When Do We Not Need Larger Vision Models?

Baifeng Shi[1], Ziyang Wu[1], Maolin Mao[1], Xin Wang[2], and Trevor Darrell[1]

[1] UC Berkeley
[2] Microsoft Research

Abstract. Scaling up the size of vision models has been the *de facto* standard to obtain more powerful visual representations. In this work, we discuss the point beyond which larger vision models are *not* necessary. First, we demonstrate the power of **S**caling on **S**cales ($\mathbf{S^2}$), whereby a pre-trained and frozen smaller vision model (*e.g.*, ViT-B or ViT-L), run over multiple image scales, can outperform larger models (*e.g.*, ViT-H or ViT-G) on classification, segmentation, depth estimation, Multimodal LLM (MLLM) benchmarks, and robotic manipulation. Notably, S^2 achieves state-of-the-art performance in detailed understanding of MLLM on the V^* benchmark, surpassing models such as GPT-4V. We examine the conditions under which S^2 is a preferred scaling approach compared to scaling on model size. While larger models have the advantage of better generalization on hard examples, we show that features of larger vision models can be well approximated by those of multi-scale smaller models. This suggests most, if not all, of the representations learned by current large pre-trained models can also be obtained from multi-scale smaller models. Our results show that a multi-scale smaller model has comparable learning capacity to a larger model, and pre-training smaller models with S^2 can match or even exceed the advantage of larger models. We release a Python package that can apply S^2 on any vision model with *one line of code*: `https://github.com/bfshi/scaling_on_scales`.

Keywords: Vision model scaling · Scaling on scales

1 Introduction

Scaling up model size has been one of the key drivers of recent progress in various domains of artificial intelligence, including language modeling [9, 50, 68], image and video generation [8, 34, 53, 77], *etc*. Similarly, for visual understanding, larger models have consistently shown improvements across a wide range of downstream tasks given sufficient pre-training data [13, 48, 63, 80]. This trend has led to the pursuit of gigantic models with up to tens of billions of parameters as a default strategy for achieving more powerful visual representations and enhanced performance on downstream tasks [13, 18, 22, 62].

In this work, we revisit the question: *Is a larger model always necessary for better visual understanding?* Instead of scaling up model size, we consider scaling up image scales, which we call **S**caling on **S**cales ($\mathbf{S^2}$). With S^2, a pre-trained and frozen smaller vision model (*e.g.*, ViT-B or ViT-L) is run on multiple image

scales to generate a multi-scale representation. We take a model pre-trained on one single image scale (*e.g.*, 224^2), interpolate the image to multiple scales (*e.g.*, 224^2, 448^2, 672^2), extract features on each scale by splitting larger images into sub-images of regular size (224^2) and processing each separately before pooling them and concatenating with features from the original representation (Fig. 1).

Surprisingly, from evaluations on visual representations of various pre-trained models (*e.g.*, ViT [21], DINOv2 [48], OpenCLIP [13], MVP [52]), we show that smaller models with S^2 scaling consistently outperform larger models on classification, semantic segmentation, depth estimation, MLLM benchmarks, and robotic manipulation, with significantly fewer parameters (0.28× to 0.07×) and comparable GFLOPs. Notably, by scaling up image scale to 1008^2, we achieve state-of-the-art performance in MLLM visual detail understanding on V^* benchmark [72], surpassing open-source and even commercial MLLMs like Gemini Pro [65] and GPT-4V [1].

We further examine conditions under which S^2 is a preferred scaling approach compared to model size scaling. We find that while smaller models with S^2 achieve better downstream performance than larger models in many scenarios, larger models can still exhibit superior generalization on hard examples. This prompts an investigation into whether smaller models can achieve the same level of generalization capability as larger ones. Surprisingly, we find that the features of larger models can be well approximated by multi-scale smaller models through a single linear transform, which means smaller models should have at least a similar learning capacity of their larger counterparts. We hypothesize that their weaker generalization stems from being pre-trained with single image scale only. Through experiments of ImageNet-21k pre-training on ViT, we show that pre-training with S^2 scaling improves the generalizability of smaller models, enabling them to match or even exceed the advantages of larger models.

2 Related Work

Multi-scale representation has been a common technique to recognize objects in a scale-invariant way since the era of feature engineering [17,19,43] and is later introduced into convolutional neural networks [37,55,67,69] to extract features with both high-level semantics and low-level details. It has become a default test-time augmentation method for tasks such as detection and segmentation [15,73], albeit at the cost of significantly slower inference speeds and typically limited image scales (up to 2×). Along with recent progress in vision transformers (ViT), variants of multi-scale ViTs [10,23,35,76] as well as hierarchical ViTs [41, 57] have been proposed. However, these studies have not explored multi-scale representation as a general scaling approach as they usually design special architectures and are not applicable to common pre-trained vision models.

Scaling Vision Models. Training models with an increasing number of parameters has been the default approach to obtaining more powerful representations for visual pre-training [21,30,42,48]. Previous research has studied how to optimally

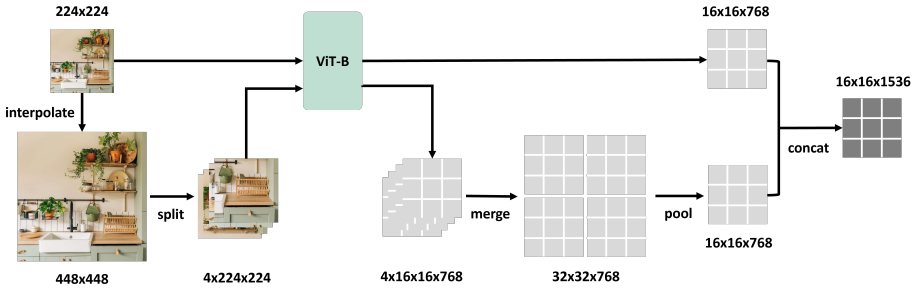

Fig. 1: S^2-Wrapper is a simple mechanism that extends any pre-trained vision model to multiple image scales in a parameter-free manner. Taking ViT-B as an example, S^2-Wrapper first interpolates the input image to different scales (*e.g.*, 224^2 and 448^2) and splits each into several sub-images of the same size as the default input size ($448^2 \rightarrow 4 \times 224^2$). For each scale, all sub-images are fed into the same model and the outputs (*e.g.*, 4×16^2) are merged into feature map of the whole image (32^2). Feature maps of different scales are average-pooled to the original spatial size (16^2) and concatenated together. The final multi-scale feature has the same spatial shape as single-scale feature while having higher channel dimension (*e.g.*, 1536 *vs*. 768).

scale up vision models in terms of balancing model width, depth, and input resolution [5, 20, 63, 64, 71], although they are usually limited to convolutional networks or even specific architectures such as ResNet [30]. Recent work also explores model size scaling of vision transformers in various settings [3, 13, 18, 54, 80]. Others have incorporated high-resolution images into pre-training [24, 41, 42, 48], although the maximum resolution typically does not exceed 512^2 due to unbearable demands of computational resources. Hu *et al*. [32] study scaling on image scales through adjusting patch size for Masked Autoencoder (MAE) [29] where scaling is only applied on pre-training but not on downstream tasks.

