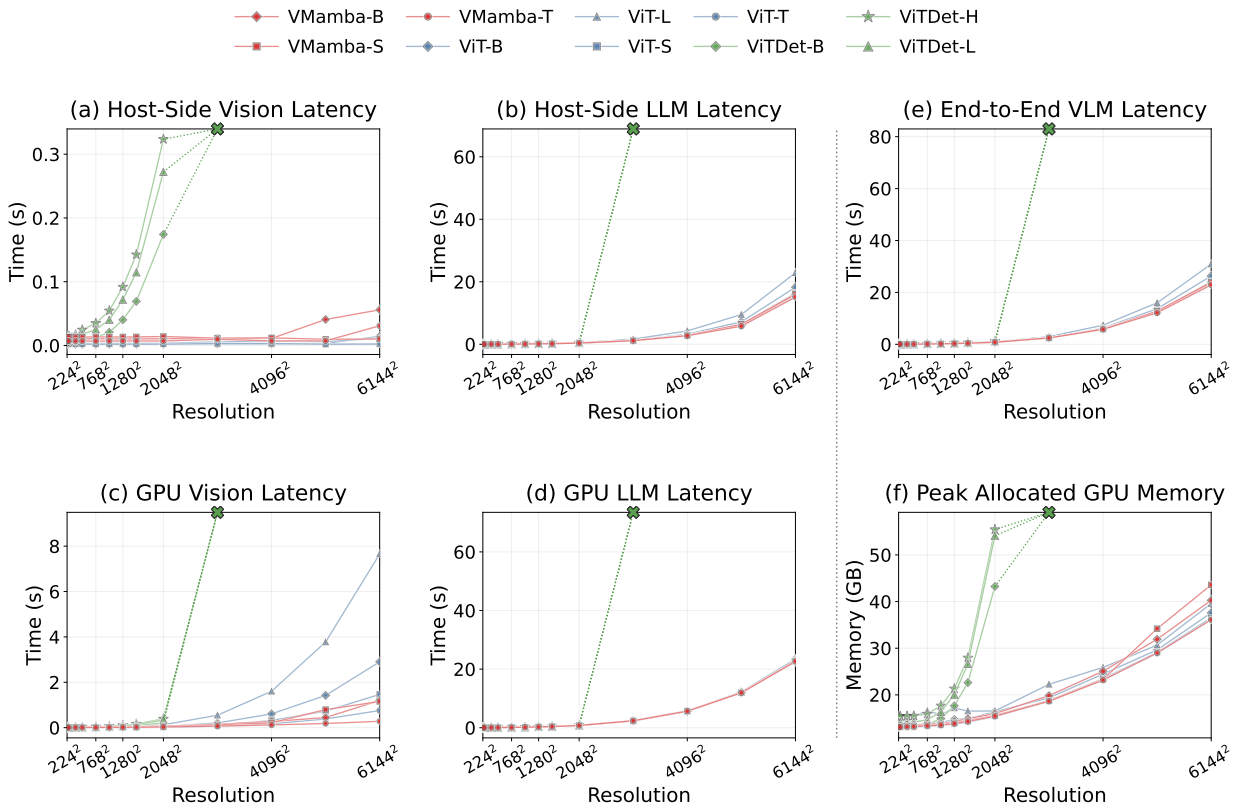


Figure 4 | **Scaling of single-GPU VLM inference cost with image resolution at batch size 1.** The x-axis shows the square input resolution, written as $R^2$. The six panels report (a) host-side vision latency, (b) host-side LLM latency, (c) GPU vision latency, (d) GPU LLM latency, (e) end-to-end VLM latency, and (f) peak allocated GPU memory. For each model and resolution, we run 100 warm-up iterations and report the mean over the next 100 iterations. Dotted segments ending with an $\times$ indicate the first resolution that exceeds GPU memory capacity. The marker is placed at the top of each subplot to show the out-of-memory threshold rather than a measured metric value. In panel (f), it therefore marks the first failed resolution, not an observed peak-memory measurement.

Second, Fig. 4(c) shows that GPU vision latency grows more rapidly for ViT as the input resolution increases, whereas VMamba scales more gradually. This suggests that VMamba handles higher-resolution visual inputs more efficiently in the vision stage. Fig. 4(a) shows a similar trend on the host side: vision-stage overhead remains relatively small for ViT and VMamba, but becomes much larger for the heavier ViTDet models.

Third, Fig. 4(b), (d), and (e) show that the language stage dominates the overall inference time for most settings. For ViT and VMamba, the end-to-end latency curves are much closer to the LLM-stage curves than to the vision-stage curves, which indicates that the vision module is often a smaller part of total VLM inference cost at this scale. The vision-stage differences become more visible mainly at very high resolutions or for much larger backbones such as ViTDet.

