Group 14

Paper presentation

Transformers for Image Recognition at Scale

ViT : Vision Transformer

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby
GoogleResearch, Brain Team

Published as a conference paper at ICLR2021

Overview

1. Transformer
2. Vision Transformer
3. Methods
4. Experiments
5. Conclusion

Transformer

Attention is All You Need

- Transformer Architecture was introduced in 2017
- A major model in the NLP field.
- Based on a self-attention mechanism
- Pre-train + Fine-tune

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

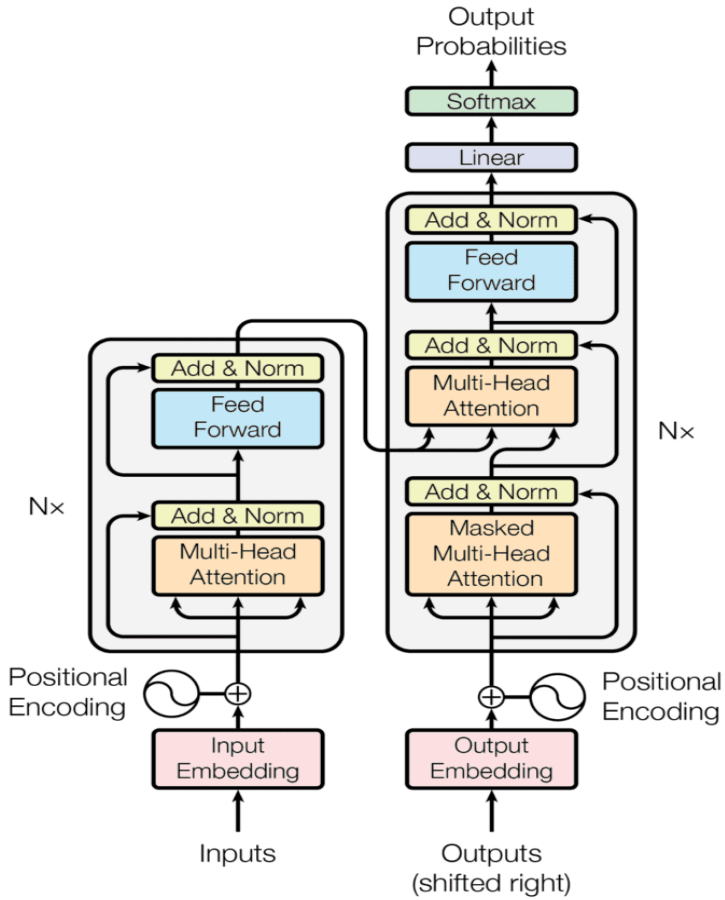
Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com



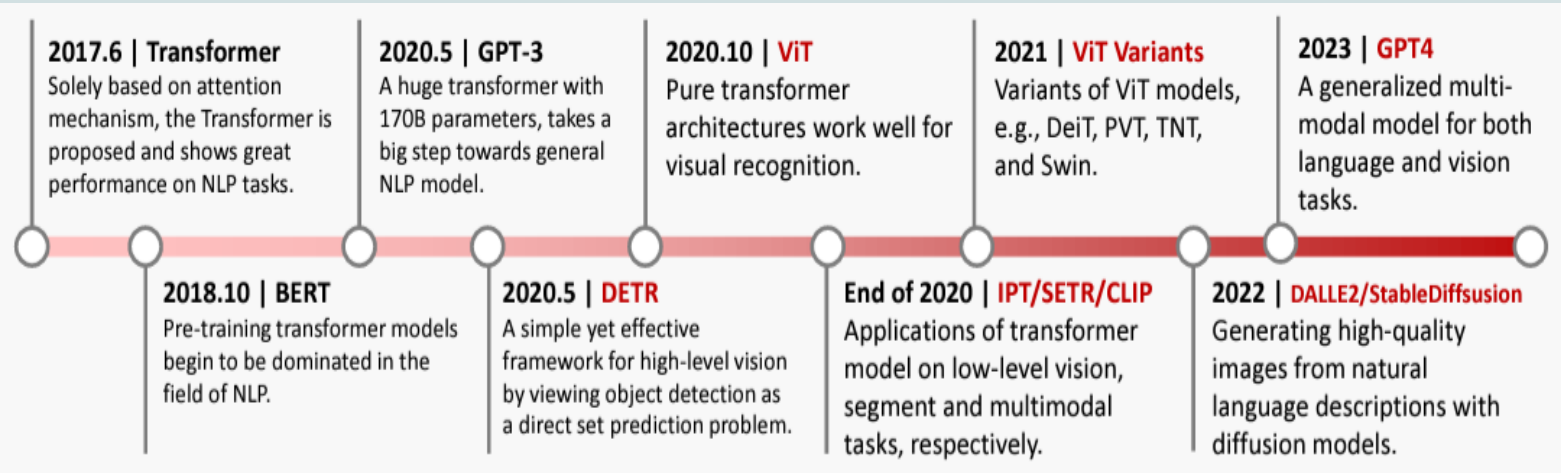
Transformer's success in NLP

Transformer Architecture

- Computational efficiency and scalability
- Train models with large parameters $>$ 100 billion

Transformer development timeline

- Since Transformer was released in 2017, many models based on Transformers have been released in the NLP and Computer Vision areas.



. . .

Transformer in Computer Vision

How to apply self-attention to CNN?

