ViTamin: Designing Scalable Vision Models in the Vision-Language Era

Jieneng Chen[1*] Qihang Yu[2*] Xiaohui Shen[2] Alan Yuille[1] Liang-Chieh Chen[2]
[1]Johns Hopkins University [2]ByteDance *equal contribution
https://beckschen.github.io/vitamin.html

Abstract

Recent breakthroughs in vision-language models (VLMs) start a new page in the vision community. The VLMs provide stronger and more generalizable feature embeddings compared to those from ImageNet-pretrained models, thanks to the training on the large-scale Internet image-text pairs. However, despite the amazing achievement from the VLMs, vanilla Vision Transformers (ViTs) remain the default choice for the image encoder. Although pure transformer proves its effectiveness in the text encoding area, it remains questionable whether it is also the case for image encoding, especially considering that various types of networks are proposed on the ImageNet benchmark, which, unfortunately, are rarely studied in VLMs. Due to small data/model scale, the original conclusions of model design on ImageNet can be limited and biased. In this paper, we aim at building an evaluation protocol of vision models in the vision-language era under the contrastive language-image pretraining (CLIP) framework. We provide a comprehensive way to benchmark different vision models, covering their zero-shot performance and scalability in both model and training data sizes. To this end, we introduce ViTamin, a new vision models tailored for VLMs. ViTamin-L significantly outperforms ViT-L by 2.0% ImageNet zero-shot accuracy, when using the same publicly available DataComp-1B dataset and the same OpenCLIP training scheme. ViTamin-L presents promising results on 60 diverse benchmarks, including classification, retrieval, open-vocabulary detection and segmentation, and large multi-modal models. When further scaling up the model size, our ViTamin-XL with only 436M parameters attains 82.9% ImageNet zero-shot accuracy, surpassing 82.0% achieved by EVA-E that has ten times more parameters (4.4B).

1. Introduction

The past decades have witnessed significant progress in computer vision, like visual recognition tasks. The advent of AlexNet [53] marked a significant milestone, catalyzing the extensive evolution and dominance of Convolutional Neural Networks (ConvNets) [8, 32, 37, 38, 46, 55, 72, 73] in computer vision. More recently, with the debut of Vision Transformer [23, 104], a growing number of transformer-based architectures [18, 71, 103, 108, 116, 121] have shown great potential to surpass the prior ConvNet counterparts.

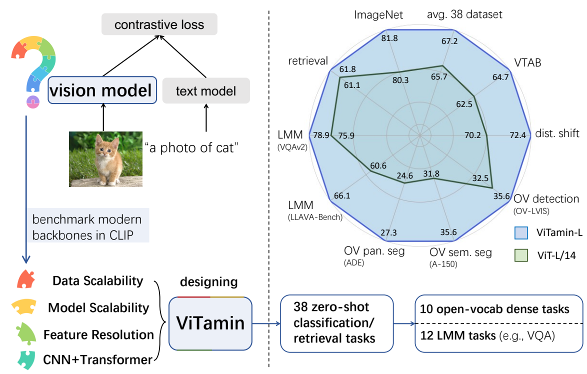

Figure 1. **Practices of designing scalable vision models in the vision-language era.** We benchmark modern vision models with various model and data scales under CLIP setting using DataComp-1B [30], leading to findings about data and model scalability, feature resolution, and hybrid architecture, which motivate us to develop ViTamin for VLM. ViTamin-L achieves superior zero-shot performance over ViT-L/14 [60] on ImageNet [86] and average 38 datasets [30], and advances a suite of 22 downstream tasks for Open-Vocabulary (OV) detection [111] and segmentation [124], and Large Multi-modal Model (LMM) tasks [67].

The rapid advancement of neural network design in computer vision can be attributed to a combination of factors. Among them, an important factor is the well-established benchmarks, allowing the community to examine the developments in a standardized way. Particularly, ImageNet [86] has become the de facto testing ground for new vision models. It not only sets a standard benchmark for visual recognition, but also serves as a mature pre-training dataset for transferring the network backbone to a variety of downstream tasks (*e.g.*, detection and segmentation) [9, 10, 15, 38, 51, 66, 73, 95, 107, 117, 122, 123].

Recently, the emergence of vision-language models (VLMs) [50, 82] has changed the paradigm by leveraging the pre-training schedule on the extremely large scale noisy

Internet data up to billions of image-text pairs [89], much larger than the ImageNet scale. VLMs not only produce strong and generalizable features [50, 82], but also excel in zero-shot downstream tasks [31, 54, 68, 77, 85, 124, 137]. However, unlike the ImageNet benchmark, where many types of neural networks are designed and blossomed [37, 44, 46, 53, 91, 97], the existing VLMs mostly employ the vanilla Vision Transformer (ViT) architecture [23] *, and the recent benchmark DataComp [30] focuses on the data curation under the common (yet unverified) belief that ViTs scale much better than any other architectures in this vision-language era [20, 61] and thus ViT is all we need.

The current trend can be characterized by several key observations: (1) The high computational demand, requiring extensive resources [48] for months, is a significant barrier for advancing VLMs [82], limiting exploring diverse vision models. (2) Traditional vision models are mainly optimized for the ImageNet benchmark, which may not scale well for larger datasets [30, 89], unlike purely transformer-based architectures [104] that have proven scalable in language tasks [81, 102] and are now being adopted for VLMs as image encoders [23]. (3) Current VLM benchmarks focus on zero-shot classification/retrieval tasks [30], with a notable lack of downstream tasks involving open-vocabulary dense prediction [21, 22, 31, 54, 114, 115, 124, 129, 135], as well as a gap in assessing Large Multi-modal Models (LMMs) [57, 67, 68, 137].

In this paper, we aim to address the aforementioned issues with practices as shown in Fig. 1. To begin with, we establish a new test bed for designing vision models under the CLIP framework [50, 82] using the DataComp-1B dataset [30], which is one of the largest publicly available datasets with high quality. Specifically, we employ two training protocols: *short schedule* for fast benchmarking vision models across model and data scales, and *long schedule* for training best performing vision models. With the *short schedule*, we re-benchmark state-of-the-art vision models found on ImageNet settings for VLMs. Particularly, we select ViT [23], ConvNeXt [72], CoAtNet [18], as representatives for pure Transformer, pure ConvNet, and hybrid architecture, respectively. We combine various model scales and data scales to provide a comprehensive evaluation towards different architectures, revealing several critical findings. First, increasing data scales improves all vision models across all model sizes, while ViT scale slightly better than others in terms of model parameters. Second, the final resolution of the extracted features affects prediction performance. Third, CoAtNet performs better than ViT and ConvNeXt in general, though it is hard to scale up CoAtNet-4 to billions of data due to computational constraints.

Those findings motivate us to develop a new vision model, named ViTamin tailored for VLM. ViTamin is a 3-stage hybrid architectures, combining two stages of MBConv blocks with a final stage of Transformer blocks. This hybrid design leverages its Transformer stage to enhance data and model scalability, along with output stride of 16 to enjoy high feature resolution. As a result, ViTamin-L outshines its ViT-L/14 counterpart [30] by +2.0% zero-shot imageNet accuracy in identical OpenCLIP training scheme and identical 256 token length. When increasing feature resolution to 576 patches, ViTamin-L further attains 81.8% zero-shot imageNet accuracy, surpassing the prior art ViT-L/14 CLIPA-v2 [60] by +1.5%. In average performance across 38 datasets, it not only exceeds ViT-L/14 counterpart [60] by +1.5%, but also outperforms the larger ViT-H/14 model [60] by +0.4% while having only half parameters. When further scaling up the model size, our ViTamin-XL with only 436M parameters attains 82.9% ImageNet zero-shot accuracy, surpassing 82.0% achieved by EVA-E (*i.e.*, EVA-02-CLIP-E/14 [94]) that has ten times more parameters (4.4B). Furthermore, we introduce an effective training scheme Locked-Text Tuning (LTT), which guides the training of vision backbone with a frozen pretrained text encoder. It enhances the small variant by +4.0% and the base variant by +4.9% without any extra cost.