3 The Power of Scaling on Scales

As an alternative to the conventional approach of scaling model size, we show the power of Scaling on Scales (S^2), *i.e.*, keeping the same size of a pre-trained model while running it on more and more image scales. From case studies on image classification, semantic segmentation, depth estimation, Multimodal LLMs, as well as robotic manipulation, we observe that S^2 scaling on a smaller vision model (*e.g.*, ViT-B or ViT-L) often gives comparable or better performance than larger models (*e.g.*, ViT-H or ViT-G), suggesting S^2 is a competitive scaling approach. In the following, we first introduce S^2-Wrapper, a mechanism that extends any pre-trained frozen vision model to multiple image scales without additional parameters (Sec. 3.1). We then compare S^2 scaling and model size scaling in Sec. 3.2 - Sec. 3.3.

3.1 Scaling Pre-Trained Vision Models to Multiple Image Scales

We introduce S^2-Wrapper, a parameter-free mechanism to enable multi-scale feature extraction on any pre-trained vision model. Regular vision models are normally pre-trained at a single image scale (*e.g.*, 224^2). S^2-Wrapper extends a pre-trained model to multiple image scales (*e.g.*, 224^2, 448^2) by splitting different scales of images to the same size as seen in pre-training. Specifically, given the image at 224^2 and 448^2 scales, S^2-Wrapper first divides the 448^2 image into four 224^2 sub-images, which along with the original 224^2 image are fed to the same pre-trained model. The features of four sub-images are merged back to the large feature map of the 448^2 image, which is then average-pooled to the same size as the feature map of 224^2 image. Output is the concatenation of feature maps across scales. The whole process is illustrated in Fig. 1. Note that instead of directly using the 448^2 resolution image, we obtain the 448^2 image by interpolating the 224^2 image. This is to make sure no additional high-resolution information is introduced so we can make a fair comparison with model size scaling which never sees the high-resolution image. For practitioners, directly using the high-resolution image is recommended.

There are several key designs that make S^2-Wrapper efficient, effective, and easy to scale: (i) splitting the large image into small sub-images, instead of directly running on the whole large image, avoids quadratic computation complexity in self-attention and prevents performance degradation caused by position embedding interpolation [7], (ii) processing individual sub-images instead of using window attention allows using a pre-trained model that does not support window attention and avoids training additional parameters (*e.g.*, relative position embedding) from scratch, (iii) interpolating the large feature map into the regular size makes sure the number of output tokens stays the same, preventing computational overhead in downstream applications such as MLLMs. Ablations of the designs can be found in Appendix. Note that we do not claim the novelty of extracting multi-scale features. Instead, we simply choose the most efficient and effective algorithm design and study its scaling property.

3.2 Scaling on Image Scales Can Beat Scaling on Model Size

S^2-Wrapper enables S^2 scaling, *i.e.*, keeping the same size of a pre-trained model while getting more and more powerful features by running on more and more image scales. Here we compare the scaling curve of S^2 to the regular approach of scaling up model size and show that S^2 scaling is a competitive, and in some cases, preferred scaling approach. To get a holistic analysis of two scaling approaches, we test their scaling curves on three representative tasks (image classification, semantic segmentation, and depth estimation) which correspond to the three dimensions of vision model capability [46], as well as on MLLMs and robotic manipulation which reflect the comprehensive ability of visual understanding.

Case study: image classification, semantic segmentation, and depth estimation. We use ImageNet [56], ADE20k [85], and NYUv2 [59] datasets for each task, respectively. We test on three families of pre-trained models (ViT [21],