- Non-local neural networks (Wang et al.,2018)
- Stand-alone self-attention in vision models (Ramachandran et al.,2019)
- Axial-DeepLab (Wang et al.,2020)

⋮

Let's use the Transformer model itself

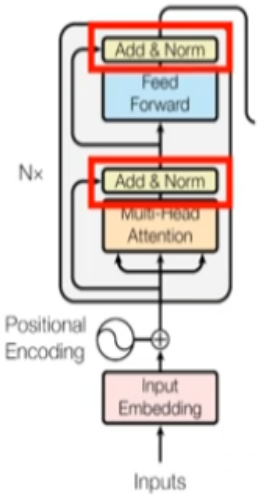
- Vision Transformer (Dosovitskiy et al.,2020)
- Data efficient image Transformer (Touvron et al.,2020)
- TransGAN (Jiang et al.,2021)

⋮

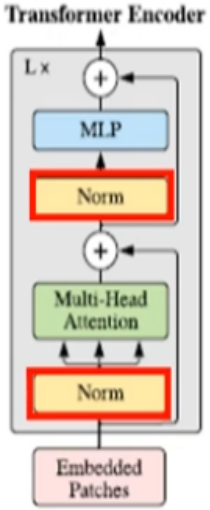
Vision Transformer

What is vision transformer?

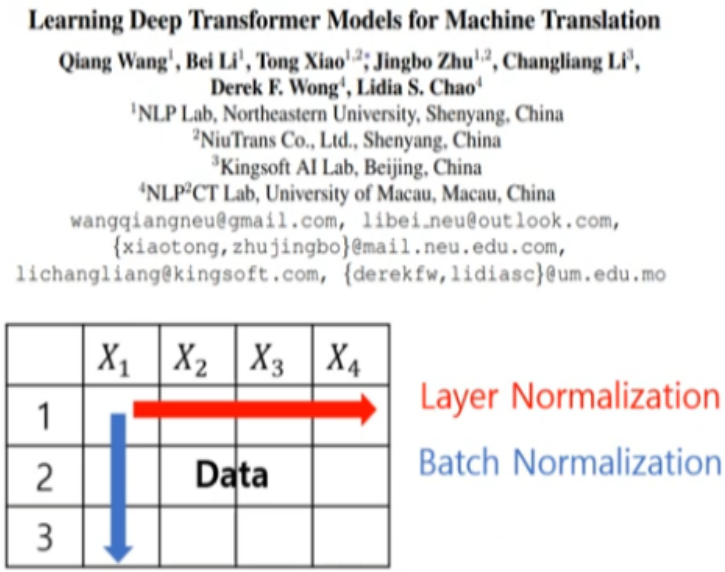
- Apply Transformer directly to image classification task with large image patches
- Based on subsequent studies, the ViT architecture uses a modified architecture that shifts the position of the normalized layer
- State-of-art performance in widely used image recognition tasks



"Vanilla" Transformer



"ViT" Transformer



Methods

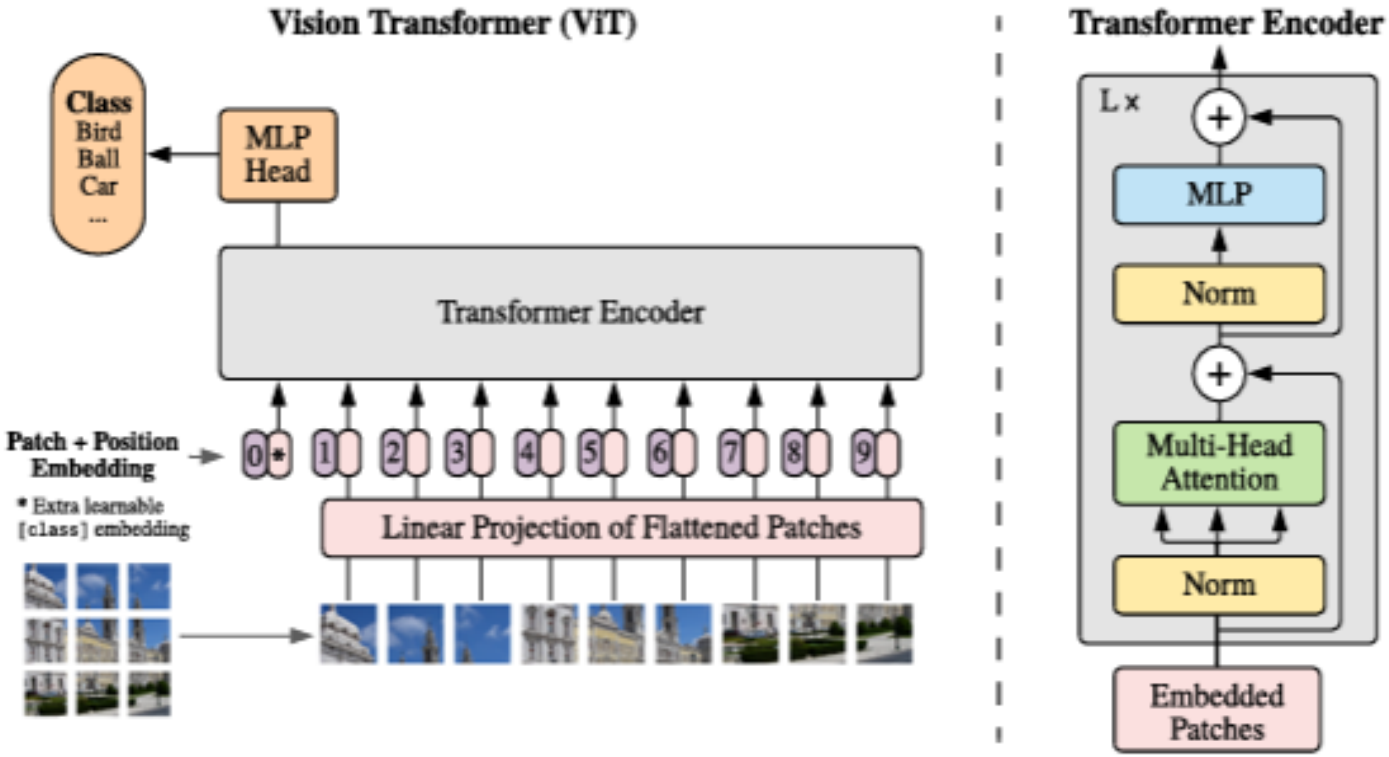


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable "classification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

