AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

Anonymous authors
Paper under double-blind review

ABSTRACT

While the Transformer architecture has become the de-facto standard for natural language processing tasks, its applications to computer vision remain limited. In vision, attention is either applied in conjunction with convolutional networks, or used to replace certain components of convolutional networks while keeping their overall structure in place. We show that this reliance on CNNs is not necessary and a pure transformer can perform very well on image classification tasks when applied directly to sequences of image patches. When pre-trained on large amounts of data and transferred to multiple recognition benchmarks (ImageNet, CIFAR-100, VTAB, etc.), Vision Transformer attain excellent results compared to state-of-the-art convolutional networks while requiring substantially fewer computational resources to train.

The paper just uses the encoder part of the total transformer architecture

1 INTRODUCTION

Self-attention based architectures, in particular Transformers (Vaswani et al., 2017), have become the model of choice in natural language processing (NLP). The dominant approach is to pre-train on a large text corpus and then fine-tune on a smaller task-specific dataset (Devlin et al., 2019). Thanks to Transformers' computational efficiency and scalability, it has become possible to train models of unprecedented size, with over 100B parameters. With the models and datasets growing, there is still no sign of saturating performance.

In computer vision, however, convolutional architectures remain dominant (Krizhevsky et al., 2012; He et al., 2016). Inspired by NLP successes, multiple works try combining CNN-like architectures with self-attention (Wang et al., 2018; Carion et al., 2020), some replacing the convolutions entirely (Ramachandran et al., 2019; Wang et al., 2020a). The latter models, while theoretically efficient, have not yet been scaled effectively on modern hardware accelerators due to the use of specialized attention patterns. Therefore, in large-scale image recognition, classic ResNet-like architectures are still state of the art (Mahajan et al., 2018; Xie et al., 2020; Kolesnikov et al., 2020).

Inspired by the Transformer scaling successes in NLP, we experiment with applying a standard Transformer directly to images, with the fewest possible modifications. For this, we split an image into patches and provide the sequence of linear embeddings of these patches as an input to a Transformer. Image patches are treated the same way as tokens (words) in an NLP application. We train the model on image classification in supervised fashion.

Such models yield modest results when trained on mid-sized datasets such as ImageNet, achieving accuracies of a few percentage points below ResNets of comparable size. This seemingly discouraging outcome may be expected: Transformers lack some inductive biases inherent to CNNs, such as translation equivariance and locality, and therefore do not generalize well when trained on insufficient amounts of data.

The lack of spatial information that can be retrieved over small datasets is a big downside.
That is wise, the encoder is able to show it's magic over large datasets.

However, the picture changes if we train the models on large datasets (14M-300M images). We find that large scale training trumps inductive bias. Transformers attain excellent results when pre-trained at sufficient scale and transferred to tasks with fewer datapoints. Our Vision Transformer, pre-trained on the JFT-300M dataset, approaches or beats state of the art on multiple image recognition benchmarks, reaching accuracy of 88.36% on ImageNet, 90.77% on ImageNet-ReaL, 94.55% on CIFAR-100, and 77.16% on the VTAB suite of 19 tasks.

2 Related Work

Transformers were proposed by Vaswani et al. (2017) for machine translation, and have since become the state of the art method in many NLP tasks. Large Transformer-based models are often pre-trained on large corpora and then fine-tuned for the task at hand: BERT (Devlin et al., 2019) uses a denoising self-supervised pre-training task, while the GPT line of work uses language modeling as its pre-training task (Radford et al., 2018; 2019; Brown et al., 2020).

