Md: Israil Hosen

Roll: 2010876110

Neural Network and Deep Learning Assignment-1

In this assignment, ten different CNN-based pretrained models were evaluated individually to measure and compare their classification performance. The models utilized for this evaluation include: Xception, VGG16, VGG19, ResNet50, InceptionV3, MobileNet, DenseNet121, EfficientNetB1, NASNetMobile, and NASNetLarge. Each model was used as a fixed feature extractor by freezing its convolutional base, and a common custom classification head was added on top to perform the final predictions. The performance of each model was assessed using test accuracy on a subset of 20 randomly selected classes from the CIFAR-100 dataset.

Xception Pretrained Model [Code link]

Short Description About Xception

[1]The Xception model (Extreme Inception) is a deep learning model for image classification, developed by François Chollet in 2016. It improves the older Inception architecture by using depthwise separable convolutions, which make it faster and more efficient. This design helps Xception achieve high accuracy on image recognition tasks while using fewer computations.

[2]The Xception model is a powerful deep learning model used for recognizing images. It gets 79.0% accuracy when guessing the top answer and 94.5% accuracy when allowed to guess up to five answers, tested on a large dataset called ImageNet with 1000 types of images. Because it's a big model, it works much faster on a GPU, though it can still run on a CPU, just more slowly. The model file is about 88 MB, and it takes about 109 milliseconds per image on CPU and only 8 milliseconds on GPU to make a prediction. It runs best on a computer with a good GPU (like 8 GB or more). Xception is built with 81 layers and has about 22.9 million parameters.

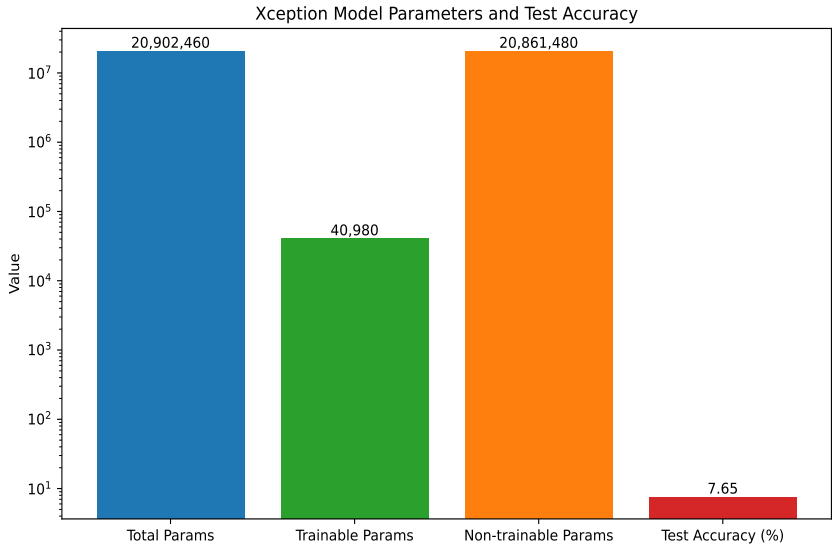

Figure 1: Investigated the results produced by the Xception model

VGG16 Pretrained Model

Short Description About VGG16

[3]The VGG-16 model is a convolutional neural network (CNN) architecture that was proposed by the Visual Geometry Group (VGG) at the University of Oxford. It is characterized by its depth, consisting of 16 layers, including 13 convolutional layers and 3 fully connected layers.

[2]The VGG16 model is a widely used deep learning architecture for image recognition tasks. It achieves about 71.3% accuracy when predicting the top answer and 90.1% accuracy when allowed to guess up to five answers, evaluated on the large ImageNet dataset containing 1000 image categories. This model is quite large, with around 138.4 million parameters spread across 16 layers, making it more computationally demanding than some newer models. Because of its size, VGG16 runs significantly faster on a GPU but can also operate on a CPU at slower speeds. The model file size is roughly 528 MB, and it takes about 69.5 milliseconds to process each image on a GPU, while on a CPU it requires approximately 4.2 milliseconds per image to generate predictions.