Our another intriguing observation is the prevailing emphasis on data filtering over vision architecture design in VLM. For instance, while the best DataComp challenge solution [119] achieved only a +2.3% gain, our ViTamin with LTT largely improves performance by +23.3% on the same dataset size, without intensive data filtering. Finally, we introduce a suite of downstream tasks, including open-vocabulary detection and segmentation, and LMMs, for evaluating VLM-specific vision models. ViTamin outperforms the ViT-L model, enhancing detector by +3.1% on OV-LVIS and segmentor by +2.6% on average 8 datasets, and excelling across 12 LMM benchmarks. Notably, ViTamin sets a new state-of-the-art on 7 benchmarks for open-vocabulary segmentation.

We aim for our findings to encourage a reevaluation of the current limitations in VLM designs and hope that our extensive benchmarking and evaluations will drive the development of more advanced vision models for VLMs.

2. Related Work

Vision Backbone: On the ImageNet benchmark [86], ConvNets [37, 46, 53, 72, 88, 91, 96–98, 113, 128] have been the dominant networks choice since the advent of AlexNet [53]. Recently, the vision community has witnessed impressive emergence of the Transformer architecture [104], a trend that began with the widespread adoption of the ViT [23] and its subsequent developments [26, 58, 62, 71, 83, 99, 106, 108, 126, 136]. Among these, hybrid architectures [14, 18, 24, 34, 41, 63, 76, 93, 103, 109, 112, 116] combine Transformer self-attention with convo-

*with only a few exceptions, *e.g.*, ConvNeXt [72] by OpenCLIP [48].

lution, where CoAtNet [18] particularly obtains impressive results on ImageNet. Notably, MaX-DeepLab [107], emerged as early as 2020, successfully developed a hybrid network backbone for dense pixel predictions, where the first two stages utilize residual bottleneck blocks [37], followed by two subsequent stages employing axial attention [106]. More recently, by leveraging the design practices of a Vision Transformer, a ResNet [37] can be modernized to ConvNeXt [72], competing favorably with ViT. Along the same direction, but not limited to the ImageNet scale, our work aims to develop a novel vision model for training with billions of data [30] in the vision-language era.

Language-Image Pre-training: Language-image pre-training has seen significant advancements [1, 3, 50, 57, 68, 82, 120] with the emergence of LLMs [7, 81, 101]. The huge progress can be attributed to the pre-training on an immense scale of noisy web-collected image-text data [30, 89], much larger than the ImageNet. Notably, CLIP [50, 82] generates strong image features and excels in zero-shot transfer learning [31, 54, 68, 77, 85, 124, 137], which make it an essential role in large multi-modal model [13, 57, 67, 68]. CLIP has been improved by advanced training strategies including self-supervised learning [65, 79], efficient tuning [69, 132] and training [59, 61, 94, 110, 133]. These studies predominantly employ ViT [23] as the only vision model. As a result, the architectural design for the CLIP vision model has not been thoroughly investigated. Thus, we attempt to bridge the gap by developing a novel vision model for VLMs.

3. Method

In the section, we revisit the problem definition of CLIP and propose two training protocols (*short* and *long* schedules) on DataComp-1B (Sec. 3.1). With *short schedule*, we re-benchmark modern vision models found on ImageNet under the CLIP setting (Sec. 3.2). We then introduce the proposed ViTamin architecture design, motivated by the discoveries in the re-benchmarking results (Sec. 3.3).

3.1. CLIP and Training Protocols

CLIP Framework: Given a batch of N image-text pairs $\{(I_1, T_1), ..., (I_N, T_N)\}$ (where I_i and T_i denote image and text for i_{th} pair), the objective of CLIP [82] learns to align the image embeddings $\mathbf{x}_i$ and text embeddings $\mathbf{y}_i$ for each pair. Formally, the loss function is defined as follows:

$$-\frac{1}{2N}\sum_{i=1}^{N}(\underbrace{\log\frac{e^{\mathbf{x}_i^T\mathbf{y}_i/\tau}}{\sum_{j=1}^{N}e^{\mathbf{x}_i^T\mathbf{y}_j/\tau}}}_{\text{image to text}}+\underbrace{\log\frac{e^{\mathbf{y}_i^T\mathbf{x}_i/\tau}}{\sum_{j=1}^{N}e^{\mathbf{y}_i^T\mathbf{x}_j/\tau}}}_{\text{text to image}}), \quad (1)$$

where $\mathbf{x}_i = \frac{f(I_i)}{\|f(I_i)\|_2}$, $\mathbf{y}_i = \frac{g(T_i)}{\|g(T_i)\|_2}$, and τ is a temperature variable. A vision model $f(.)$ and a text model $g(.)$ are trained to minimize the loss function. We focus on vision model design and use the text models from OpenCLIP [48].

Training Protocols: We employ two training protocols: *short schedule* and *long schedule*. The *short schedule* is designed for efficiently benchmarking vision models up to 1 training epoch on DataComp-1B [30] (*i.e.*, 1.28B seen samples). As detailed in Tab. 2, given a descent amount of resources (*e.g.*, 32 A100 GPUs), it takes less than two days to train a small (∼25M parameters) model variant. The *long schedule* is designed for training the best performing models with up to 40B seen samples.

3.2. Benchmarking Vision Models in CLIP Setting

The *short schedule* allows us to efficiently re-benchmark state-of-the-art vision models found on ImageNet under the CLIP setting using DataComp-1B. The experimented models are ViT [23] (a pure Transformer), ConvNeXt [72] (a pure ConvNet), and CoAtNet [18] (a hybrid model). We examine their scalability in terms of both model scales and data sizes. Each vision model has sizes varying from small (∼25M parameters), base (∼85M) to large (∼300M), while the data sizes range from 128M, 512M to 1.28B training seen samples (1 epoch is equal to 1.28B seen samples). The metric is zero-shot accuracy on ImageNet, supplemented by the results on the 38 tasks following DataComp [30]. As shown in Fig. 2, we analyze the benchmarked results from four aspects, including data scalability, model scalability, feature resolution, and hybrid architecture. For simplicity, we use "X@Y" to denote the vision model X trained with Y seen samples. See appendix for numerical results.

Data Scalability: When training seen samples increase from 128M to 1.28B, we observe a consistent trend of improvements across all model sizes and all vision models (a1-a5). Interestingly, ViT-S/16@512M (22M parameter) attains the zero-shot performance of 53.8% on ImageNet, which is better than 45.8% by ViT-B/16@128M (86M parameter). It shows the effectiveness of training large scale data that quadrupling training seen samples can be more impactful than quadrupling the number of model parameters. Additionally, ViT-B/16@512M & @1.28B significantly boost ViT-B/16@128M from 45.8% to 60.0% (+14.2%) and 65.6% (+19.8).