Naive application of self-attention to images would require that each pixel attends to every other pixel. With quadratic cost in the number of pixels, this does not scale to realistic input sizes. Thus, to apply Transformers in the context of image generation, several approximations have been tried in the past: Parmar et al. (2018) applied the self-attention only in local neighborhoods for each query pixel instead of globally. Such local multi-head dot-product self attention blocks can completely replace convolutions (Ramachandran et al., 2019; Cordonnier et al., 2020; Zhao et al., 2020). Alternatively, works such as Sparse Transformers (Child et al., 2019) employ scalable approximations to global self-attention in order to be applicable to images. An alternative way to scale attention is to apply it in blocks of varying sizes (Weissenborn et al., 2019), in the extreme case only along individual axes (Ho et al., 2019; Wang et al., 2020a). Many of these specialized attention architectures demonstrate promising results on computer vision tasks, but require complex engineering to be implemented efficiently on hardware accelerators.

There has also been a lot of interest in combining convolutional neural networks (CNNs) with forms of self-attention, e.g. by augmenting feature maps for image classification (Bello et al., 2019) or by further processing the output of a CNN using self-attention, e.g. for object detection (Hu et al., 2018; Carion et al., 2020), video processing (Wang et al., 2018; Sun et al., 2019), image classification (Wu et al., 2020), unsupervised object discovery (Locatello et al., 2020), or unified text-vision tasks (Chen et al., 2020c; Lu et al., 2019; Li et al., 2019).

We are not aware of prior application of Transformers with global self-attention to full-sized images. Closest to our model is iGPT (Chen et al., 2020a), which applies Transformers to image pixels after reducing image resolution and color space. The model is trained in an unsupervised fashion as a generative model, and the resulting representation can then be fine-tuned or probed linearly for classification performance, achieving a maximal accuracy of 72% on ImageNet.

This work adds to the increasing collection of papers that explore image recognition at larger scales than the standard ImageNet dataset. To achieve state-of-the-art results, many papers rely on additional data sources (Mahajan et al., 2018; Touvron et al., 2019; Xie et al., 2020). Sun et al. (2017) study how CNN performance scales with dataset size, and Kolesnikov et al. (2020); Djolonga et al. (2020) perform an empirical exploration of CNN transfer learning from large scale datasets such as ImageNet-21k and JFT-300M, both of which are also the focus of this study.

3 Method

We follow as closely as possible the design of the original Transformer (Vaswani et al., 2017). This intentionally simple setup has the advantage that scalable NLP Transformer architectures – and their efficient implementations – can be used almost out of the box. We aim to show that when scaled appropriately, this approach is sufficient to outperform even the best convolutional neural networks.

3.1 Vision Transformer (ViT)

Our Transformer for images follows the architecture designed for NLP. Figure 1 depicts the setup. The standard Transformer receives as input a 1D sequence of token embeddings. To handle 2D images, we reshape the image $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$ into a sequence of flattened 2D patches $\mathbf{x}_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$. (H, W) is the resolution of the original image and (P, P) is the resolution of each image patch. $N = HW/P^2$ is then the effective sequence length for the Transformer. The Transformer uses constant widths through all of its layers, so a trainable linear projection maps each vectorized patch to the model dimension D (Eq. 1), the output of which we refer to as our patch embeddings.

The general idea is to divide an image into small patches and feed them into the transformer encoder. The attention mechanism is able to reweigh these patches according to the label giben.

Similar to BERT's `[class]` token, we prepend a learnable embedding to the sequence of embedded patches ($\mathbf{z}_0^0 = \mathbf{x}_{\text{class}}$), whose state at the output of the Transformer encoder ($\mathbf{z}_0^L$) serves as the