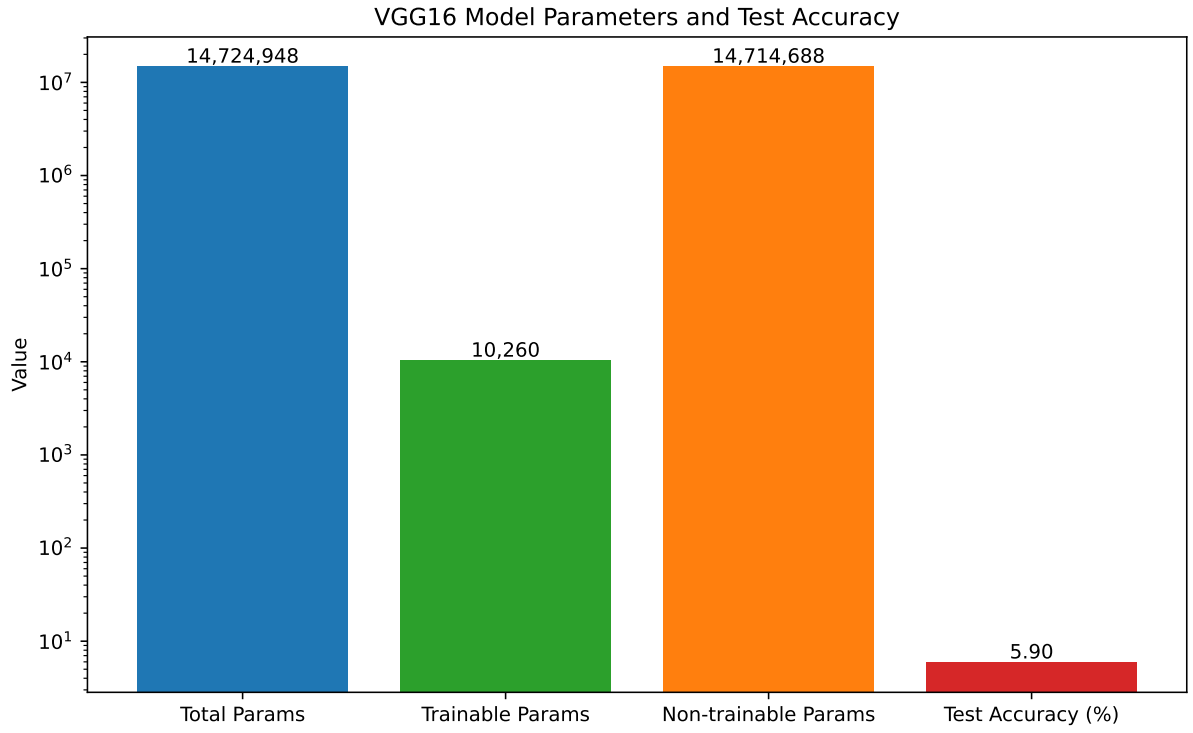

Figure 2: Investigated the results produced by the VGG16 model

VGG19 Pretrained Model

Short Description About VGG19

[4]VGG-19 is a deep convolutional neural network consisting of 19 weight layers, including 16 convolutional layers and 3 fully connected layers. It uses a simple and repetitive architecture with 3x3 convolution filters, ReLU activation functions, and max-pooling layers to gradually reduce spatial dimensions while preserving important features. The network ends with fully connected layers followed by a softmax layer to output class probabilities, making it effective and easy to implement for image classification tasks.

[2] The VGG19 model is a deep learning architecture widely used for image recognition tasks. It achieves approximately 71.3% accuracy for the top-1 prediction and about 90.0% accuracy when allowed to guess up to five answers and evaluated on the ImageNet dataset with 1000 image categories. This model is larger than VGG16, containing around 143.7 million parameters distributed over 19 layers, which makes it more computationally intensive. The model file size is roughly 549 MB. On a GPU, VGG19 processes each image in about 84.8 milliseconds, while on a CPU it takes approximately 4.4 milliseconds per image to generate predictions.

