ViTs vs. CNNs for Image Classification

Student ID: 23089470

Machine Learning (ML) is a way for computers to learn patterns from data, more like how humans learn from experience. Imagine training a pet: when it sits on command and gets a treat, it learns that sitting leads to a reward. Similarly, Machine Learning models learn by recognizing patterns and improving their predictions based on feedback.

Another relatable example is teaching a kid to differentiate between "good" and "bad" behaviour. Parents provide examples, correct mistakes, and reinforce good actions. Likewise, ML algorithms learn by processing labelled data, adjusting their internal rules to make better decisions over time.

Just as a kid or pet that learns faster with clear examples and consistent feedback, ML models improve with high-quality data and effective training methods. Whether classifying images, predicting trends, or recognizing speech, ML systems continuously refine their understanding—just like we do when learning a new skill. Let us dive into our today's tutorial topic: comparing two image classification techniques to better understand which is suitable on which type of data set.

ViT vs CNN:

Image classification has traditionally been dominated by Convolutional Neural Networks (CNNs), which leverage spatial hierarchies through convolutional filters. However, Vision Transformers (ViTs) have emerged as a powerful alternative, utilizing self-attention mechanisms to capture long-range dependencies in images. CNN's are highly efficient at learning local features due to their hierarchical structure, making them well-suited for tasks where spatial locality is crucial. In contrast, ViTs process images as a sequence of patches, allowing them to model global relationships more effectively. Despite their potential, ViTs require larger datasets and computational resources to achieve state-of-the-art performance, whereas CNNs perform well even with limited data.

We will be implementing a model on for both on a similar data set to compare the results and conclude our findings. We will be using CIFAR-10 data set for training and implementation of model, you can find the link to data set here.

Firstly, let us understand how an CNN or ViT works for image classification,

CNN(Convolutional Neural Network)

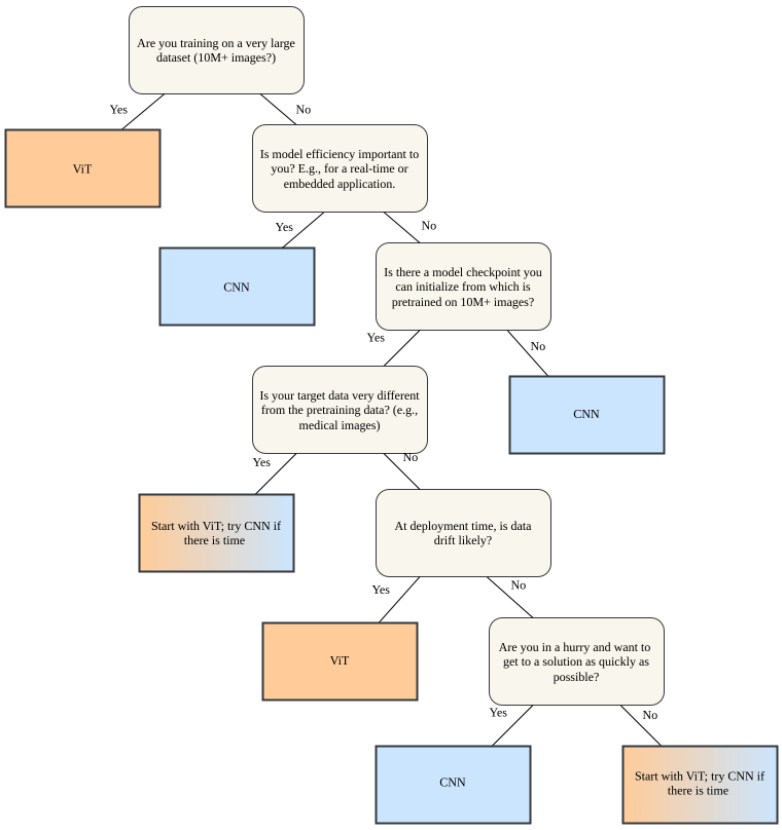
CNNs are deep neural networks designed specifically for image data. They use convolutional layers to extract local features such as edges, textures, and patterns by scanning small regions of an image. CNNs are the backbone of modern computer vision

and have achieved great success in tasks like image classification, object detection, and segmentation. They have many use cases in our day-to-day world.