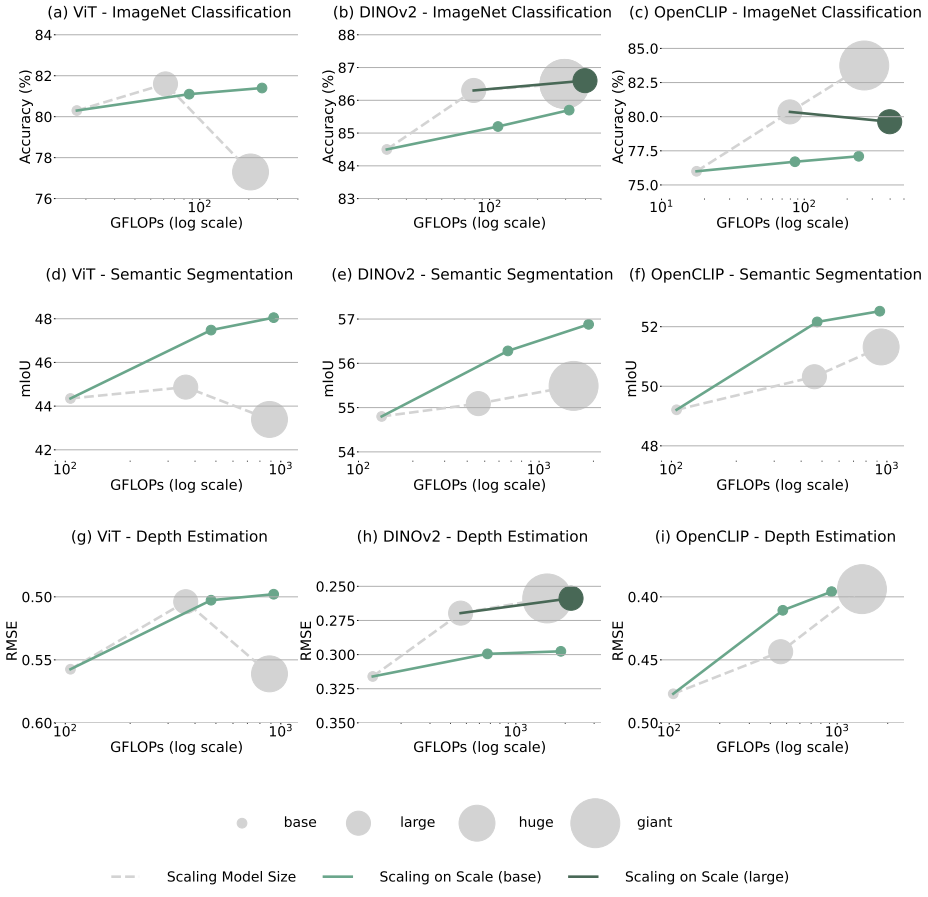

Fig. 2: Comparison of S^2 scaling and model size scaling on three models (ViT, DINOv2, and OpenCLIP) and three tasks (ImageNet classification, semantic segmentation, and depth estimation). For each model and each task, we test base, large, and huge/giant model for model size scaling (plotted in gray curve). For S^2 scaling (plotted in green curve), we test three sets of scales from single-scale (1x) to multi-scale (up to 3x), and we adjust each set of scale so that it matches the GFLOPs of the respective model size. Note that for specific models and tasks, we test S^2 scaling on both base and large models (plotted in light green and dark green curves separately). We can see that in (a), (d), (e), (f), (g), and (i), base model with S^2 scaling already achieves comparable or better performances than larger models with similar GFLOPs and much smaller model size. For (b), (h), S^2 scaling from large model is comparable with giant model, again with similar GFLOPs and fewer parameters. The only failure case is (c), where S^2 scaling on either base or large model does not compete with model size scaling. One possible reason is the model is scaled up to larger images after pre-training but not during pre-training, which may affect its generalizability. In Sec. 4.3 we show that pre-training with S^2 can further improve performance.

DINOv2 [48], and OpenCLIP [13]), spanning pre-training with different datasets (ImageNet-21k, LVD-142M, LAION-2B) and different pre-training objectives (supervised, unsupervised, and weakly-supervised). To see if the same observation holds for convolutional networks, we also test on ConvNeXt [42] (See Appendix). To fairly evaluate the representation learned from pre-training, we freeze the backbone and only train the task-specific head for all experiments. We use a single linear layer, Mask2former [11], and VPD depth decoder [83] as decoder heads for three tasks, respectively. For model size scaling, we test the performance of base, large, and huge or giant size of each model on each task. For S^2 scaling, we test three sets of scales including (1x), (1x, 2x), (1x, 2x, 3x). For example, for ViT on ImageNet classification, we use three sets of scales: (224^2), (224^2, 448^2), and (224^2, 448^2, 672^2), which have the comparable GFLOPs as ViT-B, ViT-L, and ViT-H, respectively. Note that the scales for specific models and tasks are adjusted to match the GFLOPs of respective model sizes. The detailed configurations for each experiment can be found in Appendix.

The scaling curves are shown in Fig. 2. We can see that in six out of nine cases ((a), (d), (e), (f), (g), (i)), S^2 scaling from base models gives a better scaling curve than model size scaling, outperforming large or giant models with similar GFLOPs and much fewer parameters. In two cases ((b) and (h)), S^2 scaling from base models has less competitive results than large models, but S^2 scaling from large models performs comparatively with giant models. The only failure case is (c) where both base and large models with S^2 scaling fail to compete with the giant model. Note that ViT-H is worse than ViT-L on all three tasks possibly due to the sub-optimal pre-training recipe [61]. We observe that S^2 scaling has more advantages on dense prediction tasks such as segmentation and depth estimation, which matches the intuition that multi-scale features can offer better detailed understanding which is especially required by these tasks. For image classification, S^2 scaling is sometimes worse than model size scaling (*e.g.*, multi-scale DINOv2-B *vs*. DINOv2-L). We hypothesize this is due to the weak generalizability of base model feature because we observe that the multi-scale base model has a lower training loss than the large model despite the worse performance, which indicates overfitting. In Sec. 4.3 we show that this can be fixed by pre-training with S^2 scaling as well.

Case study: Multimodal LLMs. We compare S^2 scaling and model size scaling on MLLMs. We use a LLaVA [39]-style model where LLM is a Vicuna-7B [14] and the vision backbone is OpenCLIP. We keep the same LLM and only change the vision backbone. For model size scaling, we test vision model sizes of large, huge, and big-G. For S^2 scaling, we keep the large-size model and test scales of (224^2), (224^2, 448^2), and (224^2, 448^2, 896^2). For all experiments, we keep the vision backbone frozen and only train a projector layer between the vision feature and LLM input space as well as a LoRA [31] on LLM. We follow the same training recipe as in LLaVA-1.5 [38]. We evaluate three types of benchmarks: (i) visual detail understanding (V^* [72]), (ii) VQA benchmarks (VQAv2 [27], TextVQA [60], VizWiz [28]), and (iii) MLLM benchmarks (MMMU [79], MathVista [44], MMBench [40], SEED-Bench [36], MM-Vet [78]).

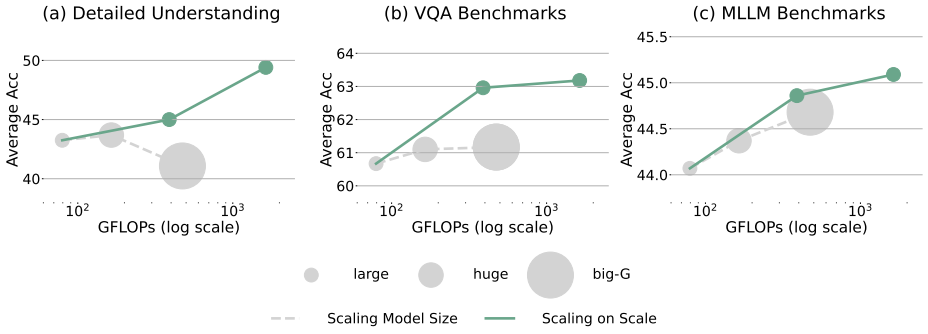

Fig. 3: Comparison of S^2 scaling and model size scaling on MLLM. S^2 scaling has comparable and even better scaling curve than model size scaling on all three types of benchmarks. Notably, S^2 scaling significantly improves the detailed understanding capability of MLLM, boosting the accuracy on V^* benchmark by over 6%. Overall, using large image scales consistently gives better performance while using larger model can degrade model performance in certain cases.

A comparison of the two scaling approaches is shown in Fig. 3. We report the average accuracy on each type of benchmarks. We can see that on all three types of benchmarks, S^2 scaling on large-size model performs better than larger models, using similar GFLOPs and much smaller model size. Especially, scaling to 896^2 improves the accuracy of detailed understanding by about 6%. On all benchmarks, larger image scales consistently improve performance while bigger models sometimes fail to improve or even hurt performance. These results suggest S^2 is a preferable scaling approach for vision understanding in MLLMs as well.