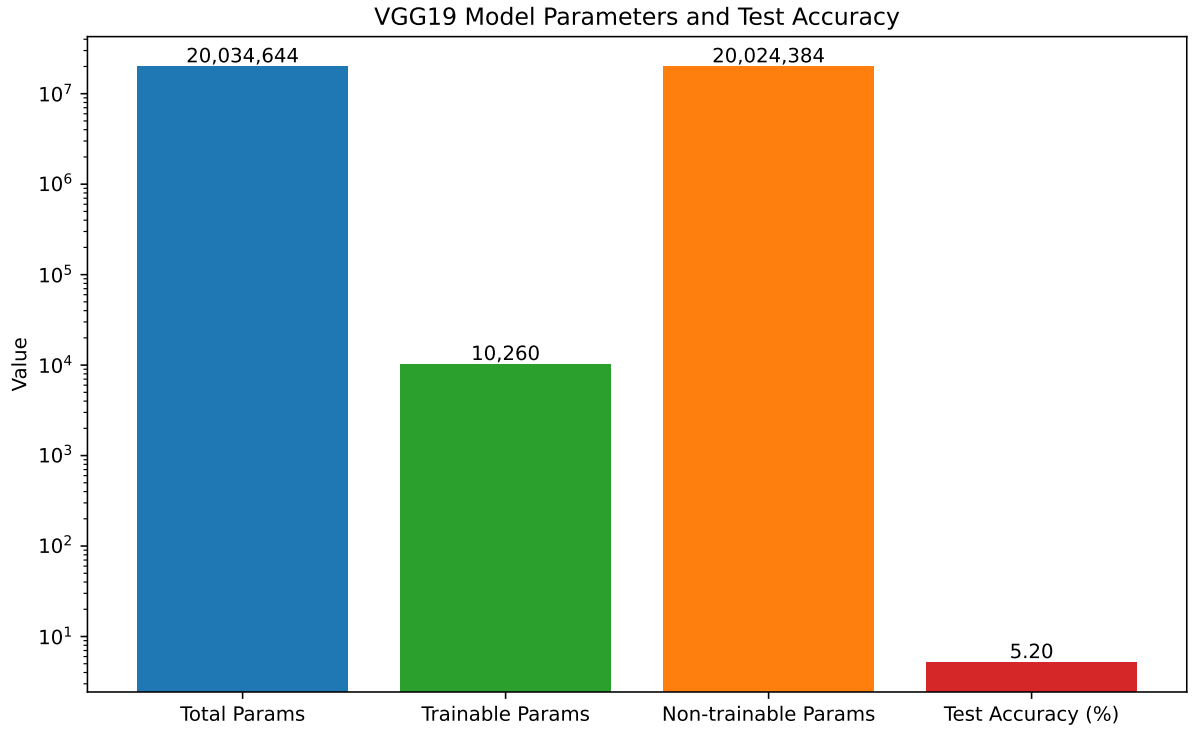

Figure 3: Investigated the results produced by the VGG19 model

ResNet50 Pretrained Model

Short Description About ResNet50

[5]ResNet50 is a deep convolutional neural network (CNN) architecture that was developed by Microsoft Research in 2015. It is a variant of the popular ResNet architecture, which stands for "Residual Network". The "50" in the name refers to the number of layers in the network, which is 50 layers deep.

[2]The ResNet50 model is a deep learning architecture commonly used for image recognition tasks. It achieves around 74.9% top-1 accuracy and approximately 92.1% top-5 accuracy on the ImageNet dataset, which includes 1000 image categories. The model contains about 25.6 million parameters spread across 50 layers. Its file size is roughly 98 MB. ResNet50 can process an image in approximately 58.2 milliseconds on a GPU, and about 4.6 milliseconds on a CPU to generate predictions.

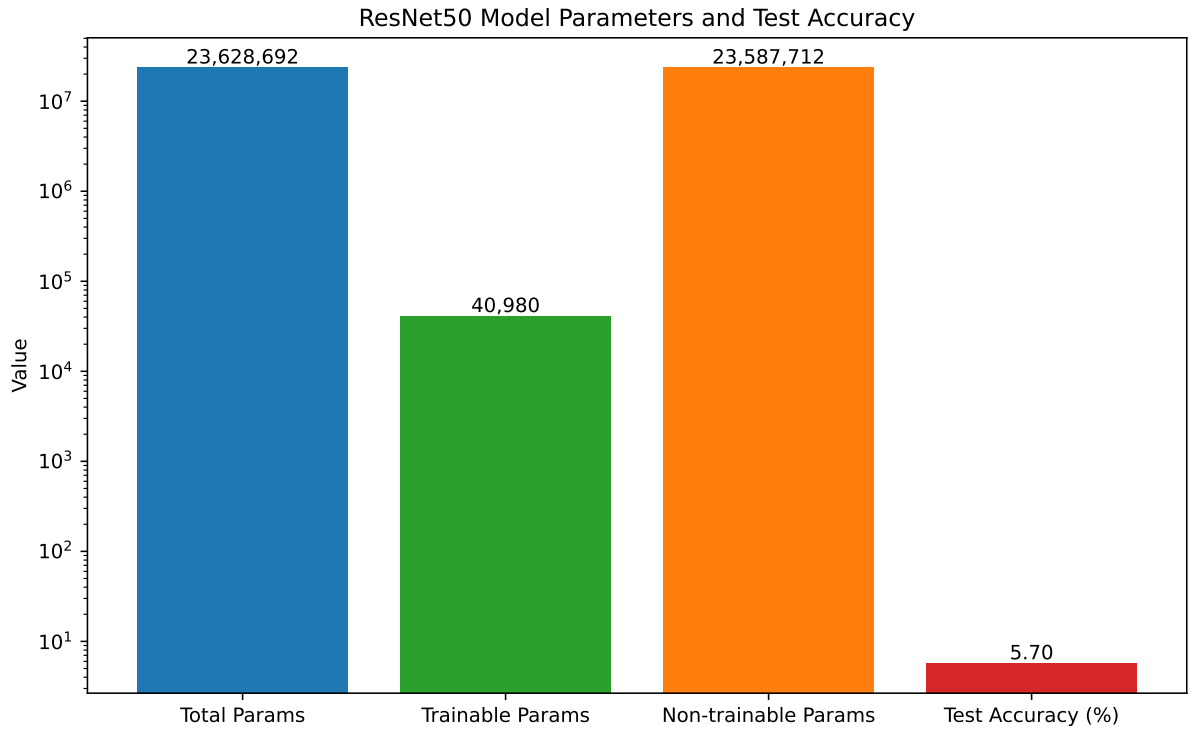

Figure 4: Investigated the results produced by the ResNet50 model

InceptionV3 Pretrained Model

Short Description About InceptionV3

[6]InceptionV3 is a convolutional neural network architectures used in computer vision tasks and developed by a team of researchers at Google in 2015. InceptionV3 is part of the Inception family and was designed to addressed some of the limitations of the earlier Inception model. InceptionV3 uses a more complex architecture, which allows for better accuracy in image classification tasks. This is achieved by using a technique called "Factorised Convolution", which reduces the computational cost and improves the model's efficiency. The original input size image for InceptionV3 is 299 x 299 pixels. InceptionV3 has been designed to process images at this specific size, and using images of different sizes may result in lower accuracy and performance.