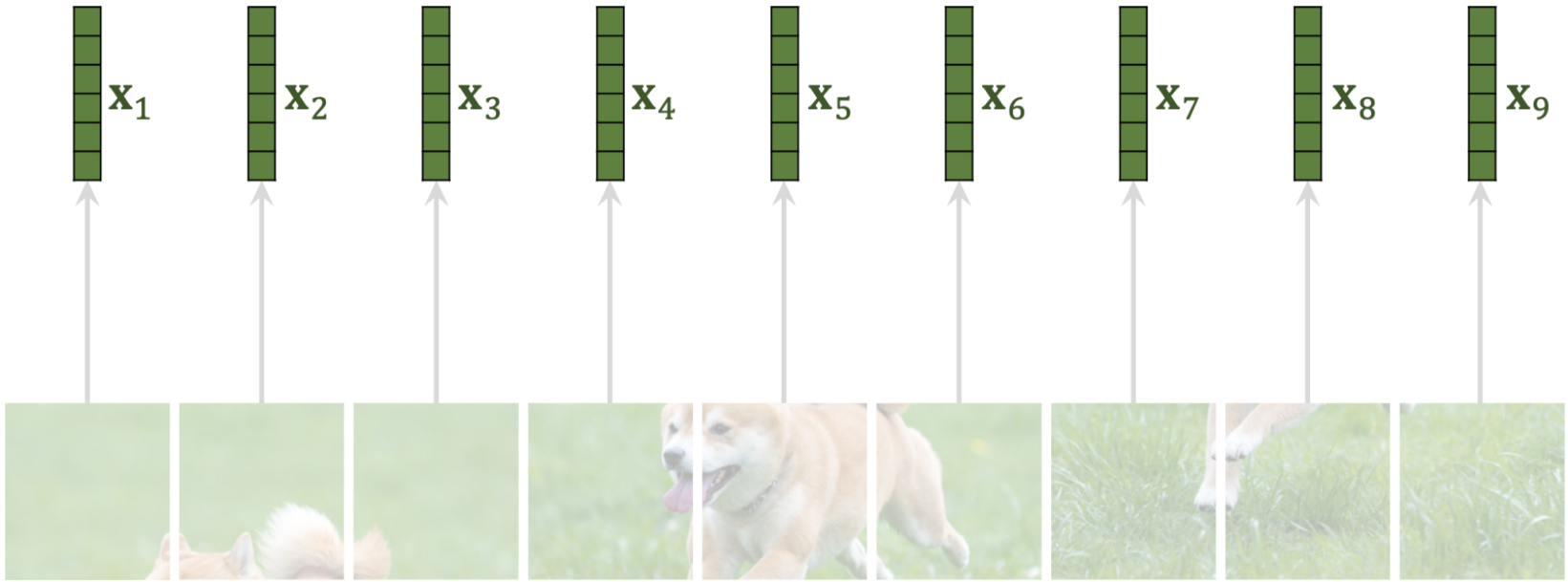
Methods

Step1: Image Patching

- $x \in \mathbb{R}^{H \times W \times C} \rightarrow x_P \in \mathbb{R}^{N \times (P^2 \cdot C)}$
  where (H, W) is original image size, C: channels. (P, P) is patch resolution, number of patches $N = H \cdot W / P^2$
- $224 \times 224 \times 3 \rightarrow 16 \times 16 \times 3$

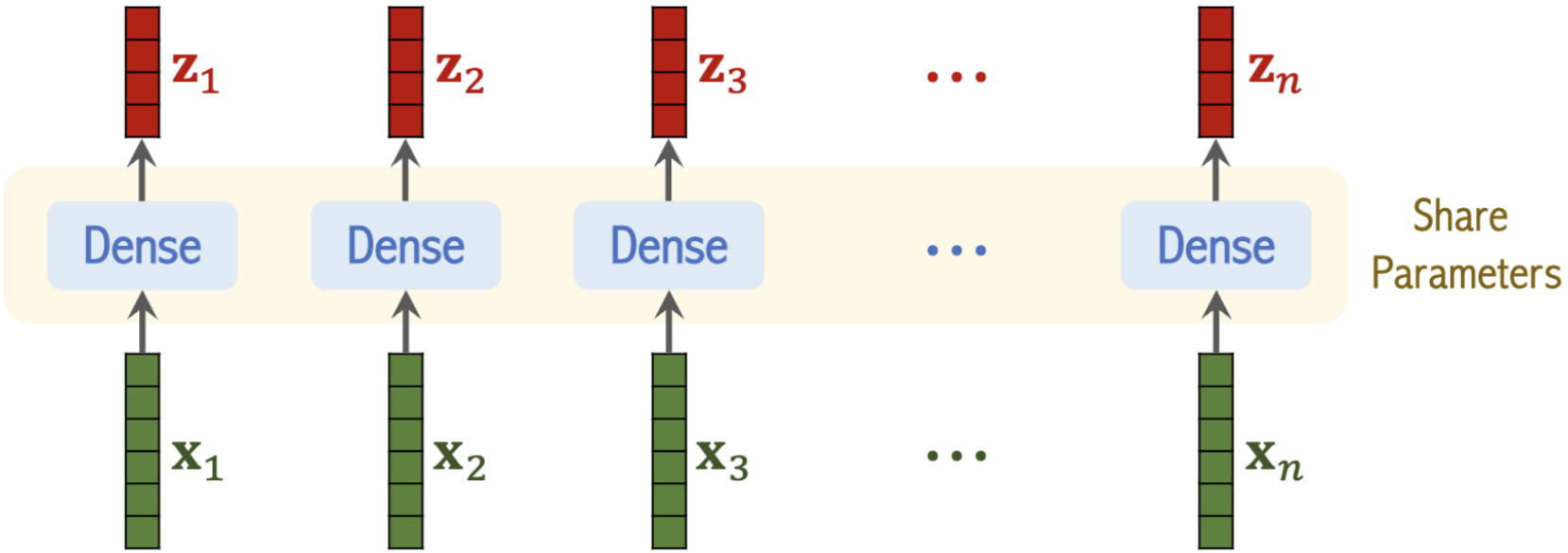
Step2: Patch Flatten

- Flatten the patches into vectors
- $16 \times 16 \times 3 \rightarrow 768 \times 1$