Overall, ViT and VMamba have similar practical inference cost at comparable scale, but VMamba achieves stronger VLM performance in our main results. ViTDet is also competitive in VLM performance, but it is much larger and reaches out-of-memory much earlier. Taken together, these results make VMamba an attractive design point: it offers a stronger performance–efficiency trade-off than a similarly sized ViT baseline, while leaving more room than ViTDet for future scaling and system design choices.

## H. Exhaustive Results

For completeness, we provide exhaustive VQA and localization results for all evaluated backbone families. Specifically, Tables 23 and 24 report VMamba, Tables 25 and 26 report ViT, Tables 27 and 28 report MaxViT, Tables 29 and 30 report MambaVision, Tables 31 and 32 report Vim, Tables 33 and 34 report ViTDet, and Tables 35 and 36 report ViT-Adapter.

Table 23 | VMamba VQA benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | VMamba-T | 30M | 224x224 | 196 | 64.99 | 54.02 | 44.96 | 44.62 | 12.22 | 81.70 | 54.58 | 62.07 |
| IN1K | VMamba-S | 50M | 224x224 | 196 | 65.24 | 54.08 | 44.95 | 44.06 | 12.24 | 82.20 | 55.54 | 62.39 |
| IN1K | VMamba-B | 80M | 224x224 | 196 | 64.20 | 53.25 | 47.97 | 43.81 | 12.08 | 80.75 | 54.05 | 61.38 |
| IN1K | VMamba-T(f) | 30M | 224x224 | 196 | 65.22 | 53.87 | 44.88 | 44.84 | 12.19 | 81.19 | 56.29 | 62.45 |
| IN1K | VMamba-S(f) | 50M | 224x224 | 196 | 65.49 | 54.75 | 47.40 | 43.77 | **12.40** | 81.70 | 55.93 | 62.68 |
| IN1K | VMamba-B(f) | 80M | 224x224 | 196 | 64.51 | 53.81 | 48.83 | 43.97 | 12.36 | 81.83 | 56.38 | 62.00 |
| IN1K → COCO | VMamba-T | 30M | 1333x800 | 4150 | 60.24 | 49.18 | **49.04** | 44.06 | 11.81 | 79.48 | 52.65 | 58.05 |
| IN1K → COCO | VMamba-S | 50M | 1333x800 | 4150 | 64.87 | 53.98 | 47.73 | 44.24 | 11.93 | 86.47 | 59.21 | 62.78 |
| IN1K → COCO | VMamba-B | 89M | 1333x800 | 4150 | 60.36 | 49.21 | 46.46 | 43.84 | 11.73 | 81.61 | 55.69 | 58.57 |
| IN1K → COCO | VMamba-T(f) | 30M | 1333x800 | 4150 | 64.46 | 53.33 | 45.21 | 43.68 | 11.90 | 84.14 | 54.27 | 61.66 |
| IN1K → COCO | VMamba-S(f) | 50M | 1333x800 | 4150 | 64.31 | 53.84 | 43.31 | 44.20 | 12.23 | 86.36 | 57.13 | 62.01 |
| IN1K → COCO | VMamba-B(f) | 89M | 1333x800 | 4150 | 60.85 | 50.25 | 40.44 | 43.99 | 11.77 | 82.71 | 55.04 | 58.84 |
| IN1K → COCO | VMamba-T | 30M | 512x512 | 1024 | **66.91** | 55.30 | 45.05 | 44.43 | 11.73 | 84.86 | 58.94 | 64.22 |
| IN1K → COCO | VMamba-S | 50M | 512x512 | 1024 | 66.42 | 55.38 | 46.14 | **44.99** | 11.98 | 86.78 | 58.20 | 63.85 |
| IN1K → COCO | VMamba-B | 89M | 512x512 | 1024 | 65.70 | 55.01 | 46.07 | 44.66 | 12.05 | **87.99** | 60.07 | 63.58 |
| IN1K → COCO | VMamba-T(f) | 30M | 512x512 | 1024 | 66.77 | 55.41 | 42.75 | 44.28 | 11.72 | 85.71 | 60.19 | **64.28** |
| IN1K → COCO | VMamba-S(f) | 50M | 512x512 | 1024 | 65.98 | 55.18 | 45.69 | 44.44 | 12.02 | 87.28 | 60.42 | 63.81 |
| IN1K → COCO | VMamba-B(f) | 89M | 512x512 | 1024 | 65.63 | 55.11 | 44.13 | 44.86 | 12.15 | 87.57 | **60.69** | 63.58 |
| IN1K → ADE20K | VMamba-T | 30M | 512x512 | 1024 | 65.45 | 55.26 | 40.91 | 44.18 | 11.85 | 83.65 | 55.66 | 62.60 |
| IN1K → ADE20K | VMamba-S | 50M | 512x512 | 1024 | 66.42 | 55.68 | 47.44 | 44.42 | 11.86 | 84.01 | 53.91 | 63.21 |
| IN1K → ADE20K | VMamba-B | 89M | 512x512 | 1024 | 66.12 | **55.89** | 40.19 | 44.50 | 12.33 | 84.39 | 53.61 | 62.87 |