[2]The InceptionV3 model is a deep learning architecture designed for image recognition tasks. It achieves approximately 77.9% top-1 accuracy and 93.7% top-5 accuracy on the ImageNet dataset, which includes 1000 image categories. The model consists of about 23.9 million parameters and has a total of 189 layers. Its file size is around 92 MB. InceptionV3 processes a single image in roughly 42.2 milliseconds on a GPU and takes about 6.9 milliseconds per image on a CPU to generate predictions.

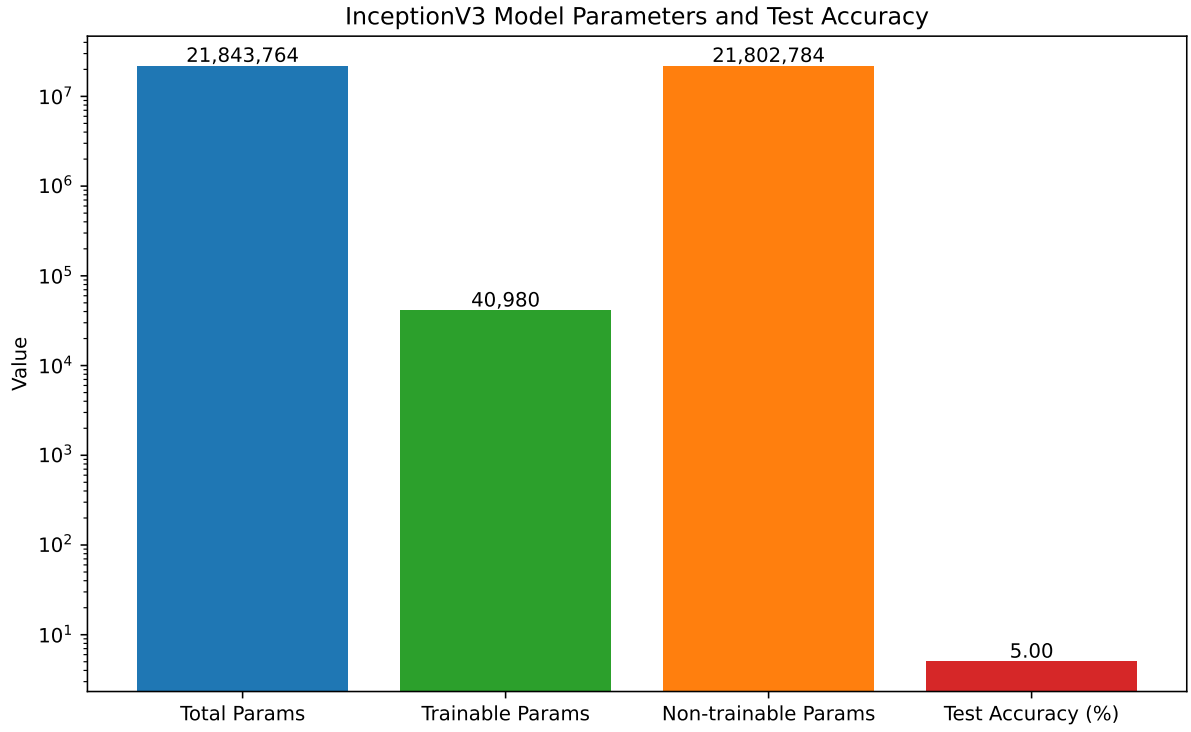

Figure 5: Investigated the results produced by the InceptionV3 model

MobileNet Pretrained Model

Short Description About MobileNet

[7]MobileNets are a type of artificial intelligence model that are designed to enable efficient and accurate image classification and object detection on mobile and embedded devices. These models are specifically optimized to run on low-power devices with limited computational resources, making them ideal for mobile and edge applications. MobileNets have become increasingly popular in recent years, particularly in the development of mobile and edge applications that require image processing and computer vision capabilities.

[2]The MobileNet model is a lightweight deep learning architecture designed for efficient image recognition, especially on mobile and embedded devices. It achieves approximately 70.4% top-1 accuracy and 89.5% top-5 accuracy on the ImageNet dataset, which includes 1000 image categories. The model contains about 4.3 million parameters and consists of 55 layers. Its file size is around 16 MB. MobileNet processes a single image in approximately 22.6 milliseconds on a GPU and takes about 3.4 milliseconds per image on a CPU to generate predictions.