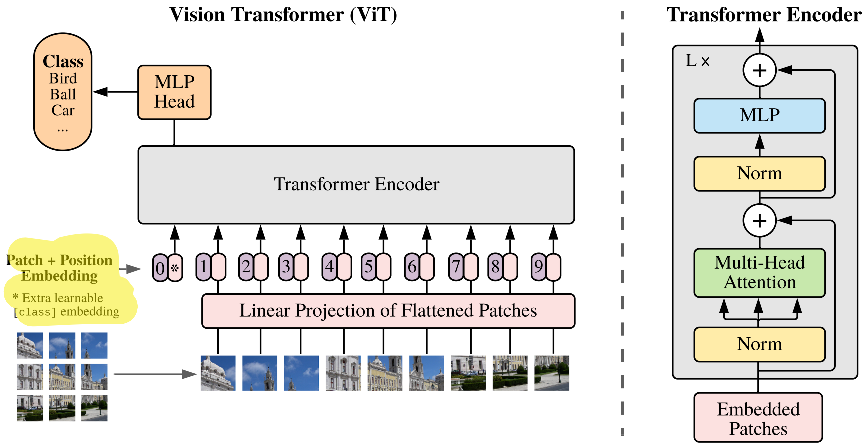

Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings to the resulting sequence of vectors, and feed the patches to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable ”classification token” to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).

image representation $\mathbf{y}$ (Eq. 4). Both during pre-training and fine-tuning, the classification head is attached to $\mathbf{z}_L^0$.

Position embeddings are added to the patch embeddings to retain positional information. We explore different 2D-aware variants of position embeddings (Appendix C.3) without any significant gains over standard 1D position embeddings. The joint embedding serves as input to the encoder.

The encoder part of the actual transformer architecture is used to understand the semantic meaning and relation between the tokens and the decoder was used to use this information to generation new meaningful results. In case of images, the pixel relationships needed to understand the correlation between them is enough to classify them into groups or classes. Thus only the encoder.

The Transformer encoder (Vaswani et al., 2017) consists of alternating layers of multiheaded self-attention (MSA, see Appendix A) and MLP blocks (Eq. 2, 3). Layernorm (LN) is applied before every block, and residual connections after every block (Wang et al., 2019; Baevski & Auli, 2019). The MLP contains two layers with a GELU non-linearity.

$$\begin{aligned}
\mathbf{z}_0 &= [\mathbf{x}_\text{class};\, \mathbf{x}^1_p\mathbf{E};\, \mathbf{x}^2_p\mathbf{E};\cdots;\, \mathbf{x}^N_p\mathbf{E}] + \mathbf{E}_{pos}, && \mathbf{E}\in\mathbb{R}^{(P^2\cdot C)\times D},\ \mathbf{E}_{pos}\in\mathbb{R}^{(N+1)\times D} && (1)\\
\mathbf{z'}_\ell &= \mathrm{MSA}(\mathrm{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, && \ell = 1\ldots L && (2)\\
\mathbf{z}_\ell &= \mathrm{MLP}(\mathrm{LN}(\mathbf{z'}_\ell)) + \mathbf{z'}_\ell, && \ell = 1\ldots L && (3)\\
\mathbf{y} &= \mathrm{LN}(\mathbf{z}^0_L) && && (4)
\end{aligned}$$

Hybrid Architecture. As an alternative to dividing the image into patches, the input sequence can be formed from intermediate feature maps of a ResNet (He et al., 2016). In this hybrid model, the patch embedding projection $\mathbf{E}$ (Eq. 1) is replaced by the early stages of a ResNet. One of the intermediate 2D feature maps of the ResNet is flattened into a sequence, projected to the Transformer dimension, and then fed as an input sequence to a Transformer. The classification input embedding and position embeddings are added as described above to the input to the Transformer.

Since a resnet or any CNN extracts a feature map of the image, the patches can also be fed into a resnet first before been fed to the encoder thus increasing the information gain.

3.2 Fine-tuning and Higher Resolution

Typically, we pre-train ViT on large datasets, and fine-tune to (smaller) downstream tasks. For this, we remove the pre-trained prediction head and attach a zero-initialized $D \times K$ feedforward layer, where K is the number of downstream classes. It is often beneficial to fine-tune at higher resolution than pre-training (Touvron et al., 2019; Kolesnikov et al., 2020). When feeding images of higher resolution, we keep the patch size the same, which results in a larger effective sequence length. The Vision Transformer can handle arbitrary sequence lengths (up to memory constraints), however, the pre-trained position embeddings may no longer be meaningful. We therefore perform 2D interpolation of the pre-trained position embeddings, according to their location in the original image. Note that this resolution adjustment and patch extraction are the only points at which an inductive bias about the 2D structure of the images is manually injected into the Vision Transformer.