Table 24 | VMamba localization benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | VMamba-T | 30M | 224x224 | 196 | 58.25 | 46.64 | 51.74 | 20.24 | 39.20 | 59.00 |
| IN1K | VMamba-S | 50M | 224x224 | 196 | 56.48 | 44.27 | 49.88 | 23.09 | 39.17 | 59.27 |
| IN1K | VMamba-B | 80M | 224x224 | 196 | 42.06 | 31.43 | 36.15 | 15.23 | 27.89 | 56.88 |
| IN1K | VMamba-T(f) | 30M | 224x224 | 196 | 58.22 | 47.30 | 51.76 | 22.00 | 40.07 | 59.44 |
| IN1K | VMamba-S(f) | 50M | 224x224 | 196 | 54.65 | 43.94 | 49.24 | 24.35 | 39.09 | 59.51 |
| IN1K | VMamba-B(f) | 80M | 224x224 | 196 | 45.35 | 33.96 | 39.32 | 16.82 | 30.29 | 57.74 |
| IN1K → COCO | VMamba-T | 30M | 1333x800 | 4150 | 29.10 | 16.83 | 19.89 | 3.95 | 14.86 | 52.25 |
| IN1K → COCO | VMamba-S | 50M | 1333x800 | 4150 | **69.52** | 52.87 | **63.50** | 28.15 | **47.94** | 60.78 |
| IN1K → COCO | VMamba-B | 89M | 1333x800 | 4150 | 28.43 | 16.44 | 19.87 | 4.97 | 15.02 | 52.72 |
| IN1K → COCO | VMamba-T(f) | 30M | 1333x800 | 4150 | 57.10 | 42.55 | 50.10 | 20.64 | 37.93 | 58.47 |
| IN1K → COCO | VMamba-S(f) | 50M | 1333x800 | 4150 | 67.40 | 51.02 | 61.38 | 28.89 | 47.06 | 60.00 |
| IN1K → COCO | VMamba-B(f) | 89M | 1333x800 | 4150 | 31.47 | 19.10 | 22.24 | 3.95 | 16.23 | 53.11 |
| IN1K → COCO | VMamba-T | 30M | 512x512 | 1024 | 61.51 | 50.50 | 53.76 | 28.50 | 44.52 | 61.57 |
| IN1K → COCO | VMamba-S | 50M | 512x512 | 1024 | 64.64 | 52.22 | 56.35 | **32.50** | 47.60 | 61.67 |
| IN1K → COCO | VMamba-B | 89M | 512x512 | 1024 | 62.77 | 50.14 | 56.33 | 30.00 | 45.63 | 61.17 |
| IN1K → COCO | VMamba-T(f) | 30M | 512x512 | 1024 | 63.69 | 52.70 | 55.68 | 30.87 | 46.75 | **61.92** |
| IN1K → COCO | VMamba-S(f) | 50M | 512x512 | 1024 | 65.09 | 52.81 | 57.13 | 31.14 | 47.38 | 61.60 |
| IN1K → COCO | VMamba-B(f) | 89M | 512x512 | 1024 | 64.57 | 50.99 | 57.07 | 31.35 | 46.90 | 61.34 |
| IN1K → ADE20K | VMamba-T | 30M | 512x512 | 1024 | 62.65 | 51.10 | 57.15 | 24.01 | 43.47 | 60.03 |
| IN1K → ADE20K | VMamba-S | 50M | 512x512 | 1024 | 64.17 | **53.98** | 59.13 | 24.58 | 44.98 | 60.76 |
| IN1K → ADE20K | VMamba-B | 89M | 512x512 | 1024 | 63.99 | 52.52 | 58.68 | 25.91 | 45.08 | 60.48 |