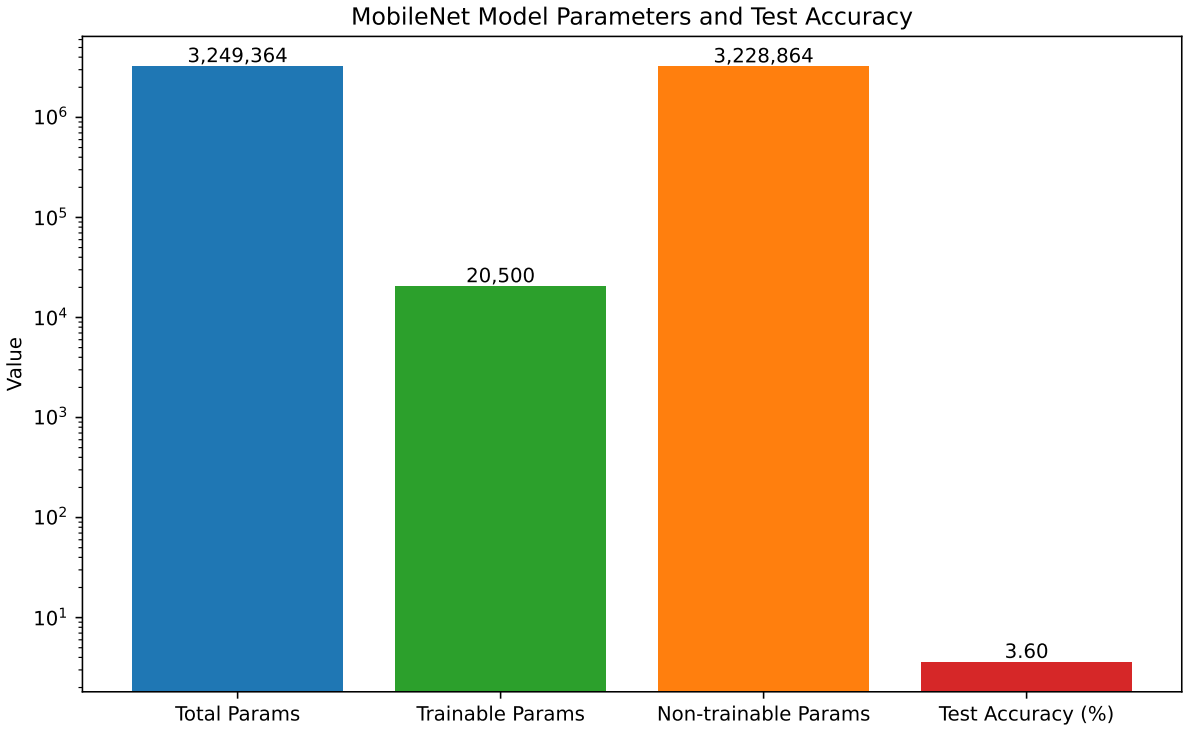

Figure 6: Investigated the results produced by the MobileNet model

DenseNet121 Pretrained Model

Short Description About DenseNet121

[8]DenseNet is a powerful deep learning architecture that uses dense connectivity between layers to boost the performance of convolutional neural networks. This advanced architecture has demonstrated significant efficiency across a wide range of computer vision tasks, such as image classification, object detection, and segmentation. Its use-cases span multiple image-related applications, including face recognition, animal type identification, object detection, cancerous cell identification, among others.

[2] The DenseNet121 model is a deep learning architecture used for image recognition tasks. It achieves approximately 75.0% top-1 accuracy and 92.3% top-5 accuracy on the ImageNet dataset, which includes 1000 image categories. The model has about 8.1 million parameters distributed across 242 layers. Its file size is approximately 33 MB. DenseNet121 processes a single image in around 77.1 milliseconds on a GPU and takes about 5.4 milliseconds per image on a CPU to generate predictions.