→ *As the training seen samples increase, the performances consistently improves in all cases.*

Model Scalability: When the model sizes increase, the performances of all vision models are also boosted (b1-b3). However, we observe a different gain among them (b4). For example, ConvNeXt-XL@128M brings only +1.4% gain over ConvNeXt-B@128M, while ViT-L/16@128M enhances ViT-B/16@128M by +3.6%. Given plenty of data, ViT still shows a better model scalability, especially scaling from base to large (*e.g.*, +6.4% for ViT *vs.* +3.6% for both CoAtNet and ConvNeXt at 512M samples; +6.3% for ViT

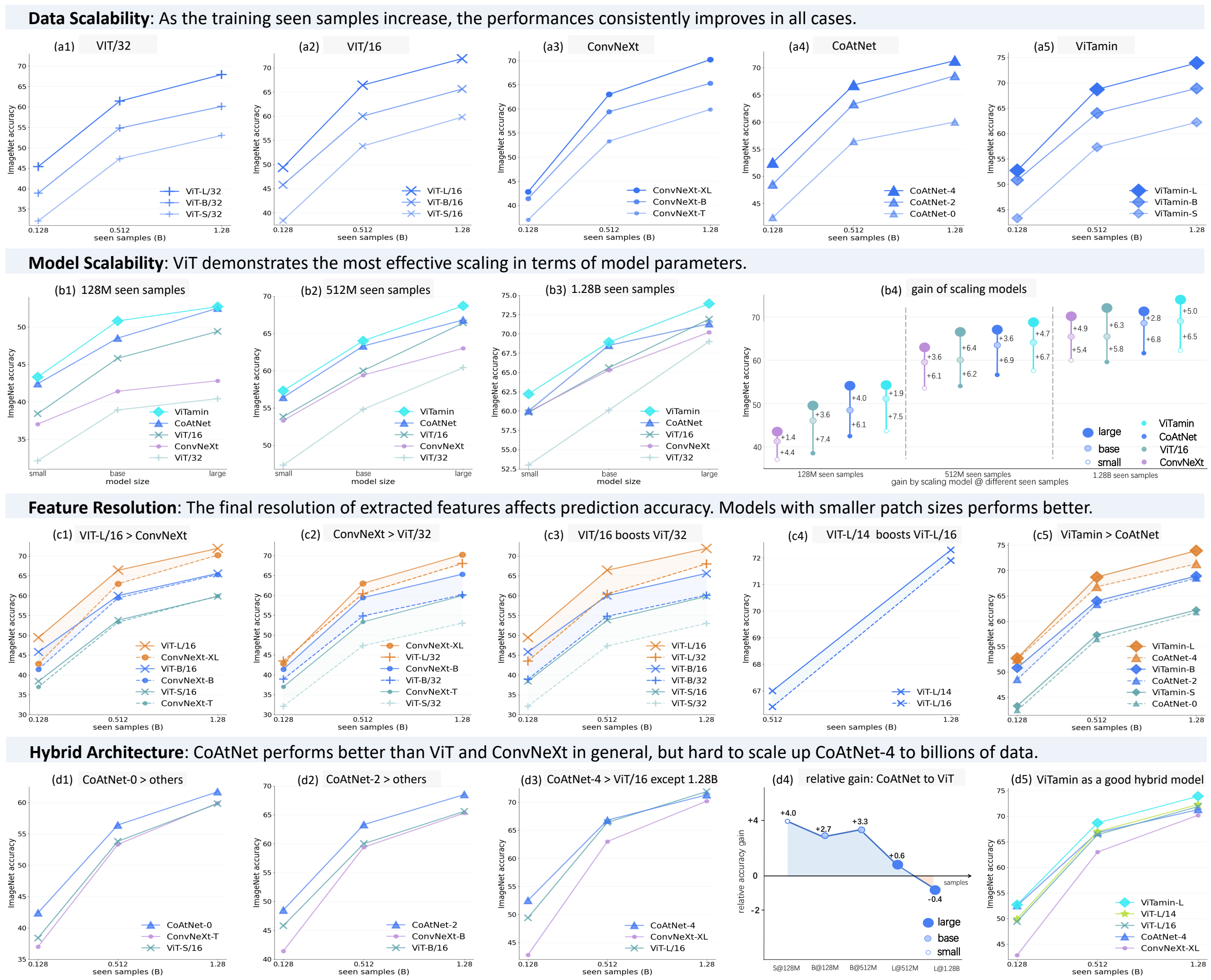

Figure 2. **Benchmarking vision models under CLIP setting on DataComp-1B**, including ViT (a pure Transformer), ConvNeXt (a pure ConvNet), and CoAtNet (a hybrid model). We examine their scalability in terms of both data sizes (1st row) and model scales (2nd row), and further analyze the results from the aspects of feature resolution (3rd row) and hybrid architecture (4th row).

vs. +2.8% for CoAtNet and + 4.9% for ConvNeXt at 1.28B samples). As a result, ViT shows the best scalability.

→ *ViT demonstrates the most effective scaling in terms of model parameters.*

Feature Resolution: Across all model scales and data sizes, ConvNeXt performs better than ViT/32 but loses its advantage to ViT/16 (c1 & c2). This trend deviates significantly from what is observed in ImageNet era, where ConvNeXt consistently outperforms ViT/16 (also see our ImageNet-scale VLM experiments in appendix). We hypothesize that, comparing to ImageNet's object class label, the text in CLIP captures broader area of information, and thus is beneficial from higher feature resolution. Besides, ViT also benefits from using smaller patch sizes (thus high feature resolution) over larger path sizes (c3 & c4).

→ *The final resolution of extracted features affects the prediction performance. ViT with smaller patch sizes outperforms ViT with larger patch sizes and ConvNeXt.*

Hybrid Architecture: We observe that ConvNeXt consistently lags behind ViT-{S,B}/16 and particularly ViT-L/14, suggesting that a pure ConvNet has limited capacity under the CLIP setting when presented with abundant of data (d1-d3). By contrast, CoAtNet significantly surpasses both ViT and ConvNeXt (*e.g.*, CoAtNet-2@1.28B has a remarkable +2.9% and +3.2% gain over ViT-B/16@1.28B and ConvNeXt-B@1.28B, respectively), indicating the effectiveness of hybrid models. However, CoAtNet requires the most GPU memory; we can only train CoAtNet-4 with batch size 8k on 64 A100 GPUs, while all the other large models are trained with batch size 16k on 32 A100 GPUs. This affects CoAtNet's scalability in large variant.

→ *CoAtNet surpasses ViT and ConvNeXt in general, yet it is hard to scale up CoAtNet-4 to billions of data.*

3.3. Novel Vision Transformer for Vision-Language

Herein, we distill from the aforementioned observations, culminating in the proposed vision model, ViTamin (**Vi**sion **T**r**A**nsfor**M**er for v**I**sion-la**N**guage), which notably takes the lead in the benchmarking results across all settings in Fig. 2. To introduce ViTamin, we commence by its macro-level network design (Sec. 3.3.1), followed by the micro-level block design (Sec. 3.3.2). Finally, we develop a vision model family with a simple scaling rule (Sec. 3.3.3).

3.3.1 Macro-level Network Design

Overview: The macro-level network design of ViTamin is inspired by the ViT and CoAtNet. Specifically, on top of a simple convolutional stem (*i.e.*, two 3×3 convolutions) [18], we adopt a 3-stage network architecture, where the first two stages employ the Mobile Convolution Blocks (MBConv) [43, 88] and the third stage uses the Transformer Blocks (TFB) [23, 104]. Fig. 3 shows the overview of ViTamin. We detail the design principles below, based on the discoveries from the re-benchmarking results

Data and Model Scalability: ViT demonstrates the best scalability in terms of both model scales and data sizes. We thus opt for using Transformer Blocks in our last stage, and we stack most blocks here across different model sizes.

Feature Resolution: We tailor the network to generate high resolution feature maps in the end. Our 3-stage network design thus yields a feature map with output stride 16 (*i.e.*, a downsampling factor of 16).

Hybrid Architecture: Similar to CoAtNet, we employ MBConv in the first two stages, resulting in a hybrid model. However, unlike CoAtNet that is constrained by its large memory usage, we propose a light-weight design of stage 1 and 2, which contain only two and four MBConv blocks.

Given the macro-level network design, we then move on to further improve the micro-level block design below.

3.3.2 Micro-level Block Design

Overview: The proposed ViTamin depends on two types of blocks: Mobile Convolution Blocks (MBConv) and Transformer Blocks (TFB). We refine each block in our model.