Model	Layers	Hidden size D	MLP size	Heads	Params
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

Table 1: Configuration of our different model variants.

4 Experiments

We evaluate the representation learning capabilities of ResNet, Vision Transformer (ViT), and the hybrid. To understand the data requirements of each model, we pre-train on datasets of varying size and evaluate many benchmark tasks. When considering the computational cost of pre-training the model, ViT performs very favourably, attaining state-of-the-art on most recognition benchmarks at a lower pre-training cost. Lastly, we perform a small experiment using self-supervision, and show that self-supervised ViT holds promising for the future.

4.1 Setup

Datasets. To explore model scalability, we use ILSVRC-2012 ImageNet dataset with 1k classes and 1.3M images (we refer to it as ImageNet in what follows), its superset ImageNet-21k with 21k classes and 14M images (Deng et al., 2009), and JFT (Sun et al., 2017) with 18k classes and 303M million high-resolution images. We de-duplicate the pre-training datasets following Kolesnikov et al. (2020). We transfer the models trained on these dataset to many benchmark tasks: ImageNet on the original validation labels, and the cleaned-up ReaL labels (Beyer et al., 2020), CIFAR-10/100 (Krizhevsky, 2009), Oxford-IIIT Pets (Parkhi et al., 2012), and Oxford Flowers-102 (Nilsback & Zisserman, 2008). For these datasets, pre-processing follows Kolesnikov et al. (2020).

We also evaluate on the 19-task VTAB classification suite (Zhai et al., 2019b). VTAB evaluates low-data transfer using 1 000 examples to diverse tasks. The tasks are divided into three groups: *Natural* – tasks like the above, Pets, CIFAR, etc. *Specialized* – medical and satellite imagery, and *Structured* – tasks that require geometric understanding like localization.

Model Variants. We base ViT configurations on those used for BERT (Devlin et al., 2019). We denote our models ViT-L/16 to mean the "Large" variant (in this case), with 16×16 input patch size. Note: the Transformer's sequence length is $\text{image\_res}^2/\text{patch\_size}^2$. Suffixes include "B" (Base), "L" (Large), and "H" (Huge); see Table 1. For the baseline CNNs, we use ResNet, but replace the Batch Norm layers with Group Norm, and used standardized convolutions. These modifications improves transfer (Kolesnikov et al., 2020), and we denote the modified model ResNet (BiT). For the hybrids, we feed the intermediate feature maps into ViT with patch size of one "pixel". To compare different sequence lengths we try (i) taking the output of stage 4 of a regular R50, and (ii) removing stage 4, and placing the same number of layers in stage 3 (so still 50 in total), and taking the output of stage 3. Option (ii) results in a longer sequence length, and a more expensive ViT head.

Training & Fine-tuning. We train all models, including ResNets, using Adam with $\beta_1 = 0.9$, $\beta_2 = 0.999$, a batch size of 4096 and apply a high weight decay of 0.1, which we found to be useful for transfer of all models (Appendix C.1 shows that, in contrast to common practices, Adam works slightly better than SGD for ResNets in our setting). We use a linear learning rate warmup and decay, see Appendix B.1 for details. For fine-tuning we use SGD with momentum, batch size 512, for all models, see Appendix B.1.1. For ImageNet results in Table 2, we fine-tuned at higher resolution: 512 for ViT-L/16 and 518 for ViT-H/14, and the latter also uses Polyak & Juditsky (1992) averaging with a factor of 0.9999 (Ramachandran et al., 2019; Wang et al., 2020b).

Metrics. We report results on downstream datasets either through few-shot or fine-tuning accuracy. Fine-tuning accuracies capture the performance of each model after fine-tuning it on the respective dataset. Few-shot accuracies are obtained by solving a regularized linear regression problem that maps the (frozen) representation of a subset of training images to $\{-1, 1\}^K$ target vectors. Though we mainly focus on fine-tuning performance, we sometimes use linear few-shot accuracies for fast on-the-fly evaluation where fine-tuning would be too costly.