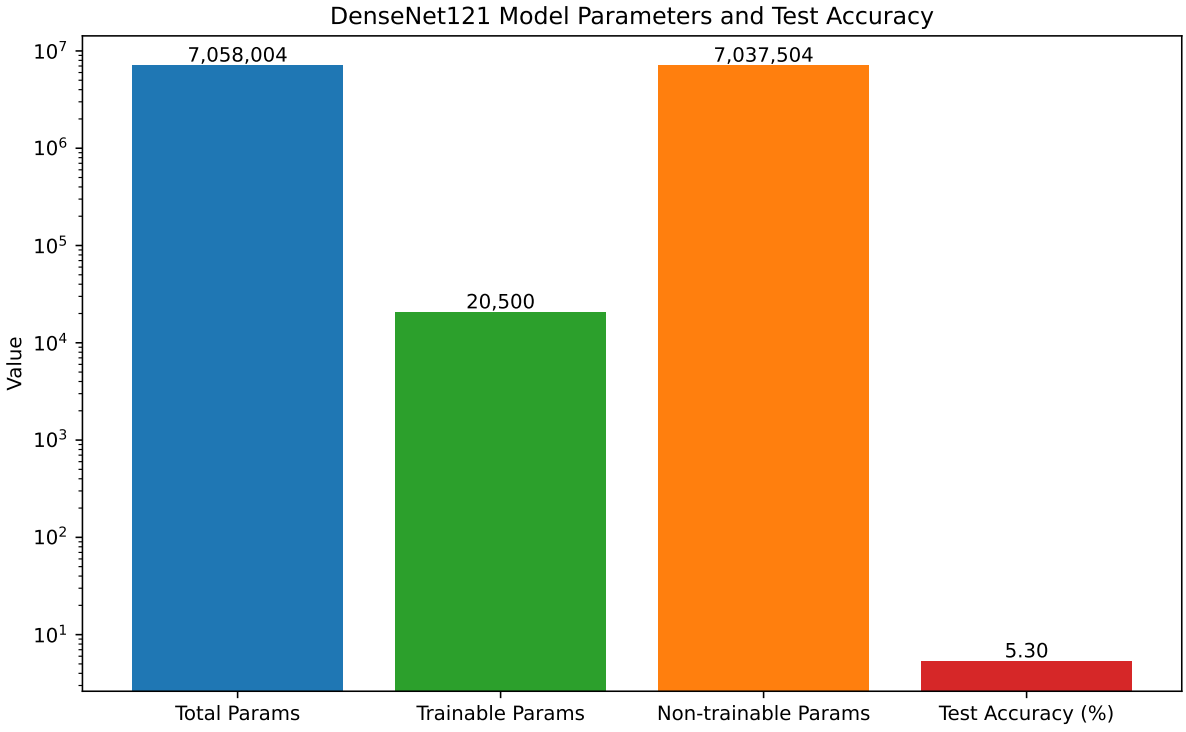

Figure 7: Investigated the results produced by the DenseNet121 model

EfficientNetB1 Pretrained Model

Short Description About EfficientNetB1

[9]EfficientNet is a convolutional neural network architecture and scaling method that uniformly scales all dimensions of depth/width/resolution using a compound coefficient. Unlike conventional practice that arbitrary scales these factors, the EfficientNet scaling method uniformly scales network width, depth, and resolution with a set of fixed scaling coefficients.

[2]The EfficientNetB1 model is a deep learning architecture designed for high accuracy and efficiency in image recognition tasks. It achieves approximately 79.1% top-1 accuracy and 94.4% top-5 accuracy on the ImageNet dataset, which contains 1000 image categories. The model includes about 7.9 million parameters across 186 layers. Its file size is around 31 MB. EfficientNetB1 processes a single image in approximately 60.2 milliseconds on a GPU and takes about 5.6 milliseconds per image on a CPU to generate predictions.

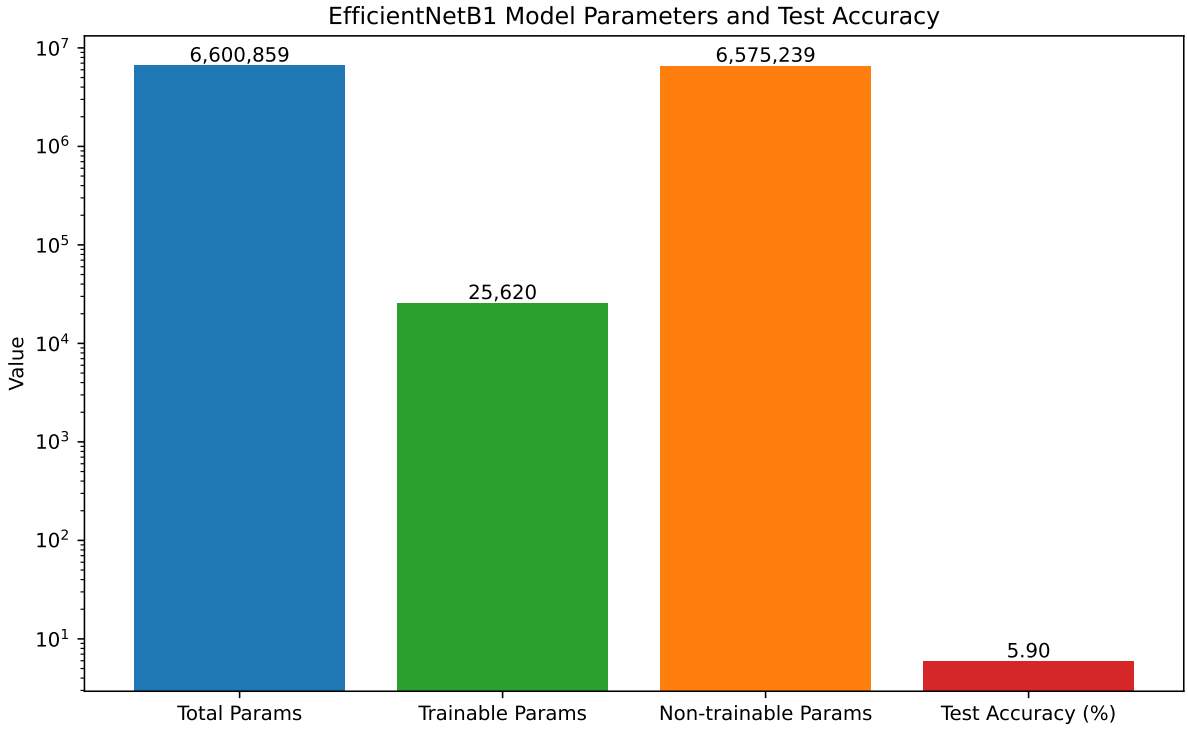

Figure 8: Investigated the results produced by the EfficientNetB1 model

NASNetMobile Pretrained Model

Short Description About NASNetMobile

[10]NASNetMobile is a lightweight convolutional neural network designed for mobile and embedded devices. It is based on Neural Architecture Search (NAS), an automated process that finds efficient network architectures. NASNetMobile offers a good balance between accuracy and computational efficiency, making it suitable for image classification tasks on devices with limited resources. It achieves competitive accuracy on the ImageNet dataset while keeping the model size and computation cost low.