MBConv-LN: The Mobile Convolution Block (MBConv) [88] employs the "inverted bottleneck" design [37], starting with a first 1×1 convolution to expand the channel size, followed by a 3×3 depthwise convolution [44] for spatial interaction, and ending with another 1×1 convolution to revert to the original channel size. Modern MBConv, as in MobileNetv3 [43], adds numerous batch normalization (BN) [49] layers and squeeze-and-excitation (SE) [45]. We adopt a simple modification by removing all BN layers and SE, and just using a single layer normalization (LN) [4] as the first layer in our block, akin to the pre-norm layer in the Transformer block, resulting in the proposed MBConv-LN. Ablation (in appendix) shows that MBConv-LN enjoys a simple design while attaining a similar performance to the original MBConv-BN-SE in MobileNetv3.

block	stride	ViTamin-S B	ViTamin-S C	ViTamin-B B	ViTamin-B C	ViTamin-L B	ViTamin-L C	ViTamin-XL B	ViTamin-XL C
conv-stem	2	2	64	2	128	2	160	2	192
MBConv-LN	4	2	64	2	128	2	160	2	192
MBConv-LN	8	4	128	4	256	4	320	4	384
TFB-GeGLU	16	14	384	14	768	31	1024	32	1152

Table 1. **ViTamin model variants.** ViTamin variants differ in the number of blocks B and number of channels C in each stage.

TFB-GeGLU: The Transformer Block (TFB) [104] contains two residual blocks: one with self-attention and the other with feed-forward network (FFN). We empirically discover that substituting the first linear layer with GeGLU [90], an enhanced version of the Gated Linear Unit [19] that has a $2\times$ expansion rate, can enhance accuracy in the FFN. We denote the Transformer Block with the updated FFN as TFB-GeGLU. Ablation (in appendix) shows that TFB-GeGLU requires 12% fewer parameters than TFB due to half expansion ratio, allowing us to stack additional Transformer blocks towards deeper architectures [96, 100, 136].

3.3.3 Meta Architecture and Scaling Rule

Meta Architecture: After introducing our macro-level network and micro-level block designs, we now put everything together to form the meta architecture of ViTamin. Specifically, ViTamin is a hybrid architecture that contains only three stages, built on top of a simple convolutional stem (*i.e.*, two 3×3 convolutions). The first two stages are composed of MBConv-LN, where we stack two and four of them for stage 1 and 2, respectively. The third stage are obtained by stacking N_B TFB-GeGLU blocks. With the meta architecture in mind, we are ready to discuss the scaling rule to generate a family of ViTamin with different model sizes.

Scaling Rule: Our scaling rule is extremely simple and straightforward, controlled by two hyper-parameters: width (*i.e.*, the channel sizes of those three stages) and depth (*i.e.*, N_B, the number of TFB-GeGLU blocks in stage 3). Note that our convolutional stem has the same channel size as the first stage. We define four model sizes: Small, Base, Large, and X-Large (S, B, and L variants have a similar amount of model parameters to ViT [23, 131]). We use the same channel size as ViT in our 3rd stage for each model variant. Specifically, we set the channel sizes of our three stages as $(C, 2C, 6C)$, where $6C = \{384, 768, 1024, 1152\}$ for Small, Base, Large and X-Large model variant, respectively[†]. Subsequently, given the target model parameter, the value of N_B (*i.e.*, the number of TFB-GeGLU blocks in stage 3) can be easily found. We show the family of ViTamin-{S, B, L, XL} in Tab. 1.

[†]We calculate the channel size for stage 1 as $1/6C$, rounding to the nearest value that is divisible by 32.

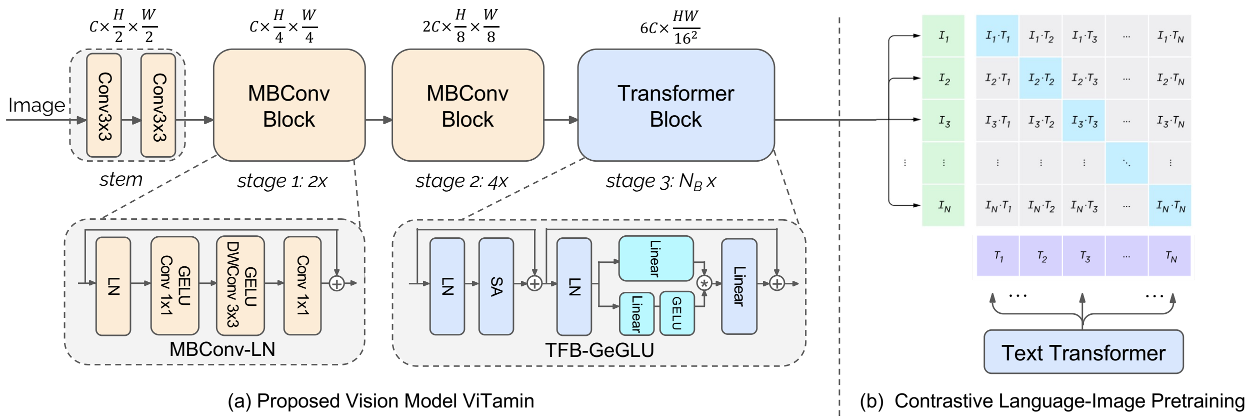

Figure 3. **Overview of ViTamin architecture**. (a) ViTamin begins with a convolutional stem, followed by Mobile Convolution Blocks (MBConv) in stage 1 and 2, and Transformer Blocks (TFB) in stage 3. The 2D input to the stage 3 is flatten to 1D. For the *macro-level* designs, the three-stage layout generates the final feature map with output stride 16, similar to ViT/16 [23]. We set channels sizes for the three stages to be (C, $2C$, $6C$). For the *micro-level* designs, the employed MBConv-LN modifies MBConv [88] by using a single LayerNorm [4]. TFB-GeGLU upgrades TFB's FFNs [104] (Feed-Forward Networks) with GELU Gated Linear Units [90]. (b) In the CLIP framework, given N image-text pairs, the vision model's output I_i is learned to align with its corresponding text Transformer's output T_i. Our text Transformers are the same as OpenCLIP [48]. +: Addition. *: Multiplication.

	short schedule for benchmarking			**long schedule**		
	ViTamin-S	ViTamin-B	ViTamin-L	ViTamin-L	ViTamin-XL	
batch size	8k	8k	16k	90k	90k	90k
image size	224	224	224	224	256	256
# A100 GPUs	32	32	32	184	312	312
# epochs	1	1	1	10	10	30
seen samples (B)	1.28	1.28	1.28	12.8	12.8	40.0
training days	1.8	3.3	5.6	11	15	46

Table 2. **Short and long training schedules on DataComp-1B.**

Locked-Text Tuning for CLIP: Besides model design, we propose Locked-Text Tuning (LTT) to exploits a pretrained frozen text encoder. In light of the aligned image and text embeddings in CLIP, we leverage the pretrained text encoder from a large VLM to guide the training of image encoders of smaller VLMs. Specifically, when training other ViTamin variants (*e.g.*, ViTamin-S and ViTamin-B), we initialize their text encoder with the one from a pretrained ViTamin-L. The text encoder is then frozen, used as a teacher to guide the training of the randomly initialized image encoder. This scheme can be considered as a way to distill the knowledge [42] from a pretrained frozen text encoder to a randomly initialized image encoder.

4. Experimental Results

In this section, we detail the implementation in Sec. 4.1, compare with the state-of-the-arts in Sec. 4.2, and deploy ViTamin to downstream tasks, including open-vocabulary detection/segmentation, and large multi-modal models in Sec. 4.3. See appendix for ablation studies.

4.1. Implementation Details

Training Strategy: We train the VLMs using OpenCLIP [48] on the public dataset DataComp-1B [30]. Tab. 2 summarizes the settings for our training schedules and model variants. We use the short schedule to benchmark vision models and conduct our ablation studies, and long schedule to train our best ViTamin-L. We closely follow the training hyper-parameter settings in OpenCLIP [30, 48]. The training and fine-tuning details are in the appendix.