We also observe that LLaVA-1.5, when equipped with S^2 scaling, is already competitive or better than state-of-the-art open-source and even commercial MLLMs. Results are shown in Table 1. Here we use OpenAI CLIP [49] as the vision model for fair comparison. On visual detail understanding, LLaVA-1.5 with S^2 scaling outperforms all other open-source MLLMs as well as commercial models such as Gemini Pro and GPT-4V. This is credited to the highly fine-grained features we are able to extract by scaling image resolution to 1008^2. A qualitative example is shown in Figure 4. We can see that LLaVA-1.5 with S^2 is able to recognize an extremely small object that only takes 23×64 pixels in a 2250×1500 image and correctly answer the question about it. In the meantime, both GPT-4V and LLaVA-1.5 fail to give the correct answer. More qualitative examples are shown in Appendix. On VQA and MLLM benchmarks, S^2 consistently improves the model performance as well. In contrast to previous experiments, here we directly use the high-resolution image instead of interpolating from the low-resolution image in order to compare with the state of the arts. Note that despite the large image scale, we keep the same number of image tokens as baseline LLaVA-1.5 since we interpolate the feature map of the large-scale images to the same size as that of the original image (see Section 3.1). This makes sure the context length (and thus the computational cost) of LLM does not increase when

Table 1: Results on MLLM. We evaluate three types of benchmarks: visual detail understanding (V^* [72]), VQA benchmarks (VQAv2 [27], TextVQA [60], VizWiz [28]), and MLLM benchmarks (MMMU [79], MathVista [44], MMBench [40], SEED-Bench [36], MM-Vet [78]). Notably, S^2 significantly improves the detailed understanding capability on V^* benchmark, outperforming commercial models such as GPT-4V.

			Detail		VQA			MLLM Benchmarks				
Model	Res.	#Tok	V^*_{Att}	V^*_{Spa}	VQA^{v2}	VQA^{T}	Viz	MMMU	Math	MMB	SEED	Vet
Commercial or proprietary models												
GPT-4V [1]	-	-	**51.3**	60.5	77.2	78.0	-	**56.8**	**49.9**	75.8	71.6	**67.6**
Gemini Pro [65]	-	-	40.9	59.2	71.2	74.6	-	47.9	45.2	73.6	70.7	64.3
Qwen-VL-Plus [66]	-	-	-	-	-	**78.9**	-	45.2	43.3	-	-	-
Open-source models												
InstructBLIP-7B [16]	224	-	25.2	47.4	-	50.1	34.5	-	-	36.0	-	26.2
QwenVL-7B [2]	448	1024	-	-	78.8	63.8	35.2	-	-	38.2	-	-
QwenVL-Chat-7B [2]	448	1024	-	-	78.2	61.5	38.9	-	-	60.6	-	-
CogVLM-Chat [70]	490	1225	-	-	**82.3**	70.4	-	41.1	34.5	**77.6**	**72.5**	51.1
LLaVA-1.5-7B [38]	336	576	43.5	56.6	78.5	58.2	50.0	36.2	25.2	64.3	65.7	30.5
- S^2 Scaling	1008	576	**51.3**	61.8	80.0	61.0	50.1	37.7	25.3	66.2	67.9	32.4
LLaVA-1.5-13B [38]	336	576	41.7	55.3	80.0	61.3	53.6	36.4	27.6	67.8	68.2	35.4
- S^2 Scaling	1008	576	50.4	**63.2**	80.9	63.1	**56.0**	37.4	27.8	67.9	68.9	36.4

using larger image scales, allowing us to use much higher resolution than the baselines.

Case study: robotic manipulation. We compare S^2 and model size scaling on a robotic manipulation task of cube picking. The task requires controlling a robot arm to pick up a cube on the table. We train a vision-based end-to-end policy on 120 demos using behavior cloning, and evaluate the success rate of picking on 16 randomly chosen cube positions, following the setting in [51]. We use MVP [52] as the pre-trained vision encoder to extract visual features which are fed to the policy. Please refer to Appendix for the detailed setting. To compare S^2 and model size scaling, we evaluate base and large model with single scale of (224^2), as well as multi-scale base model with scales of (224^2, 448^2). Results are shown in Figure 5. Scaling from base to large model improves the success rate by about 6%, while scaling to larger image scales improves the success rate by about 19%. This demonstrates the advantage of S^2 over model size scaling on robotic manipulation tasks as well.

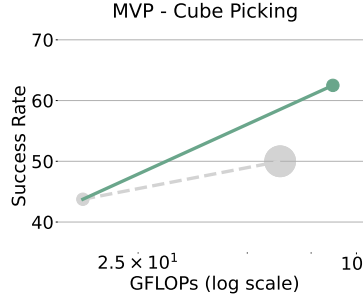

Fig. 5: S^2 *vs.* model size scaling on cube picking task. S^2 scaling on base-size model improves the success rate by about 19% while scaling from base to large model improves by about 6%.

Fig. 4: LLaVA-1.5 with S^2 scaling is able to recognize extremely fine-grained details in an image, *e.g.*, the color of a water bottle which lives in only 23×64 pixels of a 2250 × 1500 image.

3.3 The Sweet Spot Between Model Size Scaling and S^2 Scaling

While S^2 scaling outperforms model size scaling on a wide range of downstream tasks, a natural question arises: on which model size should we perform S^2 scaling? We show that it depends on different pre-trained models. For certain models, S^2 scaling from a large-size model gives an even better scaling curve when S^2 scaling from base model already beats larger models. As an example, we compare S^2 scaling from base and large models on semantic segmentation for ViT, DINOv2, and OpenCLIP. Results are shown in Fig. 6. We can see that for ViT and OpenCLIP, S^2 scaling from base model is better than from large model when the amount of computation is less than that of giant-size model. These two curves eventually converge after going beyond the GFLOPs of giant model. This means S^2 scaling from large model has no significant benefit than from base model. On the other hand, for DINOv2 we observe a clear advantage for S^2 scaling from large model. When reaching the same level of GFLOPs as giant-size model, S^2 scaling from large model beats S^2 scaling from base model by about 1 mIoU. These results indicate the optimal balancing between model size scaling and S^2 scaling varies for different models.