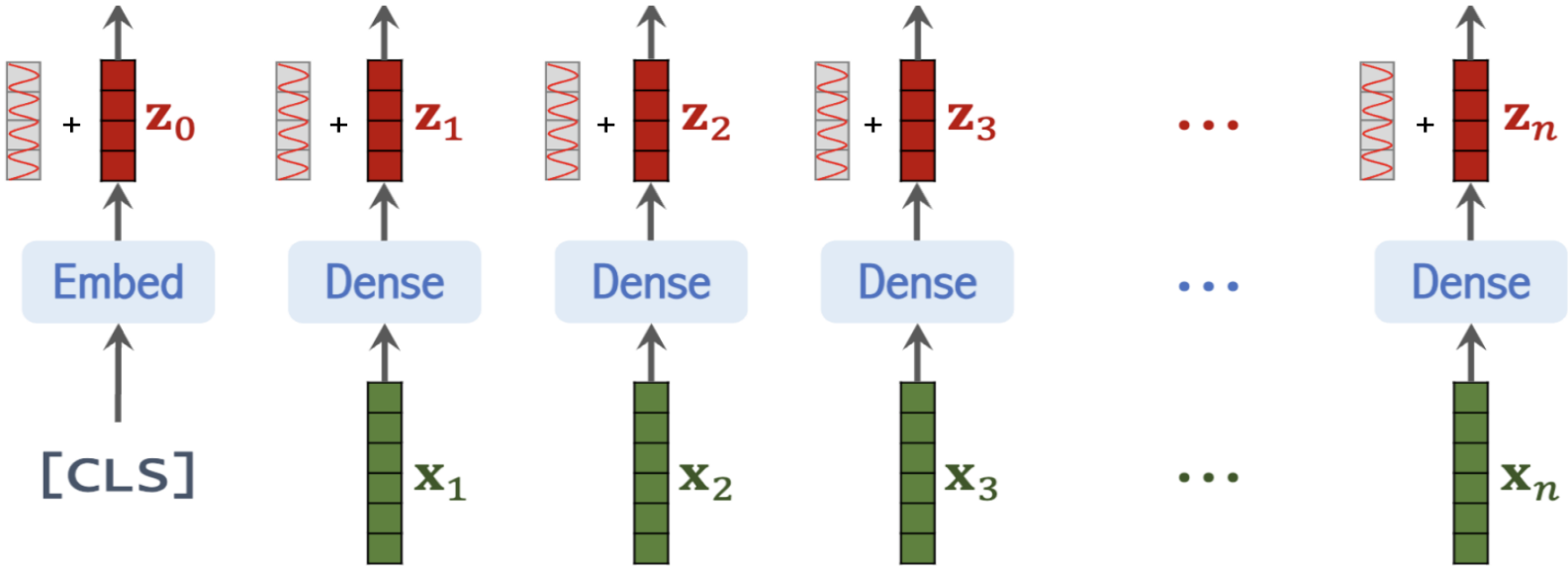
Step3: Patch Embedding

- Map to D dimensions by passing through a trainable linear projection layer.
- $z = \left[\mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; ...; \mathbf{x}_p^n\mathbf{E}\right], \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}$
- Linear projection to D-dimensional vector



Step4: Positional Encoding

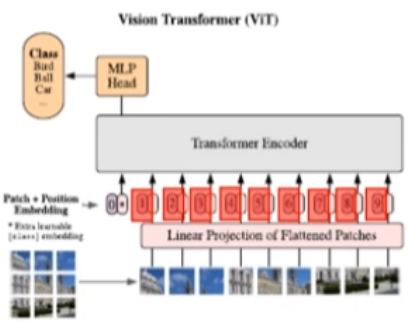
- positional information is added to the patch embedding, also add CLS token in the front.
- $z_0 = \left[\mathbf{x}_{cls}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; ...; \mathbf{x}_p^n\mathbf{E}\right] + \mathbf{E}_{pos}, \quad \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}, \mathbf{E}_{pos} \in \mathbb{R}^{(N+1) \times D}$

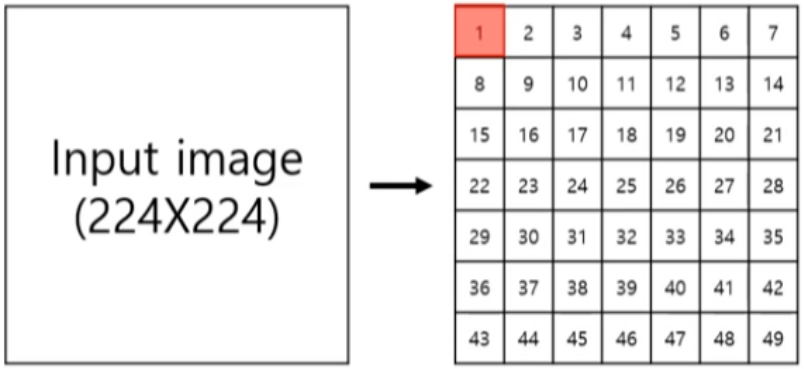


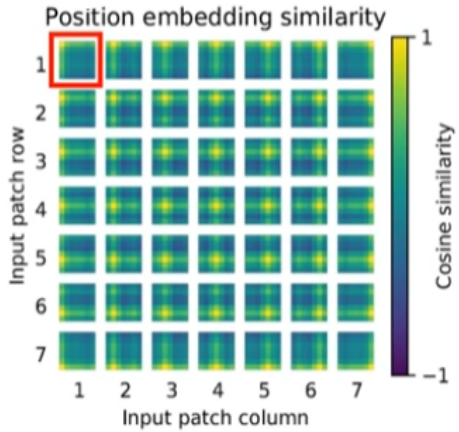
Method

Position embedding(ViT-L/32)