Table 25 | ViT VQA benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN21k | ViT-T | 5.7M | 224x224 | 196 | 62.82 | 51.30 | 46.03 | 44.62 | 11.90 | 76.94 | 52.65 | 59.95 |
| IN21k | ViT-S | 22M | 224x224 | 196 | 61.28 | 50.34 | 47.95 | 44.18 | 12.30 | 78.59 | 53.62 | 58.97 |
| IN21k | ViT-B | 87M | 224x224 | 196 | **68.16** | **55.96** | **49.40** | **44.75** | **12.77** | 82.34 | 55.26 | **64.70** |
| IN21k | ViT-L | 304M | 224x224 | 196 | 67.13 | 54.85 | 49.15 | 44.28 | 12.64 | **82.38** | 53.32 | 63.61 |
| IN21k → IN1K | ViT-T | 5.7M | 224x224 | 196 | 61.34 | 50.00 | 45.30 | 44.02 | 11.84 | 77.90 | 53.82 | 58.96 |
| IN21k → IN1K | ViT-S | 22M | 224x224 | 196 | 60.39 | 49.46 | 44.54 | 44.58 | 12.23 | 78.49 | 51.60 | 57.95 |
| IN21k → IN1K | ViT-B | 87M | 224x224 | 196 | 67.24 | 54.96 | 47.76 | 43.92 | 12.34 | 82.16 | **57.16** | 64.17 |
| IN21k → IN1K | ViT-L | 304M | 224x224 | 196 | 62.17 | 50.82 | 42.10 | 44.40 | 12.71 | 78.42 | 53.47 | 59.55 |
| IN1K | ViT-S | 22M | 224x224 | 196 | 59.86 | 50.48 | 46.69 | 44.02 | 12.30 | 77.76 | 48.96 | 57.25 |
| IN1K | ViT-B | 87M | 224x224 | 196 | 59.63 | 49.19 | 45.63 | 44.51 | 12.07 | 75.90 | 50.57 | 57.17 |

Table 26 | ViT localization benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN21k | ViT-T | 5.7M | 224x224 | 196 | 34.52 | 21.72 | 26.47 | 7.30 | 19.43 | 54.50 |
| IN21k | ViT-S | 22M | 224x224 | 196 | 24.40 | 13.79 | 17.42 | 4.73 | 13.04 | 52.80 |
| IN21k | ViT-B | 87M | 224x224 | 196 | **55.00** | 44.79 | **48.35** | **19.90** | **37.46** | **61.04** |
| IN21k | ViT-L | 304M | 224x224 | 196 | 40.64 | 31.38 | 35.21 | 11.40 | 25.86 | 58.54 |
| IN21k → IN1K | ViT-T | 5.7M | 224x224 | 196 | 31.56 | 19.84 | 23.79 | 8.35 | 18.40 | 53.51 |
| IN21k → IN1K | ViT-S | 22M | 224x224 | 196 | 24.29 | 14.05 | 16.99 | 4.05 | 12.75 | 51.88 |
| IN21k → IN1K | ViT-B | 87M | 224x224 | 196 | 54.82 | **45.00** | 48.30 | 18.44 | 36.87 | 60.50 |
| IN21k → IN1K | ViT-L | 304M | 224x224 | 196 | 22.92 | 13.04 | 15.77 | 3.18 | 11.69 | 53.12 |
| IN1K | ViT-S | 22M | 224x224 | 196 | 32.32 | 21.77 | 24.80 | 5.08 | 17.82 | 51.95 |
| IN1K | ViT-B | 87M | 224x224 | 196 | 26.66 | 15.41 | 19.12 | 4.14 | 13.92 | 51.36 |