	Ours (ViT-H/14)	Ours (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	88.36	87.61 ± 0.03	87.54 ± 0.02	$88.4/\mathbf{88.5}^{*}$
ImageNet ReaL	**90.77**	90.24 ± 0.03	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.63 ± 0.03	–
VTAB (19 tasks)	**77.16** ± 0.29	75.91 ± 0.18	76.29 ± 1.70	–
TPUv3-days	2.5k	0.68k	9.9k	12.3k

Table 2: Comparison with state of the art on popular image classification datasets benchmarks. Vision Transformer models pre-trained on the JFT300M dataset often match or outperform ResNet-based baselines while taking substantially less computational resources to pre-train. *Slightly improved 88.5% result reported in Touvron et al. (2020).

4.2 Comparison to State of the Art

We first compare our largest models – ViT-H/14 and ViT-L/16 pre-trained on JFT-300M – to state-of-the-art CNNs from the literature. The first comparison point is Big Transfer (BiT) (Kolesnikov et al., 2020), which performs supervised transfer learning with large ResNets. The second is Noisy Student (Xie et al., 2020), which is a large EfficientNet trained using semi-supervised learning on ImageNet and JFT-300M with the labels removed. Currently, Noisy Student is the state of the art on ImageNet and BiT-L on the other datasets reported here. All models we're trained on TPUv3 hardware, and we report the number of TPUv3-days taken to pre-train each.

Table 2 contains the results. The smaller ViT-L/16 model matches or outperforms BiT-L on all datasets, while requiring substantially less computational resources to train. The larger model, ViT-H-14, further improves the performance, especially on the more challenging datasets – ImageNet and CIFAR-100, and the VTAB suite. It matches or exceeds state of the art on all of the datasets, in some cases by a substantial margin (e.g. 1% on CIFAR-100). On ImageNet, ViT is roughly 0.1% below Noisy Student with the standard noisy labels, but beats state of the art when evaluated on the cleaner ReaL labels. Interestingly, our models took substantially less compute to pre-train than prior state of the art, however, we note that pre-training efficiency may be affected not only by the architecture choice, but also other parameters, such as training schedule, optimizer, weight decay, etc. We provide a controlled study of performance vs. compute for different architectures in Section 4.4.

Figure 2 decomposes the VTAB tasks into their respective groups, and compares to previous SOTA methods on this benchmark: BiT, VIVI – a ResNet co-trained on ImageNet and Youtube (Tschannen et al., 2020), and S4L – supervised plus semi-supervised learning on ImageNet (Zhai et al., 2019a). On the *Natural* tasks, ViT-H/14 is slightly outperformed by BiT-R152x4, although the difference is within repetition noise. On *Specialized* ViT just outperforms BiT (and other methods), but the largest benefit appears to be the *Structured* task group where ViT is significantly superior.

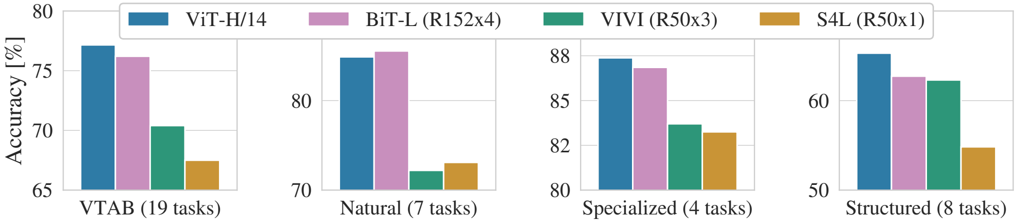

Figure 2: Breakdown of VTAB performance in *Natural*, *Specialized*, and *Structured* task groups.