- Similarity of position embeddings of ViT-L/32
- Tiles show the cosine similarity between the position embedding of the patch







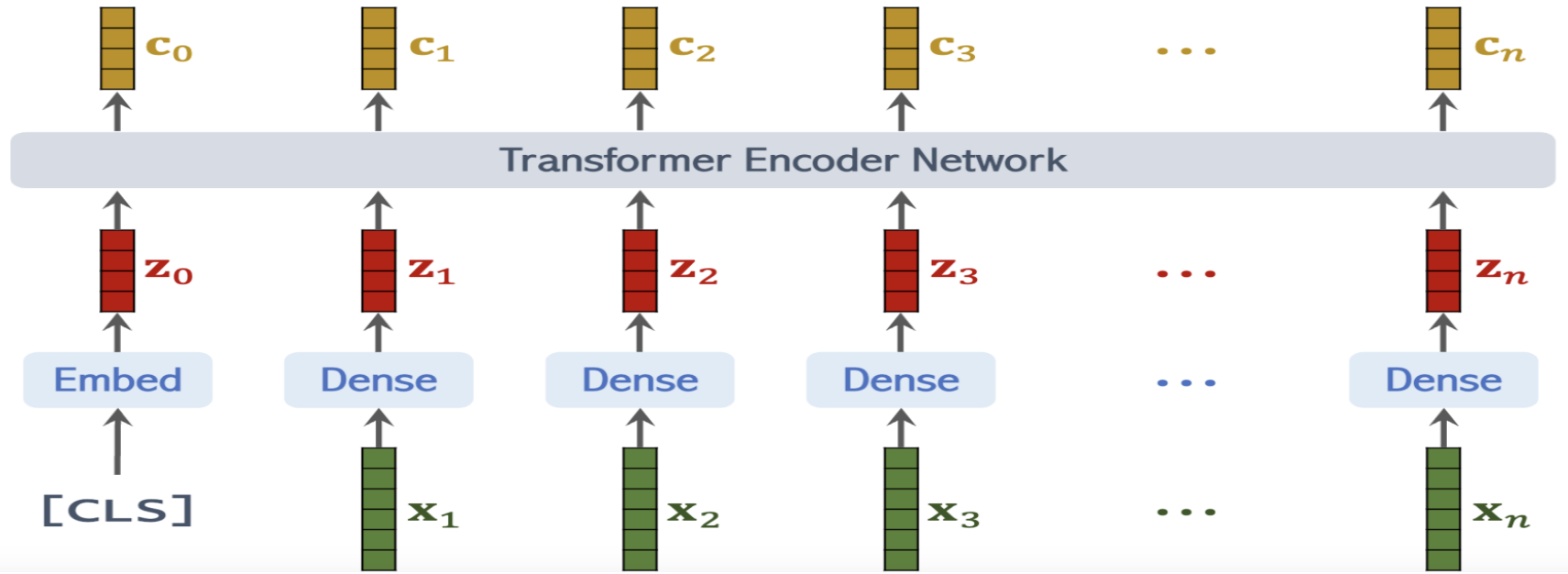
Positional Encoding

- No Positional Embedding
- 1-D Positional Embedding
- 2-D Positional Embedding
- Relational Positional Embedding

Pos. Emb.	Default/Stem	Every Layer	Every Layer-Shared
No Pos. Emb.	0.61382	N/A	N/A
1-D Pos. Emb.	0.64206	0.63964	0.64292
2-D Pos. Emb.	0.64001	0.64046	0.64022
Rel. Pos. Emb.	0.64032	N/A	N/A