4 The (Non)Necessity of Scaling Model Size

Results from Sec. 3 suggest S^2 is a preferred scaling approach than model size scaling for various downstream scenarios. Nevertheless, larger vision models seem still necessary in certain cases (such as Fig. 2(c)) where S^2 scaling cannot compete with model size scaling. In the following, we first study the advantage of larger models and show they usually generalize better on rare or hard instances than multi-scale smaller models (Sec. 4.1). Then, we explore if smaller models with S^2

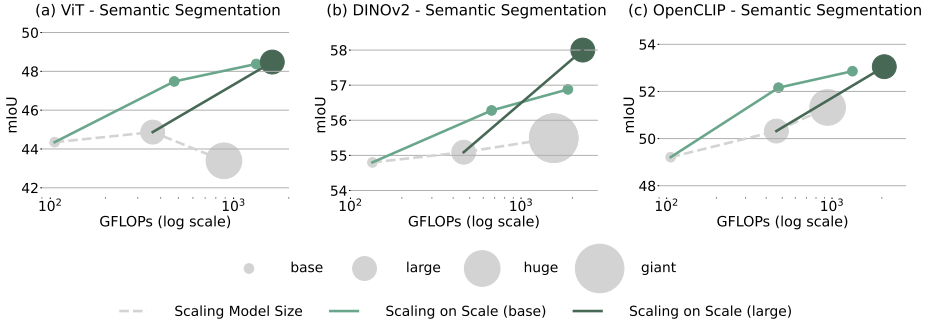

Fig. 6: Which model size should we scale up image scales on? The answer varies for different pre-trained models. For ViT and OpenCLIP, S^2 scaling from base or large model gives similar performances under computation budget beyond huge-size model while the former performs better under similar GFLOPS as large-size model. For DINOv2, S^2 scaling from large size model has better performance than scaling from base size, especially under the same level of computation budget as giant-size model.

scaling can achieve the same capability. We find that features of larger models can be well approximated by features of multi-scale smaller models, which means smaller models can learn what larger models learn to a large extent (Sec. 4.2). Based on this observation, we verify that multi-scale smaller models have similar capacity as larger models, and pre-training with S^2 scaling endows smaller models with similar or better generalization capability than larger models (Sec. 4.3).

4.1 Larger Models Generalize Better on Hard Examples

We use image classification as a testbed to understand the advantage of larger models. We conduct a qualitative analysis of what kinds of images are recognized better by a larger model but not by using more image scales. Specifically, we find samples in ImageNet that a larger model (ViT-L) improves the most over a smaller model (ViT-B) but a multi-scale model (ViT-B-S^2) fails to improve, as shown in Fig. 7. For each sample, we also find two easy samples (which two models both recognize correctly) from the same class as a comparison. We can see that there are mainly two types of images that larger models have advantages on. The first type is rare samples. For example, a television or a flute but in the form of a sculpture instead of regular ones (Fig. 7(a)). Larger models have larger capacity to learn to classify these rare examples during pre-training. The second type (Fig. 7(b)) is ambiguous examples, where the object can belong to either category (*e.g.*, lotion and soap dispenser), or there are two categories co-existing in the same image and both labels should be correct (*e.g.*, airship and traffic light). In this case, despite multiple correct labels, the large model is able to remember the label presented in the dataset during pre-training. While the second type is due to the flawed labeling process of ImageNet which makes it an unfair comparison and does not imply any disadvantage of multi-scale smaller

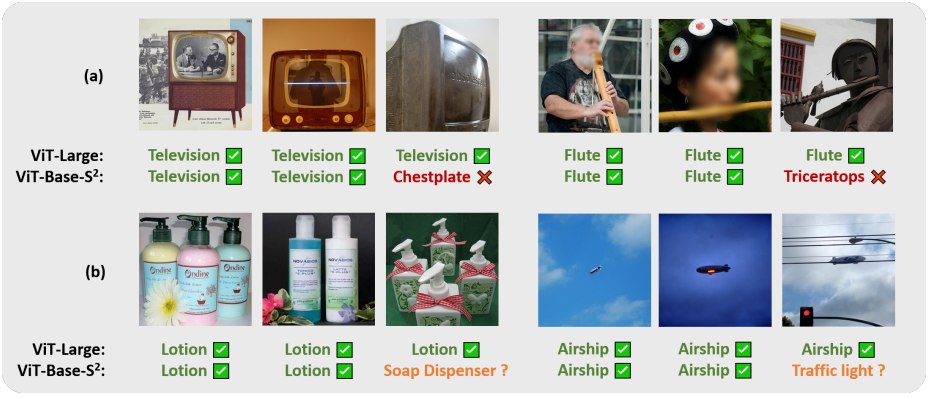

Fig. 7: Types of samples that ViT-L improves the most but ViT-B-S^2 does not. (a) Rare cases. These samples clearly belong to the class but are hard to classify due to the rare appearance (*e.g.*, sculptures of television and flute). **(b) Ambiguous cases**. These samples have ambiguous labels. For example, the lotion could also be soap dispenser due to their high similarity, or the label could be either airship or traffic light when these two objects co-exist.

models [6, 47], the first type indicates larger model can generalize better on rare or hard cases.

4.2 Can Smaller Models Learn What Larger Models Learn?

Is the advantage of larger models due to some unique representation they have learned that smaller models cannot learn? We design experiments to study how much of the representation of larger models is also learned by multi-scale smaller models. Surprisingly, our preliminary results suggest that *most, if not all, of the representation of larger models is also learned by multi-scale smaller models.*

To quantify how much of the representation of a larger model (*e.g.*, ViT-L) is also learned by a multi-scale smaller model (*e.g.*, ViT-B-S^2), we adopt a reconstruction-based evaluation, *i.e.*, we train a linear transform to reconstruct the representation of a larger model from that of a multi-scale smaller model. Intuitively, low reconstruction loss means the representation of larger model can be equivalently learned by the multi-scale smaller model (through a linear transform) to a large extent. More formally, the reconstruction loss reflects the mutual information between two sets of representations. If we use MSE loss for reconstruction, the mutual information equals $I = -\log(l/l_0)$, where l is the reconstruction loss and l_0 is the loss of vanilla reconstruction where the large model representation is reconstructed by a dummy vector (See Appendix). This quantifies how much information in the larger model representation is also contained in the multi-scale smaller model. We use a linear transform for reconstruction to (i) account for operations that keep the representation equivalence (e.g., channel permutation), (ii) measure the information that is

Table 2: Reconstructing representation of larger models from representation of regular or multi-scale smaller models. We test three classes of models (ViT, OpenCLIP, and MAE), and for each class we test base, multi-scale base (Base-S^2), and huge or giant model. We report results on both training and test set of ImageNet-1k, and for each we report the reconstruction loss, the amount of information reconstructed, and the percentage of information reconstructed compared to huge or giant model.