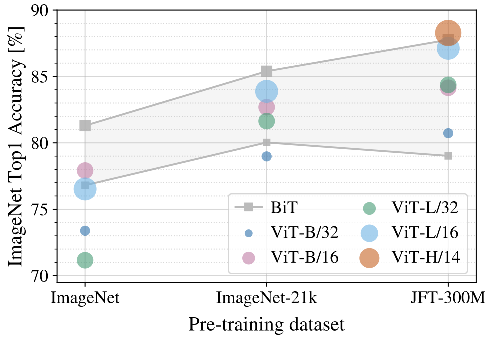

Figure 3: Transfer to ImageNet. While large ViT models perform worse than BiT ResNets (shaded area) when pre-trained on small datasets, they shine when pre-trained on larger datasets. Similarly, larger ViT variants overtake smaller ones as the dataset grows.

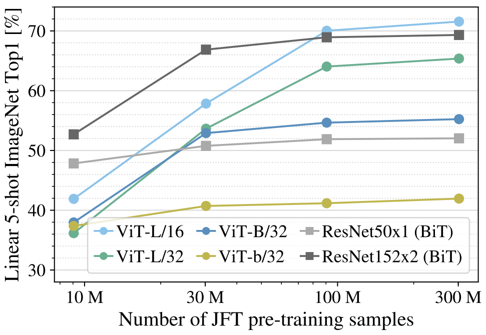

Figure 4: Linear few-shot evaluation on ImageNet versus pre-training size. ResNets perform better with smaller pre-training datasets but plateau sooner than ViT which performs better with larger pre-training. ViT-b is ViT-B with all hidden dimensions halved.

4.3 Pre-training Data Requirements

The Vision Transformer performs well when pre-trained on a large JFT-300M dataset. With fewer inductive biases for vision than ResNets, how crucial is the dataset size? We perform two series of experiments.

First, we pre-train ViT models on datasets of increasing size: ImageNet, ImageNet-21k, and JFT-300M. To get the best possible performance on the smaller datasets, we optimize three regularization parameters – weight decay, dropout, and label smoothing. Figure 3 shows the results after fine-tuning to ImageNet (other datasets in Table 3).[1] Figure 3 contains the results of fine-tuning on ImageNet. When pre-trained on the smallest dataset, ImageNet, ViT-Large models underperform compared to ViT-Base models, despite heavy regularization. However, with ImageNet-21k pre-training, their performances are similar. Only with JFT-300M, do we see the benefit of larger models. Figure 3 also shows the performance region spanned by BiT models of different sizes. The BiT CNNs outperform ViT on ImageNet (despite regularization optimization), but with the larger datasets, ViT overtakes.

Second, we train our models on random subsets of 9M, 30M, and 90M as well as the full JFT-300M dataset. We do not perform additional regularization on the smaller subsets and use the same hyper-parameters for all settings. This way, we assess the intrinsic model properties, and not the effect of regularization. We do, however, use early-stopping, and report the best validation accuracy achieved during training. To save compute, we report few-shot linear accuracy instead of full fine-tuning accuracy. Figure 4 contains the results. Vision Transformers overfit more than ResNets with comparable computational cost on smaller datasets. For example, ViT-B/32 is slightly faster than ResNet50; it performs much worse on the 9M subset, but better on 90M+ subsets. The same is true for ResNet152x2 and ViT-L/16. This result reinforces the intuition that the convolutional inductive bias is useful for smaller datasets, but for larger ones, learning the relevant patterns is sufficient, even beneficial.

Overall, the few-shot results on ImageNet (Figure 4), as well as the low-data results on VTAB (Table 2) seem promising for very low-data transfer. Further analysis of few-shot properties of ViT is an exciting direction of future work.

4.4 Scaling Study

We perform a controlled scaling study of different models. For this, we evaluate transfer performance from JFT-300M. On JFT-300M, data size does not bottleneck the models' performances, and we assess performance versus pre-training cost of each model. The model set includes: 5 ResNets,

[1] Note that the ImageNet pre-trained models are also fine-tuned, but again on ImageNet. This is because the resolution increase during fine-tuning improves the performance.