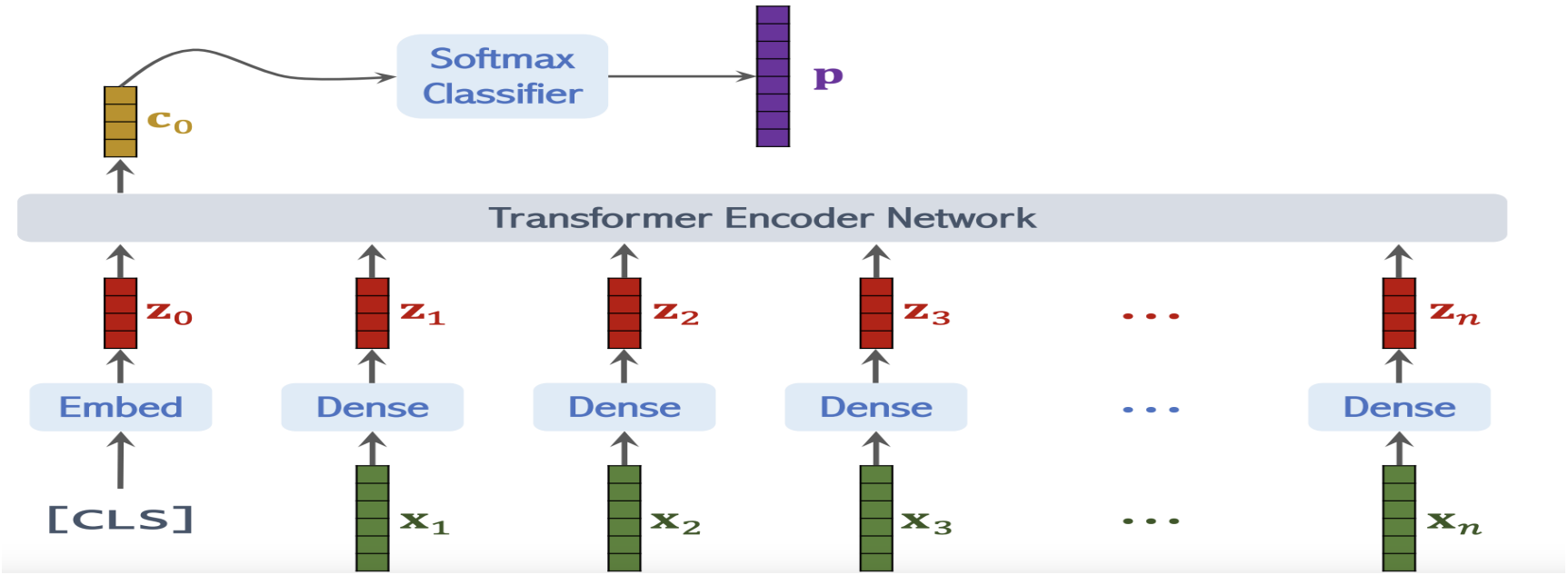
Step5: Transformer Encoder

- Consist of multi-headed self-attention(MSA). Layernorm(LN) is applied. Residential connections after every block. And stack L times.
- $\mathbf{z}'_\ell = \text{MSA}(\text{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \quad \ell = 1, \ldots, L$
- $\mathbf{z}_\ell = \text{MLP}(\text{LN}(\mathbf{z}'_\ell)) + \mathbf{z}'_\ell, \quad \ell = 1, \ldots, L$



Step6: Classification Head

- Connect the classification head with a simple classifier to get the prediction.
- $y = \text{LN}(\mathbf{z}_L^0)$



Methods

Inductive Bias

- ViT has much less image-specific inductive bias than CNNs.

Hybrid Architecture

- Alternative to raw image patches, the input sequence can be formed from feature maps of a CNN
- Patch embedding projection **E** is applied to patches extracted from a CNN feature map

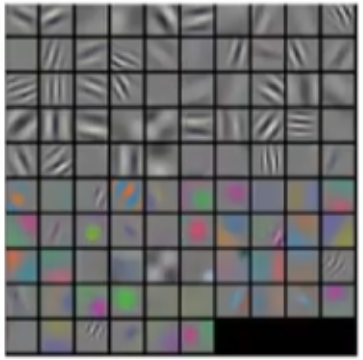
Fine-tuning and Higher Resolution

- Pre-train ViT on large datasets
- Fine-tune to (smaller) downstream tasks

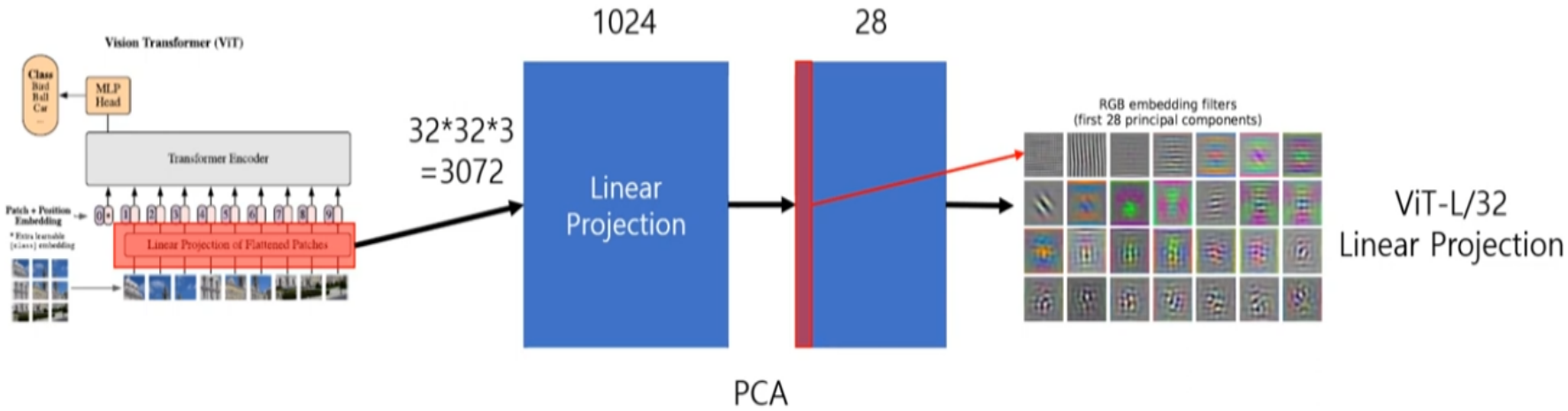
Method - Explanation

Linear Projection (ViT-L/32)

- Filters of the initial linear embedding of RGB values
- Similar to CNN's convolutional filter



CNN convolution filter



Experiments

- Evaluate the representation learning capabilities (ResNet, ViT, Hybrid)
- Pre-train varying size
- Computational cost is lower than others

Experiments - setup

Datasets

- VTAB classification suite - Natural, Specialized, structured.

Model variance

- Base is BERT. Baseline CNN is ResNet, replace BN $\rightarrow$ GN

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: Details of Vision Transformer model variants.

Compare

- ViT-H/14 and ViT-L/16 VS State of the art CNNs

Metrics

- Fine-tuning accuracies: Capture the performance of each model after fine-tuning
- Few-shot accuracies: Few-shot accuracies are obtained by solving a regularized least-squares regression problem that maps the (frozen) representation of a subset of training images to $\{1, 1\}^K$ target vectors.