ViT(Vision Transformer)

vision transformer (**ViT**) Is a transformer designed for computer vision. A ViT decomposes an input image into a series of patches, serializes each patch into a vector, and maps it to a smaller dimension with a single matrix multiplication. These vector embeddings are then processed by a transformer encoder as if they were token embeddings.

ViTs were designed as alternatives to CNNs in computer vision applications. They have different inductive biases, training stability, and data efficiency. Compared to CNNs, ViTs are less data efficient, but have higher capacity. Some of the largest modern computer vision models are ViTs, such as one with 22B parameters.



Source: [Tobias van der Werff]

Let us compare both the models on some parameters

1. Accuracy:

CNNs:

- Excellent accuracy on small and medium datasets (e.g., CIFAR-10, STL-10).
- Strong generalization with fewer parameters due to inductive biases.

ViTs:

- Achieve **state-of-the-art accuracy** on large-scale datasets (e.g., ImageNet-21K, JFT-300M).
- Capture **global relationships** in images better than CNNs

2. Training Time & Computational Cost

CNNs:

- **Efficient training** due to local feature extraction and weight sharing. Convolutions are **faster** and require fewer parameters than self-attention.

Comparison of Convergence Speed:

- **CNNs** converge **faster** and train efficiently on GPUs.
- **ViTs** require **longer training** but generalize well with more data.
- **Pretraining** ViTs on large datasets **significantly speeds up convergence**.

---

3. Interpretability

CNNs:

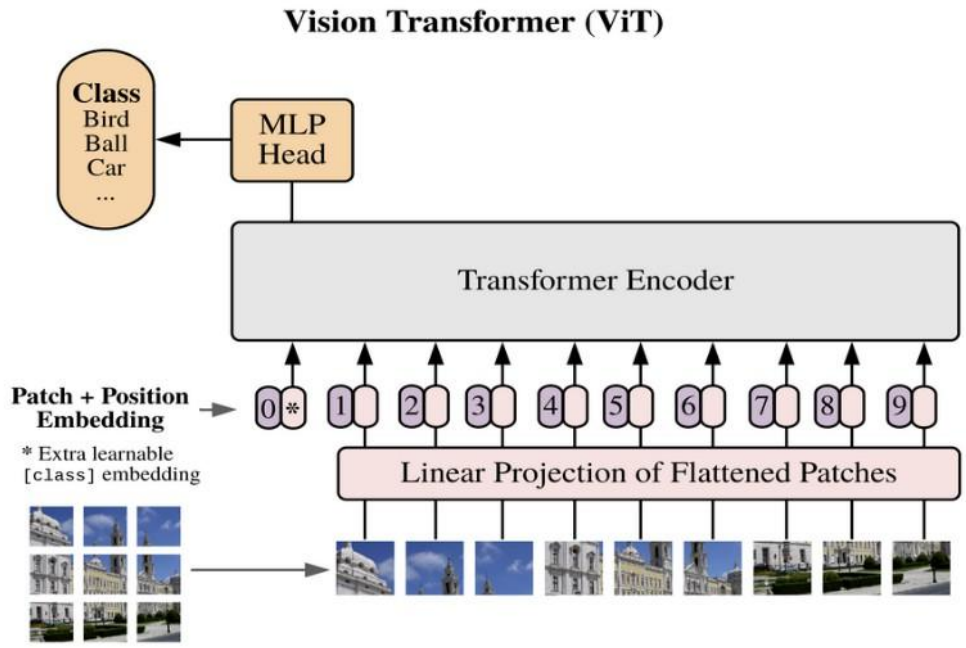
- Feature maps are easier to interpret:
- Early layers detect **edges and textures**.
- Later layers detect **objects and patterns**.

ViTs:

- Use **attention maps** to highlight important regions.

Model	Layers	Width	MLP	Heads	Params
ViT-Ti (39)	12	192	768	3	5.8M
ViT-S (39)	12	384	1536	6	22.2M
ViT-B (13)	12	768	3072	12	86M
ViT-L (13)	24	1024	4096	16	307M

Details of Vision Transformer model variants. ViT-Huge is not included here, which uses 632M parameters. (Image source: Steiner et al. 2021)



The Vision Transformer architecture. It is largely identical to the original Transformer architecture by Vaswani et al. (2017). (Image source: Dosovitskiy et al. 2020)

Let us compare the performance and accuracy of both the models to understand better