Model Class	Target	Source	Train Set			Test Set		
			Loss	Info	Ratio (%)	Loss	Info	Ratio (%)
ViT	Large	Base	0.1100	0.440	82.9%	0.0994	0.524	87.6%
		Base-S^2	0.1040	0.521	**98.1%**	0.0942	0.601	**100.5%**
		Huge	0.1033	0.531	100%	0.0944	0.598	100%
MAE	Large	Base	0.0013	7.460	97.3%	0.0010	7.840	96.0%
		Base-S^2	0.0011	7.694	**100.3%**	0.0009	7.972	**97.6%**
		Huge	0.001	7.669	100%	0.0008	8.169	100%
OpenCLIP	Large	Base	0.3693	1.495	92.7%	0.3413	1.723	90.7%
		Base-S^2	0.3408	1.611	**99.9%**	0.3170	1.830	**96.3%**
		Giant	0.3402	1.613	100%	0.3022	1.900	100%
OpenCLIP	Huge	Base	0.3926	1.407	83.2%	0.4231	1.413	80.8%
		Base-S^2	0.3670	1.504	**88.9%**	0.3970	1.505	**86.0%**
		Giant	0.3221	1.692	100%	0.3354	1.749	100%

useful for downstream tasks considering the task decoders are usually light-weight modules such as a single linear layer [75].

Moreover, in practice we find the reconstruction loss is usually nowhere near zero. We hypothesize this is because part of the feature is *non-reconstructable* by nature, *i.e.*, feature that is not relevant to the pre-training task and is learned due to randomness in weight initialization, optimization dynamics, *etc.*, thus cannot be reconstructed from another model's feature. To this end, we use an even larger (*e.g.*, ViT-G) model to reconstruct the large model features as a comparison. Its reconstruction loss and corresponding mutual information are denoted by l^* and $I^* = -\log(l^*/l_0)$. If we assume that, when pre-trained on the same task and the same dataset, any task-relevant feature learned by a smaller model can also be learned by a larger model, then all the useful features in a large-size model should be reconstructable by a huge or giant model as well. This means I^*, the amount of information reconstructed from a huge or giant model, should serve as an *upper bound* of I. We empirically find this is indeed the case (see below). Therefore, we use the reconstruction ratio I/I^* to measure how much representation in a larger model is also learned by a multi-scale smaller model.

We evaluate three classes of models: (i) ViT [21] pre-trained on ImageNet-21k, (ii) OpenCLIP [13] pre-trained on LAION-2B, and (iii) MAE [29] pre-trained on ImageNet-1k. Reconstruction loss is averaged over all output tokens and is evaluated on ImageNet-1k. Results are shown in Tab. 2. Compared to base models, we observe that multi-scale base models consistently have lower loss and

Table 3: Training loss on instance memorization and image classification. Base model with S^2 scaling has similar memorization and classification loss, which implies it has at least the same level of model capacity as large model.

Model	Mem. Loss	Cls. Loss (DINOv2)	Cls. Loss (OpenCLIP)
Base	1.223	3.855	4.396
Large	**1.206**	3.350	**3.735**
Base-S^2	**1.206**	**2.921**	3.754

Table 4: Pre-training with S^2. Applying S^2 on a already pre-trained base model has sub-optimal performance compared to large model, while pre-training with S^2 makes base model better than large model.

Model	Pre-train w/ S^2	Acc.
Base		80.3
Large		81.6
Base-S^2	✗	81.1
Base-S^2	✓	**82.4**

reconstructs more information of large model representation (*e.g.*, 0.521 *vs*. 0.440 for ViT). More interestingly, we find that the amount of information reconstructed from a multi-scale base model is usually close to that of a huge or giant model, although sometimes slightly lower but never exceeding by a large margin. For example, while OpenCLIP-Base reconstructs 92.7% of the information, the multi-scale base model can reconstruct 99.9%. For other models, the reconstruction ratio of Base-S^2 model is usually close to 100% while never exceeding by more than 0.5%. This implies (i) huge/giant models are indeed a valid upper bound of feature reconstruction, and (ii) most part of the feature of larger models is also learned by multi-scale smaller models. The only exception is when we reconstruct OpenCLIP-Huge feature, the reconstruction ratio is 88.9%. Although it's not near 100%, it is still significantly better than the base-size model which means at least a large part of the huge model feature is still multi-scale feature. These results imply smaller models with S^2 scaling should have at least a similar level of capacity to learn what larger models learn. On the other hand, we also notice that there exists a gap between train and test set, *i.e.*, the reconstruction ratio on test set can be lower than train set (*e.g.* 96.3% *vs*. 99.9% on OpenCLIP-L). One possible reason is we only apply S^2 after pre-training and the base model feature pre-trained on single image scale has weaker generalizability.

4.3 Pre-Training With S^2 Makes Smaller Models Better

Given that most of the representation larger models have learned is also learned by multi-scale smaller models, we conjecture smaller models with S^2 scaling have at least similar capacity as larger models. Since larger capacity allows memorizing more rare and atypical instances during pre-training when given sufficient data and thus improves generalization error [4, 12, 25, 26, 45], we further speculate smaller models can achieve similar or even better generalizability than larger models if pre-trained with S^2 scaling as well. We verify these in the following.

Multi-scale smaller models have similar capacity as larger models. To measure the model capacity, we use two surrogate metrics: (i) memorization

capability, and (ii) training loss on a specific task. For memorization capability, given a dataset (*e.g.*, ImageNet), we regard each image as a separate category and train the model to classify individual images, which requires the model to memorize every single image. The classification loss reflects how well each instance is memorized and thus how large the model capacity is [81]. We adopt the training pipeline from [74]. For training loss, we report classification loss on the training set of ImageNet-1k for DINOv2 and OpenCLIP. Lower loss means the model fits the training data better, which implies a larger model capacity. Results are shown in Tab. 3. For instance memorization, we can see that ViT-B with S^2 scaling (224^2 and 448^2) has a similar loss as ViT-L. For ImageNet classification, ViT-B-S^2 has a similar training loss as ViT-L for OpenCLIP, and an even lower loss for DINOv2. These results suggest that multi-scale smaller models have at least comparable model capacity as larger models.

Pre-training with S^2 makes smaller models better. We evaluate ImageNet classification of a base model scaled with S^2 either during pre-training or after pre-training. We pre-train the model on ImageNet-21k, using ViT image classification as the pre-training objective. We compare both models with single-scale base and large model. Results are shown in Tab. 4. We can see that when the base model is trained with single image scale and only scaled to multiple image scales after pre-training, it has sub-optimal performance compared to large model, which aligns with our observation in Sec. 3.2. However, when adding S^2 into pre-training, multi-scale base model is able to outperform the large model, which confirms smaller models pre-trained with S^2 can match the advantage of larger models.