Table 27 | MaxViT VQA benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN21K | MaxViT-B | 119M | 224x224 | 196 | 62.67 | 49.98 | 45.92 | 44.08 | 12.38 | 76.46 | 53.58 | 59.88 |
| IN21K | MaxViT-L | 212M | 224x224 | 196 | 65.50 | 52.03 | 47.31 | 44.30 | 12.49 | 80.59 | 53.62 | 62.23 |
| IN21K | MaxViT-XL | 475M | 224x224 | 196 | 65.55 | 51.61 | 50.14 | 44.13 | 12.58 | 80.07 | 54.78 | 62.43 |
| IN21K → IN1K | MaxViT-B | 119M | 384x384 | 576 | 63.44 | 50.92 | 50.41 | 44.38 | 12.71 | 76.72 | 54.53 | 60.71 |
| IN21K → IN1K | MaxViT-L | 212M | 384x384 | 576 | 67.15 | 52.82 | **53.89** | 43.72 | **13.28** | 82.30 | **55.45** | **63.89** |
| IN21K → IN1K | MaxViT-XL | 475M | 384x384 | 576 | 66.37 | 52.53 | 49.11 | 44.16 | 12.89 | 81.46 | 53.58 | 62.95 |
| IN21K → IN1K | MaxViT-B | 119M | 512x512 | 1024 | 63.42 | 51.07 | 49.87 | 44.10 | 12.53 | 77.67 | 54.35 | 60.69 |
| IN21K → IN1K | MaxViT-L | 212M | 512x512 | 1024 | 67.27 | 52.70 | 51.91 | **44.71** | 13.09 | 83.11 | 54.38 | 63.84 |
| IN21K → IN1K | MaxViT-XL | 475M | 512x512 | 1024 | **67.34** | 52.72 | 51.08 | 44.16 | 12.95 | **83.15** | 52.83 | 63.67 |
| IN1K | MaxViT-T | 31M | 384x384 | 576 | 64.04 | 53.04 | 48.00 | 44.69 | 11.71 | 79.86 | 53.30 | 61.13 |
| IN1K | MaxViT-S | 69M | 384x384 | 576 | 61.31 | 49.85 | 46.73 | 44.19 | 11.79 | 77.57 | 52.05 | 58.70 |
| IN1K | MaxViT-B | 119M | 384x384 | 576 | 63.90 | 52.76 | 46.61 | 43.95 | 12.27 | 81.23 | 52.83 | 60.97 |
| IN1K | MaxViT-L | 212M | 384x384 | 576 | 62.90 | 51.26 | 49.94 | 43.80 | 12.41 | 81.00 | 49.25 | 59.73 |
| IN1K | MaxViT-T | 31M | 512x512 | 1024 | 64.25 | 52.84 | 48.14 | 44.40 | 11.84 | 79.24 | 51.93 | 61.08 |
| IN1K | MaxViT-S | 69M | 512x512 | 1024 | 63.51 | 51.26 | 46.29 | 44.25 | 11.92 | 79.36 | 54.01 | 60.71 |
| IN1K | MaxViT-B | 119M | 512x512 | 1024 | 64.53 | **53.14** | 47.92 | 44.10 | 12.09 | 82.90 | 53.06 | 61.56 |
| IN1K | MaxViT-L | 212M | 512x512 | 1024 | 63.19 | 51.17 | 46.09 | 43.63 | 12.41 | 81.30 | 48.87 | 59.84 |
| IN1K | MaxViT-T | 31M | 224x224 | 196 | 61.08 | 49.43 | 47.58 | 44.31 | 12.02 | 76.77 | 53.85 | 58.75 |
| IN1K | MaxViT-S | 69M | 224x224 | 196 | 59.90 | 48.94 | 41.50 | 44.51 | 11.82 | 76.29 | 51.42 | 57.42 |
| IN1K | MaxViT-B | 119M | 224x224 | 196 | 60.10 | 49.72 | 45.00 | 44.59 | 12.29 | 76.26 | 50.94 | 57.60 |
| IN1K | MaxViT-L | 212M | 224x224 | 196 | 60.07 | 49.55 | 46.42 | 43.67 | 12.23 | 76.46 | 48.86 | 57.30 |

Table 28 | MaxViT localization benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN21K | MaxViT-B | 119M | 224x224 | 196 | 23.04 | 12.72 | 16.44 | 2.56 | 11.46 | 53.38 |
| IN21K | MaxViT-L | 212M | 224x224 | 196 | 26.27 | 15.83 | 19.53 | 4.94 | 14.30 | 55.79 |
| IN21K | MaxViT-XL | 475M | 224x224 | 196 | 24.43 | 13.70 | 18.34 | 4.46 | 13.02 | 55.79 |
| IN21K → IN1K | MaxViT-B | 119M | 384x384 | 576 | 21.22 | 12.01 | 14.56 | 2.87 | 10.77 | 54.00 |
| IN21K → IN1K | MaxViT-L | 212M | 384x384 | 576 | 26.46 | 15.92 | 19.36 | 5.79 | 14.70 | 57.28 |
| IN21K → IN1K | MaxViT-XL | 475M | 384x384 | 576 | 24.67 | 14.46 | 18.32 | 3.67 | 12.93 | 56.23 |
| IN21K → IN1K | MaxViT-B | 119M | 512x512 | 1024 | 21.72 | 12.49 | 15.26 | 2.58 | 10.97 | 54.01 |
| IN21K → IN1K | MaxViT-L | 212M | 512x512 | 1024 | 26.45 | 15.50 | 20.26 | 5.22 | 14.46 | 57.21 |
| IN21K → IN1K | MaxViT-XL | 475M | 512x512 | 1024 | 26.67 | 15.66 | 20.20 | 5.33 | 14.59 | 57.07 |
| IN1K | MaxViT-T | 31M | 384x384 | 576 | **55.01** | 43.06 | 47.75 | 22.30 | 37.97 | 58.02 |
| IN1K | MaxViT-S | 69M | 384x384 | 576 | 25.68 | 14.97 | 18.22 | 4.08 | 13.46 | 52.62 |
| IN1K | MaxViT-B | 119M | 384x384 | 576 | 28.18 | 18.45 | 20.61 | 6.08 | 15.98 | 54.92 |
| IN1K | MaxViT-L | 212M | 384x384 | 576 | 23.92 | 13.65 | 17.12 | 5.36 | 13.12 | 53.46 |
| IN1K | MaxViT-T | 31M | 512x512 | 1024 | 54.54 | **43.28** | **48.24** | **23.42** | **38.42** | **58.03** |
| IN1K | MaxViT-S | 69M | 512x512 | 1024 | 35.07 | 23.51 | 28.10 | 9.29 | 20.99 | 55.37 |
| IN1K | MaxViT-B | 119M | 512x512 | 1024 | 33.23 | 22.73 | 25.14 | 8.34 | 19.64 | 55.92 |
| IN1K | MaxViT-L | 212M | 512x512 | 1024 | 25.60 | 15.34 | 18.83 | 5.90 | 14.34 | 53.72 |
| IN1K | MaxViT-T | 31M | 224x224 | 196 | 29.44 | 17.29 | 22.10 | 5.17 | 15.79 | 52.98 |
| IN1K | MaxViT-S | 69M | 224x224 | 196 | 25.57 | 14.41 | 17.28 | 4.38 | 13.32 | 51.49 |
| IN1K | MaxViT-B | 119M | 224x224 | 196 | 22.02 | 12.67 | 15.28 | 3.36 | 11.41 | 51.39 |
| IN1K | MaxViT-L | 212M | 224x224 | 196 | 21.15 | 12.01 | 14.77 | 2.94 | 10.81 | 51.05 |