Evaluation Strategy: We follow DataComp [30] to zero-shot evaluate VLMs with a testbed of 38 tasks, including ImageNet [86], 6 distribution shift tasks [5, 39, 40, 84, 105], VTAB tasks [130], WILDS tasks [52, 87], and 3 retrieval tasks [6, 11, 118].

Other Downstream Tasks: We evaluate the trained VLM in downstream tasks. For open-vocabulary detection, we exploit the F-ViT framework [111], while for open-vocabulary segmentation, we adopt the FC-CLIP framework [124] and zero-shot evaluate on multiple segmentation datasets. Finally, we evaluate VLMs in LLaVA-1.5 [67] for LMMs across multiple benchmarks. In all the cases, F-ViT, FC-CLIP, and LLaVA employ the frozen VLM backbone to effectively ablate different pretrained VLMs.

4.2. Main Results

Comparison with other State-of-the-arts: Tab. 3 summarizes the comparison between ViTamin-L and other state-of-the-art models, which exclusively employ the ViT backbone [23] but use different training schemes and datasets. For a fair comparison, we focus on the methods that use the same training data DataComp-1B [30], but still list other methods in the table for reference. For simplicity, we use "X@Z" to denote the vision model X trained with input size Z [‡]. ImageNet zero-shot accuracy is our main metric; other results are still reported in the table. As shown

[‡]Notation @ here is slightly abused to denote the training seen samples.

image encoder	image size	num patches	text encoder depth/width	seen samples (B)	training scheme	training dataset	trainable params Image+Text (M)	MACs Image+Text (G)	ImageNet Acc.	avg. 38 datasets	ImageNet dist. shift.	VTAB	retrieval
ViT-L/14 [30]	224	256	12 / 768	12.8	OpenCLIP	DataComp-1B	304.0 + 123.7	77.8 + 6.6	79.2	66.3	67.9	65.2	60.8
ViT-L/14 [60]	224	256	12 / 768	12.8 + 0.5	CLIPA-v2	DataComp-1B	304.0 + 110.3	77.8 + 2.7	79.7	65.4	68.6	62.9	60.6
ViT-L/14 [60]	336	576	12 / 768	12.8 + 0.5 + 0.1	CLIPA-v2	DataComp-1B	304.3 + 110.3	174.7 + 2.7	80.3	65.7	70.2	62.5	61.1
ViTamin-L	224	196	12 / 768	12.8	OpenCLIP	DataComp-1B	333.3 + 123.7	72.6 + 6.6	80.8	66.7	69.8	65.3	60.3
ViTamin-L	256†	256	12 / 768	12.8 + 0.2	OpenCLIP	DataComp-1B	333.4 + 123.7	94.8 + 6.6	81.2	67.0	71.1	65.3	61.2
ViTamin-L	336	441	12 / 768	12.8 + 0.2	OpenCLIP	DataComp-1B	333.6 + 123.7	163.4 + 6.6	81.6	67.0	72.1	64.4	61.6
ViTamin-L	384†	576	12 / 768	12.8 + 0.2	OpenCLIP	DataComp-1B	333.7 + 123.7	213.4 + 6.6	81.8	67.2	72.4	64.7	61.8
ViTamin-L2	224	196	24 / 1024	12.8	OpenCLIP	DataComp-1B	333.6 + 354.0	72.6 + 23.3	80.9	66.4	70.6	63.4	61.5
ViTamin-L2	256†	256	24 / 1024	12.8 + 0.5	OpenCLIP	DataComp-1B	333.6 + 354.0	94.8 + 23.3	81.5	67.4	71.9	64.1	63.1
ViTamin-L2	336	441	24 / 1024	12.8 + 0.5	OpenCLIP	DataComp-1B	333.8 + 354.0	163.4 + 23.3	81.8	67.8	73.0	64.5	63.6
ViTamin-L2	384†	576	24 / 1024	12.8 + 0.5	OpenCLIP	DataComp-1B	334.0 + 354.0	213.4 + 23.3	82.1	68.1	73.4	64.8	63.7
ViTamin-XL	256†	256	27 / 1152	12.8 + 0.5	OpenCLIP	DataComp-1B	436.1 + 488.7	125.3 + 33.1	82.1	67.6	72.3	65.4	62.7
ViTamin-XL	384†	576	27 / 1152	12.8 + 0.5	OpenCLIP	DataComp-1B	436.1 + 488.7	281.9 + 33.1	82.6	68.1	73.6	**65.6**	**63.8**
ViTamin-XL	256†	256	27 / 1152	40.0	OpenCLIP	DataComp-1B	436.1 + 488.7	125.3 + 33.1	82.3	67.5	72.8	64.0	62.1
ViTamin-XL	336†	441	27 / 1152	40.0 + 1.0	OpenCLIP	DataComp-1B	436.1 + 488.7	215.9 + 33.1	82.7	68.0	73.9	64.1	62.6
ViTamin-XL	384†	576	27 / 1152	40.0 + 1.0	OpenCLIP	DataComp-1B	436.1 + 488.7	281.9 + 33.1	**82.9**	**68.1**	**74.1**	64.0	62.5
ViT-L/14 [94]	224	256	12 / 768	4.0	EVA-CLIP	Merged-2B	333.3 + 123.7	72.6 + 6.6	79.8	64.9	68.9	62.8	63.3
ViT-L/14 [94]	336	576	12 / 768	6.0	EVA-CLIP	Merged-2B	333.3 + 123.7	72.6 + 6.6	80.4	65.8	70.9	63.2	63.5
ViT-L/16 [133]	256	256	24 / 1024	40.0	SigLIP	WebLI	316.0 + 336.2	78.1 + 19.3	80.5	65.6	70.2	62.5	61.1
ViT-L/16 [133]	384	576	24 / 1024	40.0 + 5.0	SigLIP	WebLI	316.3 + 336.2	175.8 + 19.3	82.1	66.8	70.9	63.1	68.7
ViT-G/14 [48]	224	256	32 / 1280	39.0	OpenCLIP	LAION-2B	1844.9 + 694.7	473.4 + 48.5	80.1	66.7	69.1	64.6	63.5
ViT-H/14 [60]	336	576	24 / 1024	12.8 + 0.5 + 0.1	CLIPA-v2	DataComp-1B	632.5 + 354.0	363.7 + 9.7	81.8	66.8	72.4	63.7	62.6
ViT-E/14 [94]	224	256	24 / 1024	4.0	EVA-CLIP	LAION-2B	4350.6 + 354.0	1117.3 + 23.3	82.0	66.9	72.0	63.6	62.8
ViT-G/14 [60]	336	576	32 / 1280	12.8 + 0.5 + 0.1	CLIPA-v2	DataComp-1B	1845.4 + 672.3	1062.9 + 20.2	83.1	68.4	74.0	64.5	63.1
SoViT-400M/14 [2]	224	256	27 / 1152	40.0	SigLIP	WebLI	428.2 + 449.7	106.2 + 6.6	82.0	68.1	69.5	64.8	66.8
SoViT-400M/14 [2]	384	729	27 / 1152	40.0 + 5.0	SigLIP	WebLI	428.2 + 449.7	302.3 + 26.3	83.1	69.2	72.4	64.6	69.8
ViT-H/14 [27]	224	256	24 / 1024	39.0	OpenCLIP	DFN-5B	632.1 + 354.0	162.0 + 23.3	83.4	69.6	69.9	67.5	68.3
ViT-H/14 [27]	378	729	24 / 1024	39.0 + 5.0	OpenCLIP	DFN-5B	632.7 + 354.0	460.1 + 23.3	84.4	70.8	72.8	68.5	69.5