[2]The NASNetMobile model is a deep learning architecture optimized for mobile and efficient image recognition tasks. It achieves approximately 74.4% top-1 accuracy and 91.9% top-5 accuracy on the ImageNet dataset, which includes 1000 image categories. The model has about 5.3 million parameters spread over 389 layers. Its file size is roughly 23 MB. NASNetMobile processes a single image in about 27.0 milliseconds on a GPU and takes approximately 6.7 milliseconds per image on a CPU to generate predictions.

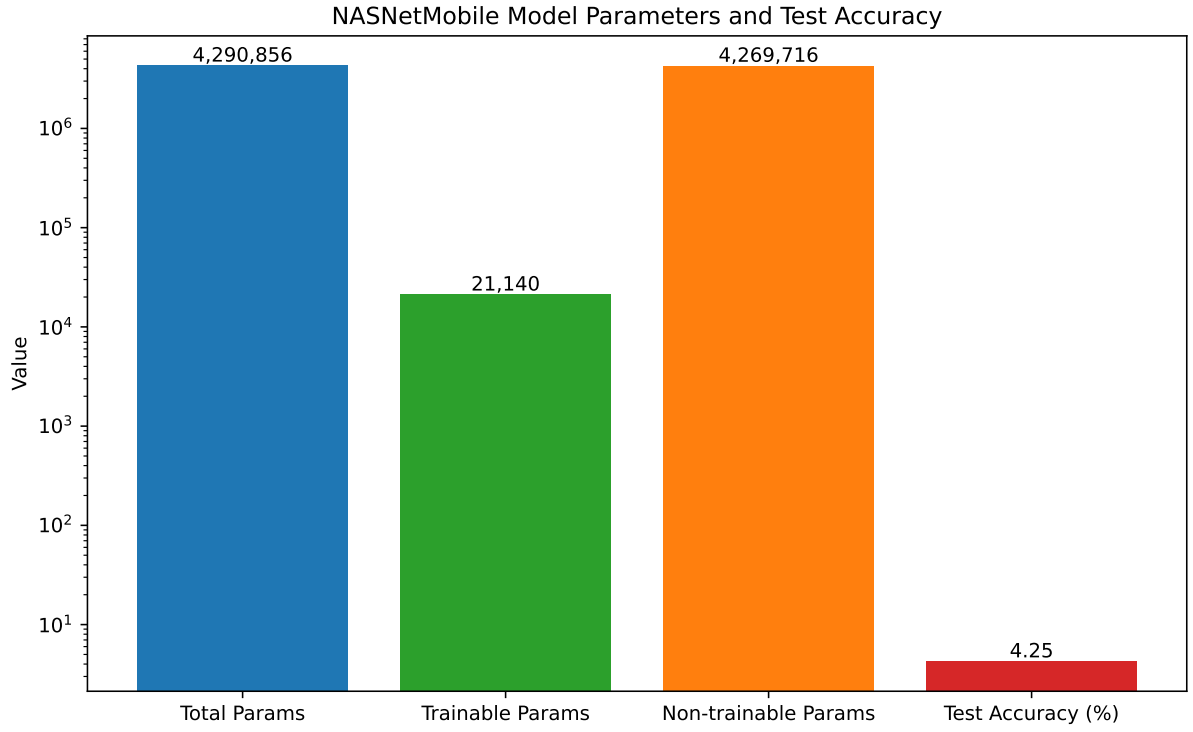

Figure 9: Investigated the results produced by the NASNetMobile model

NASNetLarge Pretrained Model

Short Description About NASNetLarge

[10]NASNetLarge is a deep convolutional neural network architecture designed using Neural Architecture Search (NAS), an automated method for discovering high-performance models. It is larger and more powerful than NASNetMobile, achieving higher accuracy on image classification benchmarks like ImageNet. Due to its complexity and size, NASNetLarge requires significant computational resources and is typically used in environments where accuracy is prioritized over speed or efficiency.

[2]The NASNetLarge model is a deep learning architecture designed for high-performance image recognition tasks. It achieves approximately 82.5% top-1 accuracy and 96.0% top-5 accuracy on the ImageNet dataset, which includes 1000 image categories. The model contains about 88.9 million parameters distributed across 533 layers. Its file size is roughly 343 MB. NASNetLarge processes a single image in approximately 344.5 milliseconds on a GPU and takes about 20.0 milliseconds per image on a CPU to generate predictions.