Table 29 | MambaVision VQA benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN21K | MambaVision-B | 98M | 224x224 | 196 | 49.42 | 42.07 | 41.98 | 42.71 | 10.36 | 58.90 | 42.05 | 47.50 |
| IN21K | MambaVision-L | 228M | 224x224 | 196 | 52.06 | 43.27 | 39.10 | 43.12 | 10.64 | 65.22 | 43.49 | 49.87 |
| IN21K | MambaVision-L2 | 242M | 512x512 | 1024 | 50.80 | 41.95 | 41.57 | 42.79 | 10.50 | 62.97 | 41.93 | 48.63 |
| IN21K | MambaVision-L3 | 740M | 256x256 | 256 | 52.18 | 43.58 | 36.27 | 42.94 | 10.66 | 65.13 | 38.51 | 49.26 |
| IN21K | MambaVision-L3 | 740M | 512x512 | 1024 | 51.50 | 42.62 | 40.91 | 43.40 | 10.45 | 61.98 | 43.19 | 49.32 |
| IN1K | MambaVision-T | 32M | 224x224 | 196 | 56.85 | 47.43 | 46.16 | 43.74 | 11.14 | 71.34 | 46.89 | 54.38 |
| IN1K | MambaVision-T2 | 35M | 224x224 | 196 | 56.69 | 47.14 | 43.17 | 43.31 | 10.82 | 69.58 | 43.74 | 53.71 |
| IN1K | MambaVision-S | 50M | 224x224 | 196 | 58.10 | 48.05 | 40.91 | 43.08 | 11.36 | 72.03 | 49.48 | 55.61 |
| IN1K | MambaVision-B | 98M | 224x224 | 196 | 59.14 | 49.23 | 44.21 | 43.33 | **11.43** | 73.05 | 49.53 | 56.53 |
| IN1K | MambaVision-L | 228M | 224x224 | 196 | 59.26 | **49.56** | 42.13 | **43.75** | 11.41 | **74.46** | **51.70** | **56.94** |
| IN1K | MambaVision-L2 | 242M | 224x224 | 196 | **59.27** | 48.83 | **48.91** | 43.30 | 11.16 | 74.15 | 50.70 | 56.86 |

Table 30 | MambaVision localization benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN21K | MambaVision-B | 98M | 224x224 | 196 | 19.44 | 10.68 | 12.99 | 3.52 | 10.12 | 42.48 |
| IN21K | MambaVision-L | 228M | 224x224 | 196 | 22.52 | 11.91 | 14.69 | 3.91 | 11.50 | 44.72 |
| IN21K | MambaVision-L2 | 242M | 512x512 | 1024 | 21.80 | 12.46 | 14.17 | 3.45 | 11.22 | 43.60 |
| IN21K | MambaVision-L3 | 740M | 256x256 | 256 | 21.39 | 11.89 | 13.93 | 3.71 | 11.06 | 44.13 |
| IN21K | MambaVision-L3 | 740M | 512x512 | 1024 | 21.27 | 11.61 | 13.46 | 3.66 | 10.89 | 44.15 |
| IN1K | MambaVision-T | 32M | 224x224 | 196 | 34.59 | 24.01 | 27.14 | 9.52 | 20.98 | 49.89 |
| IN1K | MambaVision-T2 | 35M | 224x224 | 196 | 35.78 | 24.69 | 27.61 | 9.72 | 21.56 | 49.39 |
| IN1K | MambaVision-S | 50M | 224x224 | 196 | 44.65 | 33.06 | 37.48 | 13.21 | 28.22 | 51.93 |
| IN1K | MambaVision-B | 98M | 224x224 | 196 | **48.52** | 35.57 | **40.58** | 15.84 | 31.17 | 53.12 |
| IN1K | MambaVision-L | 228M | 224x224 | 196 | 48.19 | **35.82** | 40.26 | **16.79** | **31.51** | **53.52** |
| IN1K | MambaVision-L2 | 242M | 224x224 | 196 | 47.91 | 35.41 | 40.28 | 16.48 | 31.22 | 53.42 |