Table 3. **Comparison with state-of-the-art models.** Our models are only trained on the publicly available DataComp-1B [30]. CLIPA-v2 [60] uses an advanced progressive training scheme (from smaller images to larger ones) than the original OpenCLIP [30, 48] scheme that we follow. Other methods that use different settings are marked in gray for reference. Specifically, EVA-CLIP [94] uses EVA weights [28], better training scheme FLIP [65], and different training datasets [28, 89]. SigLIP [133] employs better sigmoid loss, stronger text encoders, and an extremely long schedule on the proprietary WebLI dataset [12] (40B for training and another 5B seen samples for fine-tuning). †: ViT-L/14 benefits from more image tokens by using a smaller output stride 14 than 16 that we use. To have the same image tokens, we slightly enlarge the image size (*e.g.*, $224/14 = 256/16$ and $336/14 = 384/16$). We note that all compared results are from the OpenCLIP-results that are evaluated under the same setting to ensure a fair comparison.

in the table, ViTamin-L@224 outperforms ViT-L/14@224 OpenCLIP [48] by +1.6%. However, ViT-L/14 benefits from more image tokens by using a smaller output stride 14 than 16 that we use (as benchmarked in the appendix). To have the same image tokens, we slightly enlarge the image size. As a result, our ViTamin-L@256 surpasses ViT-L/14@224 OpenCLIP [48] and CLIPA-v2 [60] by 2.0% and 1.5%, respectively. After fine-tuning on larger input sizes, ViTamin-L@384 and ViTamin-L@336 still performs better than ViT-L/14@336 CLIPA-v2 [60] by +1.5% and +1.3%, respectively. Impressively, with only half the parameters, our ViTamin-L attains an average of 67.2% performance across 38 datasets, exceeding the larger ViT-H/14 CLIPA-v2 model's performance by +0.4%. Scaling up the text encoder to match the model size of the image encoder (specifically, ViTamin-L2) notably increases zero-shot ImageNet accuracy to 82.1% and average 38 datasets performance to 68.1%. Further scaling up the model parameters (*i.e.*, ViTamin-XL) and 40 billion seen samples reaches 82.9% zero-shot ImageNet accuracy.

Locked-Text Tuning: Fig. 4 shows that our LTT improves our ViTamin-S/-B by a large margin, especially when data sizes are small. Notably, LTT lifts ViTamin-B to the next scale of model performance, surpassing ViT-L/16 by +14% in 128M samples and +1.1% in 512M seen samples. Interestingly, LTT can save 10% training budget for ViTamin-B as the text tower is completely frozen.

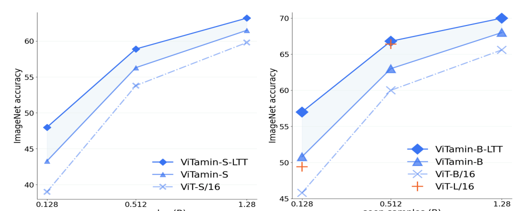

Figure 4. **Locked-text tuning (LTT)**. LTT exploits a pretrained frozen text encoder, and effectively boosts the model performance.

Data Quality *vs.* Model Capacity: The DataComp challenge [30] underscores the role of data filtering for VLM, however, using a fixed ViT model. As shown in Tab. 4, the leading solution [119] of DataComp challenge in ICCV 2023 employed a complicated 24 filtering rules to improve the dataset quality, resulting in +2.3% gain. Surprisingly, our ViTamin-B improves the performance by a healthy margin of +12.8% accuracy, and locked-text tuning can lift the gain to +23.3%. The result highlights the importance to co-design the vision-language dataset and model.

4.3. New Suite of Downstream Tasks

The evaluations so far are mostly on classification/retrieval-based task, highlighting a lack of downstream tasks similar to those employed in the ImageNet era. Yet, in contrast to ImageNet-based vision models where downstream tasks mainly involve transfer learning for conventional de-

image encoder	data filtering	dataset size	seen samp.	IN acc. (%)	avg. 38 datasets (%)
leaderboard					
ViT-B/32	DataComp [30]	14M	128M	29.7	32.8
ViT-B/32	SIEVE [75]	24M	128M	30.3 (+0.6)	35.3 (+2.5)
ViT-B/32	Top-1 Solution [119]	23M	128M	32.0 (+2.3)	37.1 (+4.3)
our experiments					
ViT-B/32	DataComp [30]	14M	128M	29.4	31.5
ViT-B/16	DataComp [30]	14M	128M	35.8 (+6.4)	34.6 (+3.1)
ViTamin-B	DataComp [30]	14M	128M	42.2 (+12.8)	38.3 (+6.8)
ViT-B/16-LTT	DataComp [30]	14M	128M	43.6 (+14.2)	41.1 (+9.6)
ViTamin-B-LTT	DataComp [30]	14M	128M	52.7 (+23.3)	47.2 (+15.7)

Table 4. **Data quality *vs.* model capacity.** The leaderboard results are from ICCV 2023 DataComp challenge medium filtering track.

image encoder	pretraining dataset	pretraining scheme	OV-COCO [129] (AP_{50}^{novel})	OV-LVIS [35] (AP_r)
ViT-L/14	DataComp-1B	CLIPA-v2	36.1	32.5
ConvNeXt-L	LAION-2B	OpenCLIP	36.4	29.1
ViTamin-L	DataComp-1B	OpenCLIP	37.5	35.6

Table 5. **Open-vocabulary detection.** Different image encoders (ViT-L/14 by [60], ConvNeXt-L by [48]) are using the F-ViT framework [111] in a sliding window manner [125], trained on OV-COCO [129] and OV-LVIS [35]. ConvNeXt-L is marked in gray due to different pretrained dataset.

image encoder	pretraining dataset	pretraining scheme	panoptic dataset (PQ) ADE [134]	Cityscapes [17]	MV [80]	semantic dataset (mIoU) A-150 [134]	A-847 [134]	PC-459 [78]	PC-59 [78]	PAS-21 [25]
ViT-L/14	DataComp-1B	CLIPA-v2	24.6	40.7	16.5	31.8	14.3	18.3	55.1	81.5
ConvNeXt-L	LAION-2B	OpenCLIP	26.8	**44.0**	**18.3**	34.1	14.8	18.2	**58.4**	81.8
ViTamin-L	DataComp-1B	OpenCLIP	27.3	44.0	18.2	35.6	16.1	20.4	58.4	83.4

Table 6. **Open-vocabulary segmentation.** Different image encoders (ViT-L/14 by [60], ConvNeXt-L by [48]) are using the FC-CLIP framework [124] in a sliding window manner [125], trained on COCO [66] and zero-shot evaluated on the other datasets. ConvNeXt-L is marked in gray due to different pretrained dataset.

tection and segmentation, VLMs excel with zero-shot capability and provides feature embeddings that are well-aligned across the vision-language domain. In light of this, we introduce a novel suite of downstream tasks aimed at the holistic evaluation of VLMs, including open-vocabulary detection and segmentation and multi-modal LLM.

Open-Vocabulary Detection and Segmentation: To examine how well the trained VLMs can adapt to downstream tasks, we consider two simple yet effective frameworks F-ViT [111] and FC-CLIP [124] which utilize a frozen CLIP backbone for open-vocabulary detection and segmentation, respectively. Specifically, we consider different VLMs as plug-in frozen backbones to these frameworks, while for ViT and ViTamin that may not easily generalize to high resolution input, we extract the feature in a sliding window manner [125], with window size equal to the pretrain image size, resulting in Sliding F-ViT and Sliding FC-CLIP, respectively. Tab. 5 illustrates that ViTamin-L serves as a stronger image encoder for open-vocabulary detection, surpassing its ViT-L/14 counterpart by 1.4% and 3.1% on OV-COCO and OV-LVIS. Tab. 6 shows that ViTamin-L outperforms ViT-L/14 by 2.6% on average 3 panoptic datasets and by 2.6% on average 5 semantic datasets. Notably, surpassing prior art, ViTamin-L sets a new state-of-the-art performance across seven benchmarks for open-vocabulary panoptic segmentation and semantic segmentation.