Experiments - Comparision to state of the art

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Table 2: Comparison with state of the art on popular image classification benchmarks. We report mean and standard deviation of the accuracies, averaged over three fine-tuning runs. Vision Transformer models pre-trained on the JFT-300M dataset outperform ResNet-based baselines on all datasets, while taking substantially less computational resources to pre-train. ViT pre-trained on the smaller public ImageNet-21k dataset performs well too. *Slightly improved 88.5% result reported in Touvron et al. (2020).

Experiments - Comparision to state of the art

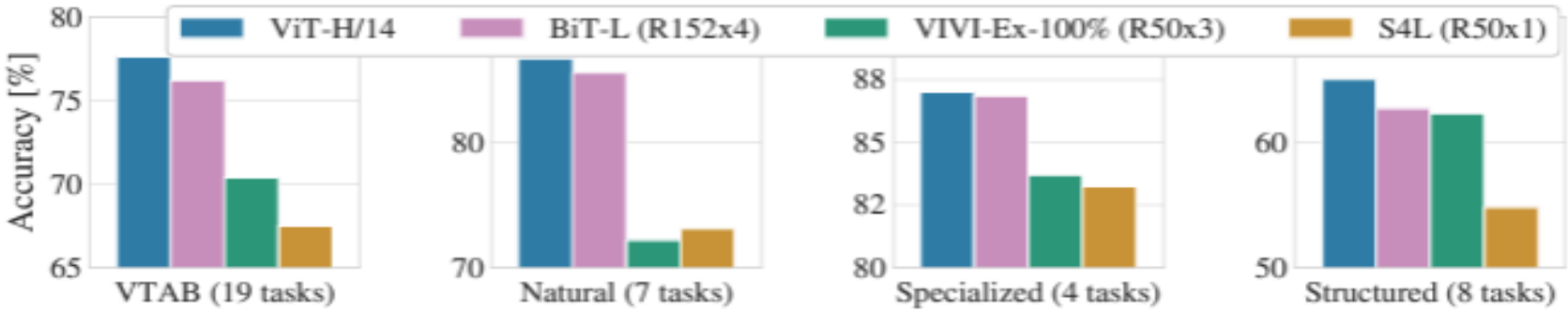


Figure 2: Breakdown of VTAB performance in *Natural*, *Specialized*, and *Structured* task groups.

Big Transfer(BiT)

- Supervised transfer learning with large ResNet

Noisy Student

- which is a large EfficientNet trained using semi-supervised learning on ImageNet and JFT300M with the labels removed.

Experiments - Pre-training Data Requirements

Pre-trained with a large data set

- ViT performs well when pre-trained an a large JFT-300M data set
- With fewer inductive biases for vision than ResNet

How crucial is the data size? - two experiments

1. Pre-train ViT models on data sets of increasing size: ImageNet, ImageNet-21K, and JFT-300M.
2. Train models on random subsets

Experiments - Pre-train ViT models on data sets of increasing size

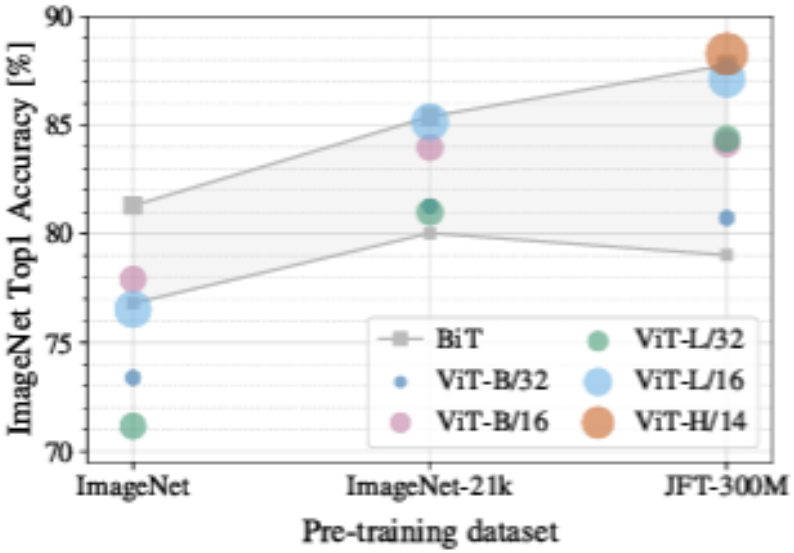


Figure 3: Transfer to ImageNet. While large ViT models perform worse than BiT ResNets (shaded area) when pre-trained on small datasets, they shine when pre-trained on larger datasets. Similarly, larger ViT variants overtake smaller ones as the dataset grows.

Pre-train ViT models on data sets of increasing size

- To boost the performance on the smaller datasets, optimize three basic regularization parameters. - Weight decay, Dropout, Label smoothing

Experiments -Train models on random subsets

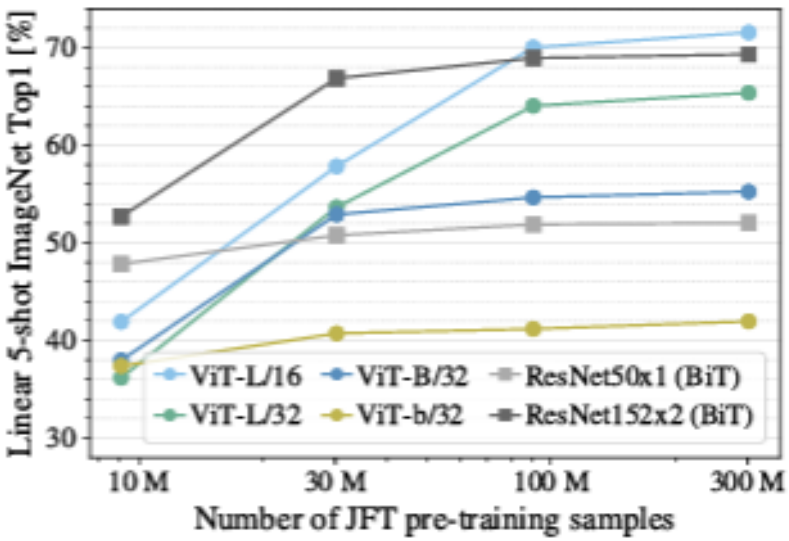


Figure 4: Linear few-shot evaluation on ImageNet versus pre-training size. ResNets perform better with smaller pre-training datasets but plateau sooner than ViT, which performs better with larger pre-training. ViT-b is ViT-B with all hidden dimensions halved.