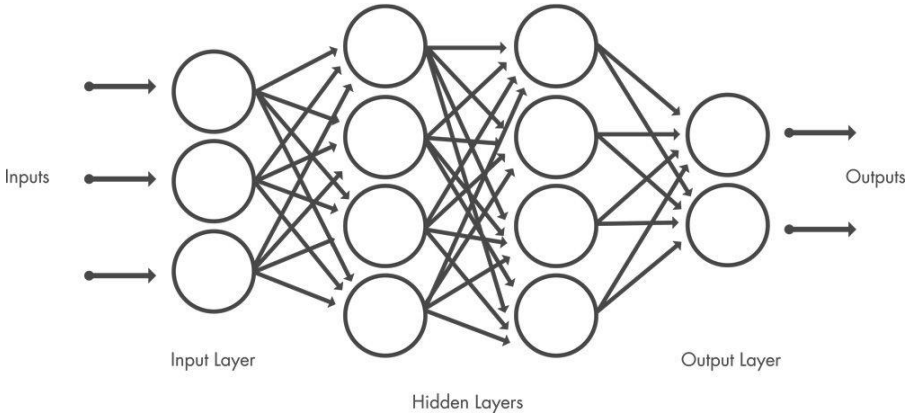
Model	CIFAR-10 Accuracy (%)	ImageNet Accuracy (%)
ResNet-18 (CNN)	90.1%	72.0%
ViT-Base	92.2%	84.5%
ResNet-50 (CNN)	93.2%	78.3%
ViT-Large	96.1%	88.5%

Table for accuracy of both models on two different datasets

Now let us deep dive into the working of both the models

Working of CNN:

A CNN is composed of an input layer, an output layer, and many hidden layers in between.



These layers perform operations that alter the data with the intent of learning features specific to the data. Three of the most common layers are convolution, activation or ReLU, and pooling.

These operations are repeated over tens or hundreds of layers, with each layer learning to identify different features.

Unlike a traditional neural network, a CNN has shared weights and bias values, which are the same for all hidden neurons in a given layer.

This means that all hidden neurons are detecting the same feature, such as an edge or a blob, in different regions of the image. This makes the network tolerant to translation of objects in an image. For example, a network trained to recognize cars will be able to do so wherever the car is in the image.

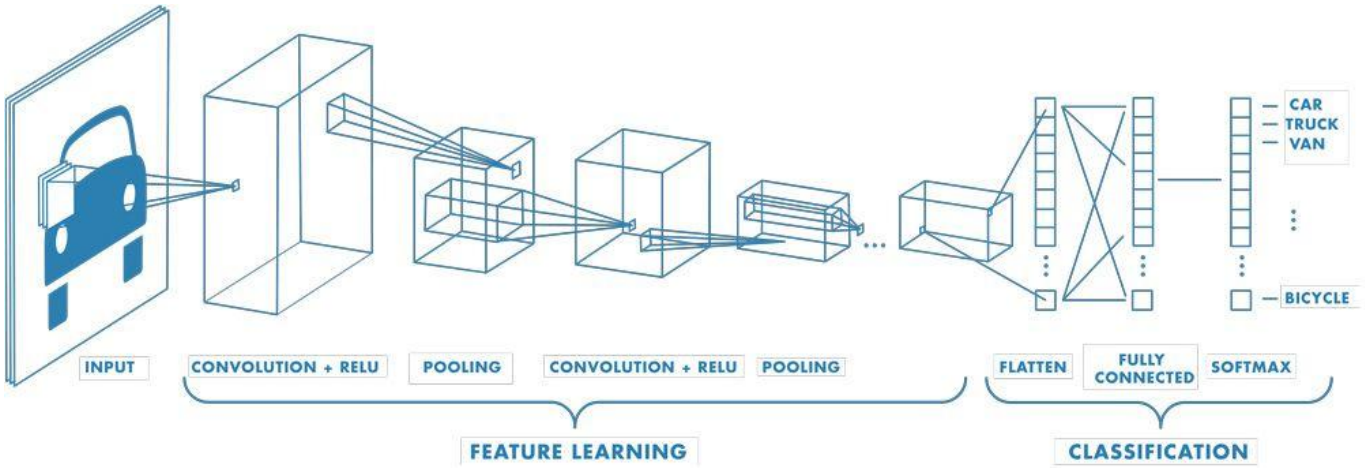


Image Source: Mathworks

2) Working of ViT

Image is processed as Patches instead of working with raw pixel values directly, ViTs split an image into fixed-size non-overlapping patches (e.g., 16×16 pixels). Each patch is flattened into a vector and treated like a token in NLP. The total number of patches depends on the image size and patch size. For example, a 224×224 image with 16×16 patches results in $(224/16)^2 = 196$ patches**.** The core of ViT is the Transformer Encoder, which consists of Multi-Head Self-Attention (MHSA): Computes relationships between all patches to capture global dependencies.