5 Discussion

In this work, we ask the question *is a larger model always necessary for better visual understanding?* We find that scaling on the dimension of image scales, which we call Scaling on Scales (S^2), instead of model size usually obtains better performance on a wide range of downstream tasks including image classification, semantic segmentation, depth estimation, as well as on MLLM benchmarks and robotic manipulation tasks. We find that although larger models generalize better on rare or hard instances compared to smaller models, smaller models with S^2 can learn most of the representation learned by larger models, and pre-training smaller models with S^2 can further improve the performance and match or even exceed the advantage of larger models. S^2 has a few implications for future work, including **(i) scale-selective processing**, *i.e.*, not every scale at every position in an image contains equally useful features, and depending on image content and high-level task, it is much more efficient to select certain scales to process for each region, which resembles the bottom-up and top-down selection mechanism in human visual attention [33,58,84], **(ii) parallel processing of single image**, *i.e.*, in contrast with regular ViT where the whole image is processed together at once, the fact that each sub-image is processed independently in S^2 enables parallel processing of different sub-images for a single image, which is especially helpful for scenarios where latency on processing single large image is critical [82].

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., Zhou, J.: Qwen-vl: A frontier large vision-language model with versatile abilities. arXiv preprint arXiv:2308.12966 (2023)
3. Bai, Y., Geng, X., Mangalam, K., Bar, A., Yuille, A.L., Darrell, T., Malik, J., Efros, A.A.: Sequential modeling enables scalable learning for large vision models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22861–22872 (2024)
4. Bartlett, P.L., Long, P.M., Lugosi, G., Tsigler, A.: Benign overfitting in linear regression. Proceedings of the National Academy of Sciences **117**(48), 30063–30070 (2020)
5. Bello, I., Fedus, W., Du, X., Cubuk, E.D., Srinivas, A., Lin, T.Y., Shlens, J., Zoph, B.: Revisiting resnets: Improved training and scaling strategies. Advances in Neural Information Processing Systems **34**, 22614–22627 (2021)
6. Beyer, L., Hénaff, O.J., Kolesnikov, A., Zhai, X., Oord, A.v.d.: Are we done with imagenet? arXiv preprint arXiv:2006.07159 (2020)
7. Bolya, D., Ryali, C., Hoffman, J., Feichtenhofer, C.: Window attention is bugged: How not to interpolate position embeddings. arXiv preprint arXiv:2311.05613 (2023)
8. Brooks, T., Peebles, B., Homes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., Ng, C., Wang, R., Ramesh, A.: Video generation models as world simulators (2024), `https://openai.com/research/video-generation-models-as-world-simulators`
9. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020)
10. Chen, C.F.R., Fan, Q., Panda, R.: Crossvit: Cross-attention multi-scale vision transformer for image classification. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 357–366 (2021)
11. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1290–1299 (2022)
12. Cheng, C., Duchi, J., Kuditipudi, R.: Memorize to generalize: on the necessity of interpolation in high dimensional linear regression. In: Conference on Learning Theory. pp. 5528–5560. PMLR (2022)
13. Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., Jitsev, J.: Reproducible scaling laws for contrastive language-image learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2818–2829 (2023)
14. Chiang, W.L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J.E., et al.: Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. See https://vicuna. lmsys. org (accessed 14 April 2023) (2023)
15. Contributors, M.: MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. `https://github.com/open-mmlab/mmsegmentation` (2020)
16. Dai, W., Li, J., Li, D., Tiong, A.M.H., Zhao, J., Wang, W., Li, B., Fung, P., Hoi, S.: Instructblip: Towards general-purpose vision-language models with instruction tuning (2023)

17. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). vol. 1, pp. 886–893. Ieee (2005)
18. Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., Steiner, A.P., Caron, M., Geirhos, R., Alabdulmohsin, I., et al.: Scaling vision transformers to 22 billion parameters. In: International Conference on Machine Learning. pp. 7480–7512. PMLR (2023)
19. Dollár, P., Appel, R., Belongie, S., Perona, P.: Fast feature pyramids for object detection. IEEE transactions on pattern analysis and machine intelligence **36**(8), 1532–1545 (2014)
20. Dollár, P., Singh, M., Girshick, R.: Fast and accurate model scaling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 924–932 (2021)
21. 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)
22. El-Nouby, A., Klein, M., Zhai, S., Bautista, M.A., Toshev, A., Shankar, V., Susskind, J.M., Joulin, A.: Scalable pre-training of large autoregressive image models. arXiv preprint arXiv:2401.08541 (2024)
23. Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., Feichtenhofer, C.: Multiscale vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6824–6835 (2021)
24. Fang, Y., Wang, W., Xie, B., Sun, Q., Wu, L., Wang, X., Huang, T., Wang, X., Cao, Y.: Eva: Exploring the limits of masked visual representation learning at scale. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19358–19369 (2023)
25. Feldman, V.: Does learning require memorization? a short tale about a long tail. In: Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing. pp. 954–959 (2020)
26. Feldman, V., Zhang, C.: What neural networks memorize and why: Discovering the long tail via influence estimation. Advances in Neural Information Processing Systems **33**, 2881–2891 (2020)
27. 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)
28. Gurari, D., Li, Q., Stangl, A.J., Guo, A., Lin, C., Grauman, K., Luo, J., Bigham, J.P.: Vizwiz grand challenge: Answering visual questions from blind people. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3608–3617 (2018)
29. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16000–16009 (2022)
30. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
31. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021)

32. Hu, R., Debnath, S., Xie, S., Chen, X.: Exploring long-sequence masked autoencoders. arXiv preprint arXiv:2210.07224 (2022)
33. Itti, L., Koch, C.: Computational modelling of visual attention. Nature reviews neuroscience **2**(3), 194–203 (2001)
34. Kondratyuk, D., Yu, L., Gu, X., Lezama, J., Huang, J., Hornung, R., Adam, H., Akbari, H., Alon, Y., Birodkar, V., et al.: Videopoet: A large language model for zero-shot video generation. arXiv preprint arXiv:2312.14125 (2023)
35. Lee, Y., Kim, J., Willette, J., Hwang, S.J.: Mpvit: Multi-path vision transformer for dense prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7287–7296 (2022)
36. Li, B., Wang, R., Wang, G., Ge, Y., Ge, Y., Shan, Y.: Seed-bench: Benchmarking multimodal llms with generative comprehension. arXiv preprint arXiv:2307.16125 (2023)
37. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
38. Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning. arXiv preprint arXiv:2310.03744 (2023)
39. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. arXiv preprint arXiv:2304.08485 (2023)
40. Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., et al.: Mmbench: Is your multi-modal model an all-around player? arXiv preprint arXiv:2307.06281 (2023)
41. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
42. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11976–11986 (2022)
43. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision **60**, 91–110 (2004)
44. Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.W., Galley, M., Gao, J.: Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. arXiv preprint arXiv:2310.02255 (2023)
45. Lukasik, M., Nagarajan, V., Rawat, A.S., Menon, A.K., Kumar, S.: What do larger image classifiers memorise? arXiv preprint arXiv:2310.05337 (2023)
46. Malik, J., Arbeláez, P., Carreira, J., Fragkiadaki, K., Girshick, R., Gkioxari, G., Gupta, S., Hariharan, B., Kar, A., Tulsiani, S.: The three r's of computer vision: Recognition, reconstruction and reorganization. Pattern Recognition Letters **72**, 4–14 (2016)
47. Northcutt, C.G., Athalye, A., Mueller, J.: Pervasive label errors in test sets destabilize machine learning benchmarks. arXiv preprint arXiv:2103.14749 (2021)
48. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
49. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)

50. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog **1**(8), 9 (2019)
51. Radosavovic, I., Shi, B., Fu, L., Goldberg, K., Darrell, T., Malik, J.: Robot learning with sensorimotor pre-training. arXiv preprint arXiv:2306.10007 (2023)
52. Radosavovic, I., Xiao, T., James, S., Abbeel, P., Malik, J., Darrell, T.: Real-world robot learning with masked visual pre-training. In: Conference on Robot Learning. pp. 416–426. PMLR (2023)
53. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125 **1**(2), 3 (2022)
54. Riquelme, C., Puigcerver, J., Mustafa, B., Neumann, M., Jenatton, R., Susano Pinto, A., Keysers, D., Houlsby, N.: Scaling vision with sparse mixture of experts. Advances in Neural Information Processing Systems **34**, 8583–8595 (2021)
55. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. pp. 234–241. Springer (2015)
56. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**, 211–252 (2015)
57. Ryali, C., Hu, Y.T., Bolya, D., Wei, C., Fan, H., Huang, P.Y., Aggarwal, V., Chowdhury, A., Poursaeed, O., Hoffman, J., et al.: Hiera: A hierarchical vision transformer without the bells-and-whistles. arXiv preprint arXiv:2306.00989 (2023)
58. Shi, B., Darrell, T., Wang, X.: Top-down visual attention from analysis by synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2102–2112 (2023)
59. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12. pp. 746–760. Springer (2012)
60. 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)
61. Steiner, A., Kolesnikov, A., Zhai, X., Wightman, R., Uszkoreit, J., Beyer, L.: How to train your vit? data, augmentation, and regularization in vision transformers. arXiv preprint arXiv:2106.10270 (2021)
62. Sun, Q., Fang, Y., Wu, L., Wang, X., Cao, Y.: Eva-clip: Improved training techniques for clip at scale. arXiv preprint arXiv:2303.15389 (2023)
63. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. pp. 6105–6114. PMLR (2019)
64. Tan, M., Le, Q.: Efficientnetv2: Smaller models and faster training. In: International conference on machine learning. pp. 10096–10106. PMLR (2021)
65. Team, G., Anil, R., Borgeaud, S., Wu, Y., Alayrac, J.B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., et al.: Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805 (2023)
66. Team, Q.: Introducing qwen-vl (Jan 2024), `https://qwenlm.github.io/blog/qwen-vl/`
67. Tompson, J., Goroshin, R., Jain, A., LeCun, Y., Bregler, C.: Efficient object localization using convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 648–656 (2015)

68. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023)
69. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., et al.: Deep high-resolution representation learning for visual recognition. IEEE transactions on pattern analysis and machine intelligence **43**(10), 3349–3364 (2020)
70. Wang, W., Lv, Q., Yu, W., Hong, W., Qi, J., Wang, Y., Ji, J., Yang, Z., Zhao, L., Song, X., et al.: Cogvlm: Visual expert for pretrained language models. arXiv preprint arXiv:2311.03079 (2023)
71. Wightman, R., Touvron, H., Jégou, H.: Resnet strikes back: An improved training procedure in timm. arXiv preprint arXiv:2110.00476 (2021)
72. Wu, P., Xie, S.: V*: Guided visual search as a core mechanism in multimodal llms. arXiv preprint arXiv:2312.14135 (2023)
73. Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R.: Detectron2. `https://github.com/facebookresearch/detectron2` (2019)
74. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3733–3742 (2018)
75. Xu, Y., Zhao, S., Song, J., Stewart, R., Ermon, S.: A theory of usable information under computational constraints. arXiv preprint arXiv:2002.10689 (2020)
76. Yang, J., Li, C., Zhang, P., Dai, X., Xiao, B., Yuan, L., Gao, J.: Focal attention for long-range interactions in vision transformers. Advances in Neural Information Processing Systems **34**, 30008–30022 (2021)
77. Yu, J., Xu, Y., Koh, J.Y., Luong, T., Baid, G., Wang, Z., Vasudevan, V., Ku, A., Yang, Y., Ayan, B.K., et al.: Scaling autoregressive models for content-rich text-to-image generation. arXiv preprint arXiv:2206.10789 **2**(3), 5 (2022)
78. Yu, W., Yang, Z., Li, L., Wang, J., Lin, K., Liu, Z., Wang, X., Wang, L.: Mm-vet: Evaluating large multimodal models for integrated capabilities. arXiv preprint arXiv:2308.02490 (2023)
79. Yue, X., Ni, Y., Zhang, K., Zheng, T., Liu, R., Zhang, G., Stevens, S., Jiang, D., Ren, W., Sun, Y., et al.: Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. arXiv preprint arXiv:2311.16502 (2023)
80. Zhai, X., Kolesnikov, A., Houlsby, N., Beyer, L.: Scaling vision transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12104–12113 (2022)
81. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning (still) requires rethinking generalization. Communications of the ACM **64**(3), 107–115 (2021)
82. Zhang, W., He, Z., Liu, L., Jia, Z., Liu, Y., Gruteser, M., Raychaudhuri, D., Zhang, Y.: Elf: accelerate high-resolution mobile deep vision with content-aware parallel offloading. In: Proceedings of the 27th Annual International Conference on Mobile Computing and Networking. pp. 201–214 (2021)
83. Zhao, W., Rao, Y., Liu, Z., Liu, B., Zhou, J., Lu, J.: Unleashing text-to-image diffusion models for visual perception. arXiv preprint arXiv:2303.02153 (2023)
84. Zhaoping, L.: Understanding vision: theory, models, and data. Oxford University Press (UK) (2014)
85. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 633–641 (2017)