image encoder	training scheme	VQAv2 [33]	GQA [47]	VizWiz [36]	SQA [74]	T-VQA [92]	POPE [29]	MME [64]	MMBench [70]	MMB^{CN} [70]	SEED [56]	$LLaVA^{W}$ [68]	MM-Vet [127]
ViT-L/14	OpenAI	78.5	**62.0**	50.0	66.8	58.2	**85.9**	**1511**	64.3	58.3	**58.6**	65.4	31.1
ViT-L/14	CLIPA-v2	75.9	60.3	48.8	65.6	55.0	84.9	1396	60.8	54.6	54.6	60.6	28.6
ViTamin-L	OpenCLIP	78.4	61.6	51.1	66.9	58.7	84.6	1421	**65.4**	**58.4**	57.7	64.5	**33.6**
ViTamin-L†	OpenCLIP	**78.9**	61.6	**55.4**	**67.6**	**59.8**	85.5	1447	64.5	58.3	57.9	**66.1**	**33.6**

Table 7. **Large Multi-modal Model (LMM) performance with different VLMs.** The results in 1st row originate from LLaVa-1.5 paper [67] and are marked in gray due to pretraining on OpenAI WIT dataset [82] unlike DataComp-1B [30] used by other rows. All listed models are trsained following the same settings in LLaVA-1.5 [67] with Vicuna-V1.5-7B [16], for a fair comparison. †: image size of 384 rather than the default 336.

Large Multi-modal Models: Another key application of VLMs lies in their role as vision encoders within LMMs [57, 68, 137], as image features in VLMs that is well-aligned with text, thereby bridging the visual comprehension gap for LLMs. Specifically, we consider LLaVA-1.5 [67] as the evaluated framework. We follow [67] for all experimental settings, where the image is processed through a frozen CLIP model and a MLP projector, retaining the image as visual tokens, which are prepended to a text sequence and fed into a frozen Vicuna-v1.5-7B [16]. We run evaluation on 12 LMM benchmarks following [67], with results in Tab. 7. It should be noted that while OpenAI-trained ViT-L/14 underperforms CLIPAv2-trained counterpart by -3.7% ImageNet accuracy, it excels remarkably in LLaVA (+4.4% on VQAv2 and +4.3% on VizWiz). This highlights the need for incorporating a variety of downstream tasks to ensure a comprehensive evaluation. Surprisingly, simply replacing LLaVA's image encoder to ViTamin-L can achieve new state-of-the-art across various benchmarks.

5. Conclusion

In this work, we build an evaluation protocols of modern vision models in VLM and re-benchmark them under CLIP setting. We examine vision models from four aspects of data scalability, model scalability, feature resolution and hybrid architecture. The four pillars motivate us to propose ViTamin, which not only competes favorably with ViT in zero-shot ImageNet accuracy and average 38 dataset accuracy, but also achieves the state-of-the-art on 22 downstream tasks covering open-vocaburary detection and segmentation and large multi-modal models. We hope that our design practices will drive the development of more advanced vision models for VLMs.

Acknowledgement: We thank Haichao Yu and Zhanpeng Zeng for the discussions about DataComp challenge and micro-level block design, respectively. This work was supported in part by ONR N00014-23-1-2641.

References

[1] Gpt-4v(ision) system card. 2023. 3

[2] Ibrahim Alabdulmohsin, Xiaohua Zhai, Alexander Kolesnikov, and Lucas Beyer. Getting vit in shape: Scaling laws for compute-optimal model design. *NeurIPS*, 2023. 7

[3] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *NeurIPS*, 2022. 3

[4] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 5, 6

[5] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *NeurIPS*, 2019. 6

[6] Yonatan Bitton, Nitzan Bitton Guetta, Ron Yosef, Yuval Elovici, Mohit Bansal, Gabriel Stanovsky, and Roy Schwartz. Winogavil: Gamified association benchmark to challenge vision-and-language models. *NeurIPS*, 2022. 6

[7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 2020. 3

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 1

[9] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 40(4):834–848, 2017. 1

[10] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 1

[11] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 6

[12] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022. 7

[13] Xi Chen, Xiao Wang, Lucas Beyer, Alexander Kolesnikov, Jialin Wu, Paul Voigtlaender, Basil Mustafa, Sebastian Goodman, Ibrahim Alabdulmohsin, Piotr Padlewski, et al. Pali-3 vision language models: Smaller, faster, stronger. *arXiv preprint arXiv:2310.09199*, 2023. 3

[14] Yinpeng Chen, Xiyang Dai, Dongdong Chen, Mengchen Liu, Xiaoyi Dong, Lu Yuan, and Zicheng Liu. Mobile-former: Bridging mobilenet and transformer. In *CVPR*, 2022. 2

[15] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-DeepLab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *CVPR*, 2020. 1

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

[17] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 8

[18] Zihang Dai, Hanxiao Liu, Quoc V Le, and Mingxing Tan. Coatnet: Marrying convolution and attention for all data sizes. *NeurIPS*, 2021. 1, 2, 3, 5

[19] Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *ICML*, 2017. 5

[20] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *ICML*, 2023. 2

[21] Xueqing Deng, Qihang Yu, Peng Wang, Xiaohui Shen, and Liang-Chieh Chen. Coconut: Modernizing coco segmentation. In *CVPR*, 2024. 2

[22] Zheng Ding, Jieke Wang, and Zhuowen Tu. Open-vocabulary universal image segmentation with maskclip. In *ICML*, 2023. 2

[23] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 1, 2, 3, 5, 6

[24] Stéphane d'Ascoli, Hugo Touvron, Matthew L Leavitt, Ari S Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *ICML*, 2021. 2

[25] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88:303–338, 2010. 8

[26] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *ICCV*, 2021. 2

[27] Alex Fang, Albin Madappally Jose, Amit Jain, Ludwig Schmidt, Alexander Toshev, and Vaishaal Shankar. Data filtering networks. *arXiv preprint arXiv:2309.17425*, 2023. 7

[28] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *CVPR*, 2023. 7

[29] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui

Yang, Xiawu Zheng, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 8

[30] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *arXiv preprint arXiv:2304.14108*, 2023. 1, 2, 3, 6, 7, 8

[31] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *ECCV*, 2022. 2, 3

[32] Ross Girshick. Fast r-cnn. In *CVPR*, 2015. 1

[33] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, 2017. 8

[34] Benjamin Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, and Matthijs Douze. Levit: a vision transformer in convnet's clothing for faster inference. In *CVPR*, 2021. 2

[35] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *ICLR*, 2022. 8

[36] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *CVPR*, 2018. 8

[37] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 2, 3, 5

[38] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *CVPR*, 2017. 1

[39] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, 2021. 6

[40] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *CVPR*, 2021. 6

[41] Byeongho Heo, Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Junsuk Choe, and Seong Joon Oh. Rethinking spatial dimensions of vision transformers. In *ICCV*, 2021. 2

[42] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 6

[43] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *ICCV*, 2019. 5

[44] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 2, 5

[45] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 5

[46] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 1, 2

[47] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019. 8

[48] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. 2, 3, 6, 7, 8

[49] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 5

[50] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 1, 2, 3

[51] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, 2019. 1

[52] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *ICML*, 2021. 6

[53] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *NeurIPS*, 2012. 1, 2

[54] Weicheng Kuo, Yin Cui, Xiuye Gu, AJ Piergiovanni, and Anelia Angelova. F-vlm: Open-vocabulary object detection upon frozen vision and language models. In *ICLR*, 2023. 2, 3

[55] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 1

[56] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023. 8

[57] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 2, 3, 8

[58] Kunchang Li, Yali Wang, Junhao Zhang, Peng Gao, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. Uniformer: Unifying convolution and self-attention for visual recognition. *TPAMI*, 2023. 2

[59] Runze Li, Dahun Kim, Bir Bhanu, and Weicheng Kuo. Reclip: Resource-efficient clip by training with small images. *arXiv preprint arXiv:2304.06028*, 2023. 3

[60] Xianhang Li, Zeyu Wang, and Cihang Xie. Scaling clip training with 81.1% zero-shot imagenet accuracy within a \$10,000 budget; an extra \$4,000 unlocks 81.8% accuracy. *arXiv preprint arXiv:2306.15658*, 2023. 1, 2, 7, 8