Table 31 | Vim VQA benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | Vim-T | 7M | 224x224 | 196 | 55.83 | 46.93 | 39.39 | 43.82 | 11.67 | 72.91 | 45.69 | 53.40 |
| IN1K | Vim-T-ft | 7M | 224x224 | 729 | 56.82 | 47.62 | 41.51 | 43.56 | 11.76 | 74.69 | 47.73 | 54.52 |
| IN1K | Vim-S | 26M | 224x224 | 196 | 56.84 | 47.73 | 40.47 | 43.81 | 11.58 | 73.58 | 49.58 | 54.74 |
| IN1K | Vim-S-ft | 26M | 224x224 | 729 | 57.68 | 48.55 | **45.68** | 43.78 | 11.89 | 74.31 | 45.96 | 55.02 |
| IN1K | Vim-B | 98M | 224x224 | 196 | **62.93** | **52.11** | 45.52 | **44.81** | **12.50** | **78.97** | **55.07** | **60.46** |

Table 32 | Vim localization benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN1K | Vim-T | 7M | 224x224 | 196 | 21.35 | 12.42 | 14.99 | 3.51 | 11.21 | 47.73 |
| IN1K | Vim-T-ft | 7M | 224x224 | 729 | 21.28 | 12.30 | 15.13 | 3.96 | 11.37 | 48.72 |
| IN1K | Vim-S | 26M | 224x224 | 196 | 22.33 | 12.60 | 15.32 | 2.70 | 11.20 | 48.88 |
| IN1K | Vim-S-ft | 26M | 224x224 | 729 | 22.99 | 13.09 | 15.01 | 3.57 | 11.80 | 49.21 |
| IN1K | Vim-B | 98M | 224x224 | 196 | **44.62** | **34.10** | **38.54** | **13.17** | **28.56** | **56.17** |

Table 33 | ViTDet VQA benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IN1K → COCO | ViTDet-B | 111M | 1024x1024 | 4096 | 65.83 | 53.61 | **50.58** | 43.75 | 11.60 | 84.46 | 55.96 | 63.00 |
| IN1K → COCO | ViTDet-L | 331M | 1024x1024 | 4096 | 60.00 | 48.59 | 44.78 | 43.88 | 11.78 | 78.14 | 51.94 | 57.64 |
| IN1K → COCO | ViTDet-H | 662M | 1024x1024 | 4096 | 64.43 | 52.62 | 48.99 | 43.93 | 11.50 | 84.29 | 55.95 | 61.89 |
| IN1K → COCO | ViTDet-B(f) | 111M | 1024x1024 | 4096 | 63.68 | 52.14 | 46.14 | 43.64 | 11.69 | 82.58 | 53.52 | 60.89 |
| IN1K → COCO | ViTDet-L(f) | 331M | 1024x1024 | 4096 | 66.43 | 53.54 | 45.52 | 43.76 | 11.58 | 84.82 | 57.24 | 63.55 |
| IN1K → COCO | ViTDet-H(f) | 662M | 1024x1024 | 4096 | **66.89** | **54.22** | 41.01 | **44.04** | **11.82** | **87.03** | **58.08** | **64.05** |

Table 34 | ViTDet localization benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| IN1K → COCO | ViTDet-B | 111M | 1024x1024 | 4096 | 66.03 | 51.17 | 58.17 | 22.37 | 43.74 | 60.42 |
| IN1K → COCO | ViTDet-L | 331M | 1024x1024 | 4096 | 24.62 | 13.79 | 17.44 | 4.62 | 13.05 | 51.65 |
| IN1K → COCO | ViTDet-H | 662M | 1024x1024 | 4096 | 56.41 | 40.48 | 50.25 | 16.01 | 35.38 | 58.33 |
| IN1K → COCO | ViTDet-B(f) | 111M | 1024x1024 | 4096 | 46.03 | 33.49 | 39.40 | 11.72 | 28.26 | 56.51 |
| IN1K → COCO | ViTDet-L(f) | 331M | 1024x1024 | 4096 | 65.73 | 52.09 | 58.86 | 23.87 | 44.58 | 61.00 |
| IN1K → COCO | ViTDet-H(f) | 662M | 1024x1024 | 4096 | **69.37** | **54.27** | **61.97** | **26.51** | **47.40** | **61.81** |