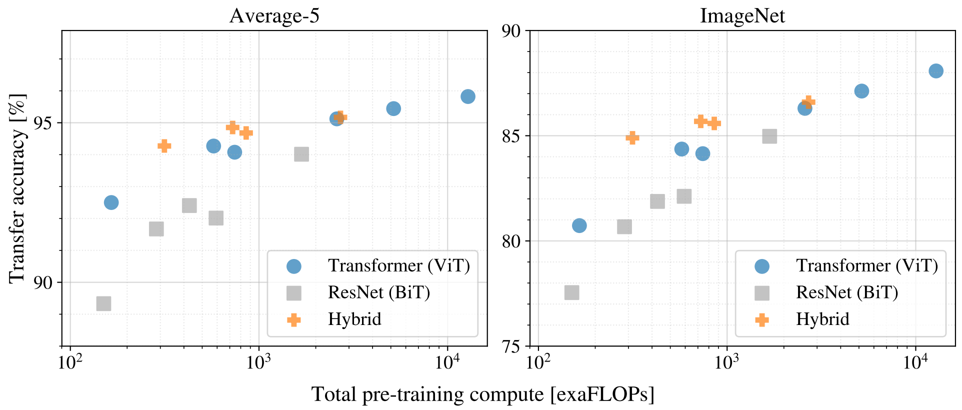

Figure 5: Performance versus cost for different architectures: Vision Transformers, ResNets, and hybrids. Vision Transformers generally outperform ResNets with the same computational budget. Hybrids improve upon pure Transformers for smaller model sizes, but the gap vanished for larger models.

R50x1, R50x2 R101x1, R152x1, R152x2, pre-trained for 7 epochs; 6 Vision Transformers, ViT-B/32, B/16, L/32, L/16, pre-trained for 7 epochs, plus L/16 H/14 pre-trained for 14 epochs; and 4 hybrids, R50 + ViT-B/32, B/16, L/32, L/16 pre-trained for 7 epochs. Figure 5 contains the transfer performance versus total pre-training compute (see Appendix C.4 for details on computational costs computation).

A few patterns can be observed. First, Vision Transformer dominate ResNets on the performance/compute trade-off. ViT uses approximately 2× less compute to attain the same performance (average 5 datasets). Second, hybrids slightly outperform ViT at small computational budgets, but the difference vanishes for larger ones. This result was surprising, since one might expect convolutional local feature processing to assist the ViT at any size. Third, Vision Transformers appear not to saturate within the range tried, motivating future scalability efforts.

4.5 Inspecting Vision Transformer

To begin to understand how the Vision Transformer processes image data, we analyze its internal representations. The first layer of the Vision Transformer linearly projects the flattened patches into a lower-dimensional space (Eq. 1). Figure 7 (left) shows the top principal components of the the learned embedding filters. The components resemble plausible basis functions for a low-dimensional representation of the fine structure within each patch.

After the projection, a learned position embedding is added to the patch representations. Figure 7 (center) shows that the model learns to encode distance within the image in the similarity of position embeddings, i.e. closer patches tend to have more similar position embeddings. Further, the row-column structure appears; patches in the same row/column have similar embeddings. Finally, a sinusoidal structure is sometimes apparent for larger grids (Appendix C). That the position embeddings learn to represent 2D image topology explains why hand-crafted 2D-aware embedding variants do not yield improvements (Appendix C.3).

Self-attention allows ViT to integrate information across the entire image even in the lowest layers. We investigate to what degree the network makes use of this capability. Specifically, we compute the average distance in image space across which information is integrated, based on the attention weights (Figure 7, right). This "attention distance" is analogous to receptive field size in CNNs.