[61] Xianhang Li, Zeyu Wang, and Cihang Xie. An inverse scaling law for clip training. *NeurIPS*, 2023. 2, 3

[62] Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Mvitv2: Improved multiscale vision transformers for classification and detection. In *CVPR*, 2022. 2

[63] Yanyu Li, Geng Yuan, Yang Wen, Ju Hu, Georgios Evangelidis, Sergey Tulyakov, Yanzhi Wang, and Jian Ren. Efficientformer: Vision transformers at mobilenet speed. *NeurIPS*, 2022. 2

[64] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023. 8

[65] Yanghao Li, Haoqi Fan, Ronghang Hu, Christoph Feichtenhofer, and Kaiming He. Scaling language-image pre-training via masking. In *CVPR*, 2023. 3, 7

[66] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1, 8

[67] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 1, 2, 3, 6, 8

[68] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 2023. 2, 3, 8

[69] Haotian Liu, Kilho Son, Jianwei Yang, Ce Liu, Jianfeng Gao, Yong Jae Lee, and Chunyuan Li. Learning customized visual models with retrieval-augmented knowledge. In *CVPR*, 2023. 3

[70] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023. 8

[71] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 1, 2

[72] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *CVPR*, 2022. 1, 2, 3

[73] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1

[74] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *NeurIPS*, 2022. 8

[75] Anas Mahmoud, Mostafa Elhoushi, Amro Abbas, Yu Yang, Newsha Ardalani, Hugh Leather, and Ari S Morcos. Sieve: Multimodal dataset pruning using image captioning models. *arXiv preprint arXiv:2310.02110*, 2023. 8

[76] Sachin Mehta and Mohammad Rastegari. Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer. *arXiv preprint arXiv:2110.02178*, 2021. 2

[77] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection. In *ECCV*, 2022. 2, 3

[78] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 2014. 8

[79] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training. In *ECCV*, 2022. 3

[80] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *ICCV*, 2017. 8

[81] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2, 3

[82] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1, 2, 3, 8

[83] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jon Shlens. Stand-alone self-attention in vision models. *NeurIPS*, 2019. 2

[84] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *ICML*, 2019. 6

[85] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 3

[86] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115 (3):211–252, 2015. 1, 2, 6

[87] Shiori Sagawa, Pang Wei Koh, Tony Lee, Irena Gao, Sang Michael Xie, Kendrick Shen, Ananya Kumar, Weihua Hu, Michihiro Yasunaga, Henrik Marklund, et al. Extending the wilds benchmark for unsupervised adaptation. In *ICLR*, 2022. 6

[88] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, 2018. 2, 5, 6

[89] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 2022. 2, 3, 7

[90] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020. 5, 6

[91] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[92] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *CVPR*, 2019. 8

[93] Aravind Srinivas, Tsung-Yi Lin, Niki Parmar, Jonathon Shlens, Pieter Abbeel, and Ashish Vaswani. Bottleneck transformers for visual recognition. In *CVPR*, 2021. 2

[94] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 2, 3, 7

[95] Shuyang Sun, Weijun Wang, Andrew Howard, Qihang Yu, Philip Torr, and Liang-Chieh Chen. Remax: Relaxing for better training on efficient panoptic segmentation. *NeurIPS*, 2024. 1

[96] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 2, 5

[97] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019. 2

[98] Mingxing Tan and Quoc Le. Efficientnetv2: Smaller models and faster training. In *ICML*, 2021. 2

[99] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 2

[100] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *ICCV*, 2021. 5

[101] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 3

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

[103] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. In *ECCV*, 2022. 1, 2

[104] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 1, 2, 5, 6

[105] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *NeurIPS*, 2019. 6

[106] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-DeepLab: Stand-Alone Axial-Attention for Panoptic Segmentation. In *ECCV*, 2020. 2, 3

[107] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *CVPR*, 2021. 1, 3

[108] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *ICCV*, 2021. 1, 2

[109] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *ICCV*, 2021. 2

[110] Kan Wu, Houwen Peng, Zhenghong Zhou, Bin Xiao, Mengchen Liu, Lu Yuan, Hong Xuan, Michael Valenzuela, Xi Stephen Chen, Xinggang Wang, et al. Tinyclip: Clip distillation via affinity mimicking and weight inheritance. In *ICCV*, 2023. 3

[111] Size Wu, Wenwei Zhang, Lumin Xu, Sheng Jin, Xiangtai Li, Wentao Liu, and Chen Change Loy. Clipself: Vision transformer distills itself for open-vocabulary dense prediction. *arXiv preprint arXiv:2310.01403*, 2023. 1, 6, 8

[112] Tete Xiao, Mannat Singh, Eric Mintun, Trevor Darrell, Piotr Dollár, and Ross Girshick. Early convolutions help transformers see better. *NeurIPS*, 2021. 2

[113] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *ICCV*, 2017. 2

[114] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *CVPR*, 2023. 2

[115] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for zero-shot semantic segmentation with pre-trained vision-language model. In *ECCV*, 2022. 2

[116] Chenglin Yang, Siyuan Qiao, Qihang Yu, Xiaoding Yuan, Yukun Zhu, Alan Yuille, Hartwig Adam, and Liang-Chieh Chen. Moat: Alternating mobile convolution and attention brings strong vision models. In *ICLR*, 2023. 1, 2

[117] Xuan Yang, Liangzhe Yuan, Kimberly Wilber, Astuti Sharma, Xiuye Gu, Siyuan Qiao, Stephanie Debats, Huisheng Wang, Hartwig Adam, Mikhail Sirotenko, and Liang-Chieh Chen. Polymax: General dense prediction with mask transformer. *arXiv preprint arXiv:2311.05770*, 2024. 1

[118] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. 6

[119] Haichao Yu, Yu Tian, Sateesh Kumar, Linjie Yang, and Heng Wang. The devil is in the details: A deep dive into the rabbit hole of data filtering. *arXiv preprint arXiv:2309.15954*, 2023. 2, 7, 8

[120] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 3

[121] Qihang Yu, Yingda Xia, Yutong Bai, Yongyi Lu, Alan L Yuille, and Wei Shen. Glance-and-gaze vision transformer. *NeurIPS*, 2021. 1

[122] Qihang Yu, Huiyu Wang, Dahun Kim, Siyuan Qiao, Maxwell Collins, Yukun Zhu, Hartwig Adam, Alan Yuille,

and Liang-Chieh Chen. Cmt-deeplab: Clustering mask transformers for panoptic segmentation. In *CVPR*, 2022. 1

[123] Qihang Yu, Huiyu Wang, Siyuan Qiao, Maxwell Collins, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. k-means Mask Transformer. In *ECCV*, 2022. 1

[124] Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Convolutions die hard: Open-vocabulary segmentation with single frozen convolutional clip. *NeurIPS*, 2023. 1, 2, 3, 6, 8

[125] Qihang Yu, Xiaohui Shen, and Liang-Chieh Chen. Towards open-ended visual recognition with large language model. *arXiv preprint arXiv:2311.08400*, 2023. 8

[126] Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. Metaformer is actually what you need for vision. In *CVPR*, 2022. 2

[127] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023. 8

[128] Xiaowei Yu, Yao Xue, Lu Zhang, Li Wang, Tianming Liu, and Dajiang Zhu. Exploring the influence of information entropy change in learning systems. *arXiv preprint arXiv:2309.10625*, 2023. 2

[129] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *CVPR*, 2021. 2, 8

[130] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. The visual task adaptation benchmark. 2019. 6

[131] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *CVPR*, 2022. 5

[132] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *CVPR*, 2022. 3

[133] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023. 3, 7

[134] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 8

[135] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *ECCV*, 2022. 2

[136] Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Zihang Jiang, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021. 2, 5

[137] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2, 3, 8