Table 35 | ViT-Adapter VQA benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | VQA-v2 | GQA | VizWiz | TextVQA +OCR | TextVQA | POPE | TallyQA | Weighted VQA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Multi-Modal → ADE20K | Perceiver | 364M | 512x512 | 256 | 65.62 | 54.63 | 46.25 | 44.26 | 12.25 | 83.79 | 56.62 | 62.92 |
| IN22K → ADE20K | BeiT-L | 451M | 640x640 | 400 | 68.78 | 56.52 | 48.06 | **44.75** | 13.01 | 86.88 | 58.24 | 65.70 |
| IN22K → ADE20K | BeiT-L | 568M | 640x640 | 400 | 69.57 | 57.12 | 51.95 | 44.03 | 13.14 | 86.57 | 58.64 | 66.41 |
| IN22K+COCO → ADE20K | BeiT-L | 571M | 896x896 | 784 | 69.57 | 57.47 | 49.73 | 44.58 | 12.89 | 88.09 | 59.03 | 66.49 |
| IN22K+COCO → ADE20K | BeiTv2-L | 571M | 896x896 | 784 | **70.84** | **58.58** | 49.84 | 44.28 | **13.39** | **88.24** | **61.32** | **67.80** |
| IN22K → ADE20K | AugReg-T | 36M | 512x512 | 256 | 58.52 | 50.25 | 47.98 | 43.92 | 11.53 | 76.08 | 52.32 | 56.65 |
| IN22K → ADE20K | AugReg-B | 134M | 512x512 | 256 | 61.51 | 52.68 | 46.41 | 44.65 | 11.99 | 78.28 | 54.20 | 59.29 |
| IN22K → ADE20K | AugReg-L | 364M | 512x512 | 256 | 65.92 | 55.60 | 50.37 | 44.42 | 12.32 | 83.23 | 54.98 | 63.01 |
| IN1K → ADE20K | DeiT-S | 58M | 512x512 | 256 | 61.56 | 52.37 | **52.16** | 44.25 | 11.74 | 79.82 | 53.60 | 59.36 |
| IN1K → ADE20K | DeiT-B | 134M | 512x512 | 256 | 63.37 | 53.32 | 49.40 | 43.98 | 11.97 | 81.64 | 53.11 | 60.69 |

Table 36 | ViT-Adapter localization benchmarks

| Pretrained Dataset | Visual Encoder | Encoder Size | Image Size | Vision Token # | RefCOCO | RefCOCO+ | RefCOCOg | OCID-Ref | Weighted Loc. | Weighted Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| Multi-Modal → ADE20K | Perceiver | 364M | 512x512 | 256 | **63.31** | **51.55** | **55.88** | 20.81 | 42.29 | 60.14 |
| IN22K → ADE20K | BeiT-L | 451M | 640x640 | 400 | 52.99 | 42.75 | 47.81 | 16.94 | 35.22 | 61.61 |
| IN22K → ADE20K | BeiT-L | 568M | 640x640 | 400 | 56.09 | 46.49 | 51.45 | 18.60 | 37.94 | 62.58 |
| IN22K+COCO → ADE20K | BeiT-L | 571M | 896x896 | 784 | 55.53 | 46.07 | 49.43 | 18.51 | 37.45 | 62.59 |
| IN22K+COCO → ADE20K | BeiTv2-L | 571M | 896x896 | 784 | 60.34 | 49.29 | 53.33 | **24.69** | **42.34** | **64.38** |
| IN22K → ADE20K | AugReg-T | 36M | 512x512 | 256 | 41.19 | 29.14 | 34.17 | 11.32 | 25.31 | 52.44 |
| IN22K → ADE20K | AugReg-B | 134M | 512x512 | 256 | 48.32 | 36.36 | 41.44 | 15.54 | 31.29 | 55.53 |
| IN22K → ADE20K | AugReg-L | 364M | 512x512 | 256 | 56.58 | 45.16 | 50.47 | 20.28 | 38.32 | 59.69 |
| IN1K → ADE20K | DeiT-S | 58M | 512x512 | 256 | 49.52 | 37.07 | 42.46 | 15.35 | 31.78 | 55.65 |
| IN1K → ADE20K | DeiT-B | 134M | 512x512 | 256 | 52.17 | 39.78 | 45.61 | 16.22 | 33.77 | 57.07 |