Train models on random subsets

- Random subsets of 9M, 30M, and 90M as well as full JFT-300M dataset.
- No additional regulaization
- To save compute, we report few-shot linear accuracy instead of full fine-tuning accuracy.
- ViT overfit more than ResNets with comparable computational cost on smaller datasets.
- Convolutional inductive bias is useful for smaller datasets, but for larger ones, learning the relevant patterns directly form data is sufficient even beneficial.

Experiments -Scaling Study

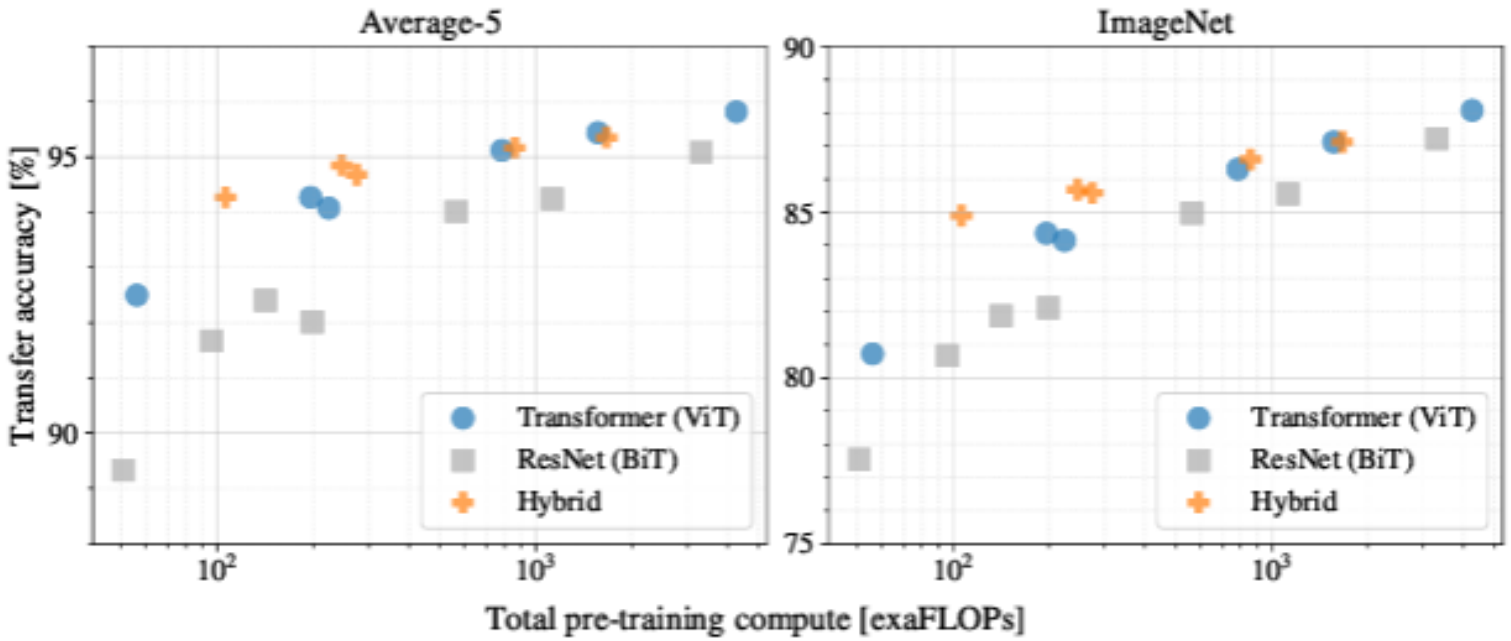


Figure 5: Performance versus pre-training compute for different architectures: Vision Transformers, ResNets, and hybrids. Vision Transformers generally outperform ResNets with the same computational budget. Hybrids improve upon pure Transformers for smaller model sizes, but the gap vanishes for larger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

Summary

Current trends in Vit using Image processing

- Current Trends in Image Processing via ViT :
- Augmented and Virtual Reality: ViTs could potentially enhance the way AR/VR systems process complex scenes.
- Medical Imaging: ViTs are increasingly used in medical imaging for tasks like tumor detection, segmentation, and diagnosis assistance
- Autonomous vehicles/self-driving cars :
  Image processing in autonomous vehicles involves real-time analysis for object detection, classification, and decision-making.
- ViTs may improve the performance of systems in handling diverse and complex driving scenarios.

Summary

Summary

- Vision Transformers have been a super hot topic the past 1-2 years!
- Very different architecture vs traditional CNNs
- Applications to all tasks: classification, detection, segmentation, etc
- Vison transformers are an evolution, not a revolution.
- We can still fundamentally solve the same problems as with CNNs.
- Matrix multiply is more hardwarefriendly than convolution, so ViTs with same FLOPs as CNNs can train and run much faster

Any Questions ?

Thank you!

Reference

Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[J]. arXiv preprint arXiv:2010.11929, 2020.
https://web.eecs.umich.edu/ justincj/slides/eecs498/WI2022/598$_W I2$
https://github.com/wangshusen/DeepLearning

latex Copy code article amsmath

Question:

Consider two infinite, parallel plates, a distance of $2B$ apart as shown in the figure below. Assuming steady state, laminar flow of a Newtonian fluid of constant properties, determine the velocity distribution in the fluid. Plot the velocity distribution schematically on the figure given below.