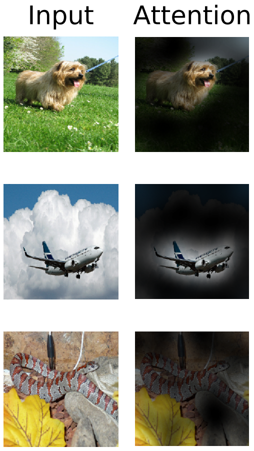

Figure 6: Representative examples of attention from the output token to the input space. See Appendix C.6 for details.

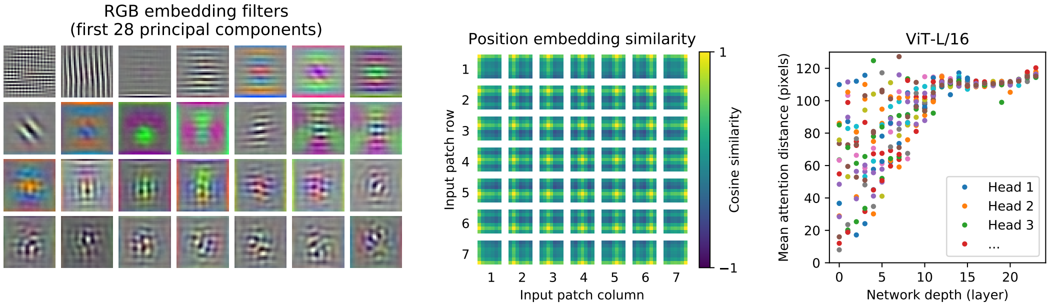

Figure 7: **Left:** Filters of the initial linear embedding of RGB values of ViT-L/32. **Center:** Similarity of position embeddings of ViT-L/32. Tiles show the cosine similarity between the position embedding of the patch with the indicated row and column and the position embeddings of all other patches. **Right:** Size of attended area by head and network depth. Each dot shows the mean attention distance across images for one of 12 heads at one layer. See Appendix C.6 for details.

We find that some heads attend to most of the image already in the lowest layers, showing that the ability to integrate information globally is indeed used by the model. Other attention heads have consistently small attention distances in the low layers. This highly localized attention is less pronounced in hybrid models that apply a ResNet before the Transformer (Figure 7, right), suggesting that it may serve a similar function as early convolutional layers in CNNs. Further, the attention distance increases with network depth. Globally, we find that the model attends to image regions that are semantically relevant for classification (Figure 6).

4.6 Self-supervision

Transformers show impressive performance on NLP tasks. However, much of their success stems not only from their excellent scalability but also from large scale self-supervised pre-training (Devlin et al., 2019; Radford et al., 2018). We also perform a preliminary exploration on *masked patch prediction* for self-supervision, mimicking the masked language modeling task used in BERT. With self-supervised pre-training, our smaller ViT-B/16 model achieves 79.9% accuracy on ImageNet, a significant improvement of 2% to training from scratch, but still 4% behind supervised pre-training. Appendix B.1.2 contains further details. We leave exploration of contrastive pre-training (Chen et al., 2020b; He et al., 2020; Bachman et al., 2019; Hénaff et al., 2020) to future work.

5 Conclusion

We have explored the direct application of Transformers to image recognition. Unlike prior works using self-attention in computer vision, we do not introduce any image-specific inductive biases into the architecture. Instead, we interpret an image as a sequence of patches and process it by a standard Transformer encoder as used in NLP. This simple, yet scalable, strategy works surprisingly well when coupled with pre-training on large datasets. Thus, Vision Transformer matches or exceeds the state of the art on many image classification datasets, whilst being relatively cheap to pre-train.

While these initial results are encouraging, many challenges remain. One is to apply ViT to other computer vision tasks, such as detection and segmentation. Our results, coupled with those in Carion et al. (2020), indicate the promise of this approach. Another – to continue exploring self-supervised pre-training methods. Our initial experiments show improvement from self-supervised pre-training, but there is still large gap between self-supervised and large-scale supervised pre-training. Finally – to further scale ViT, given that the performance does not seem yet to be saturating with the increased model size.