- Feed-Forward Network (FFN): Applies non-linearity to enhance feature representation.

- Layer Normalization & Residual Connections: Stabilize training and improve gradient flow.

Classification Head

- A special [CLS] (classification token) is added to the sequence before passing through the transformer layers.
- After processing, the final [CLS] token representation is passed to a fully connected (FC) layer for classification.

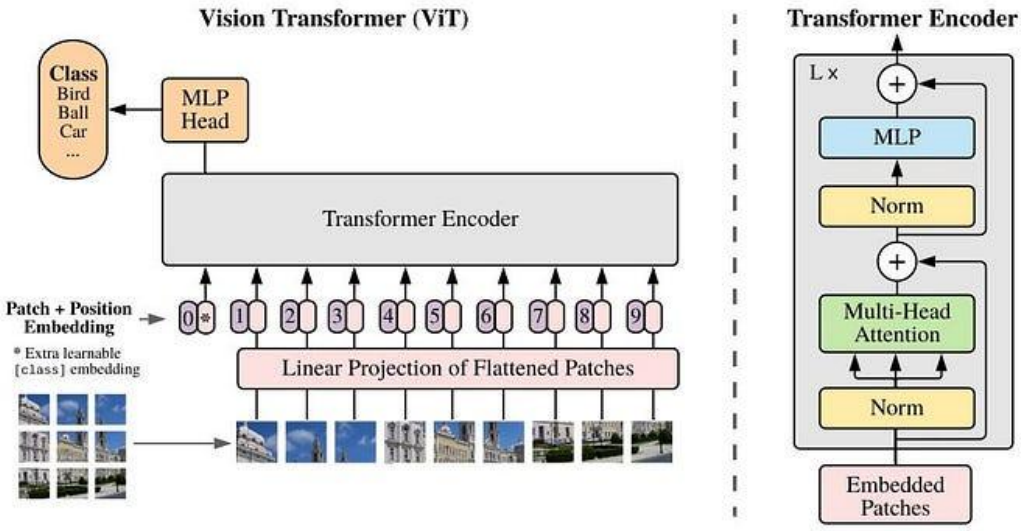


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable "classification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

Figure (taken from the original paper) describes a model that processes 2D images by transforming them into sequences of flattened 2D patches. The patches are then mapped to a constant latent vector size with a trainable linear projection. A learnable embedding is prepended to the sequence of patches and its state at the output of the Transformer encoder serves as the image representation. The image representation is then passed through a classification head for either pre-training or fine-tuning. Position embeddings are added to retain positional information and the sequence of embedding vectors serves as input to the Transformer encoder, which consists of alternating layers of multiheaded self-attention and MLP blocks.

ViT vs CNN Which model is suitable for which type of data and use cases

In the past CNN were widely used for image classification, they dominated the field of image classification in Machine Learning, they learn from large datasets and are self-learning models, due to hidden layers the performance is also great.

It is also worth to mention that Vit are self-learning models so they learn and adapt quickly on their own.

We can conclude that both Vit and CNN have different use cases and are used widely for image classification, CNN has been an image classification model for longer time than ViT.

Vit has a complex architecture which may be difficult to understand for someone new to machine learning and are trying to learn image classification techniques.

In conclusion we can understand that we can use

- Use CNNs when you need fast, efficient models on small datasets.
- Use ViTs for large-scale datasets where self-attention can leverage global context.

I will provide some links to various free resources if you want to learn more about Vision Transformers or want to develop a model based on CNN or ViT

1) Youtube link to learn how to build an ViT model
2) Original Research Paper on ViT
3) Youtube Video to learn CNN for beginners

In this tutorial I tried to cover most of the important aspects of CNN and ViT but there may be better resources or topics,aspects I missed due to my error.

Finally, I want to conclude that there is no better way to learn than self-learning, as you can observe that ViT also self-learn their mistakes and improve their accuracy, in the same way we should learn our mistakes and improve our knowledge and accuracy.