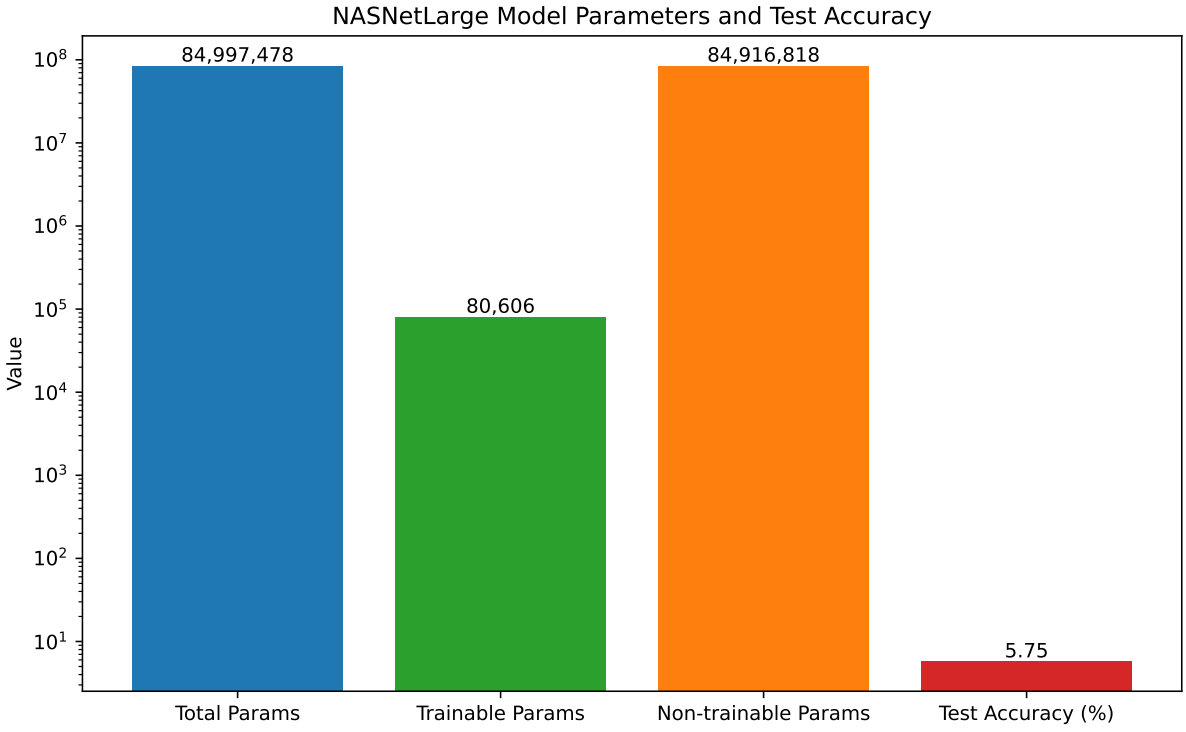

Figure 10: Investigated the results produced by the NASNetLarge model

Overall Model Accuracy Comparison

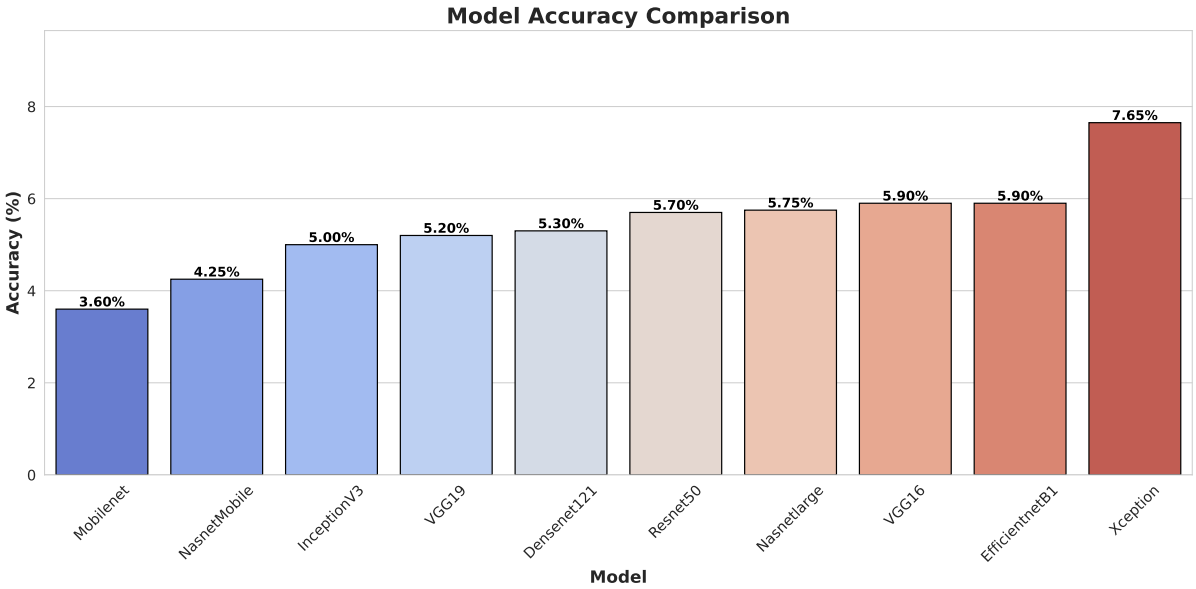

Figure 11: Overall Model Accuracy Comparison

References

[1] http://www.baguett.eu/obsidiangpt/imagedata/imageclassification/xception.html.

[2] https://keras.io/api/applications/.

[3] https://www.geeksforgeeks.org/vgg-16-cnn-model/.

[4] https://www.geeksforgeeks.org/vgg-net-architecture-explained/.

[5] https://medium.com/@nitishkundu1993/exploring-resnet50-an-in-depth-look-at-the-model-architecture-and-code-implementation-d8d8fa67e46f.

[6] https://medium.com/@nocodingai/inceptionv3-10e6f48e4553.

[7] https://medium.com/@nocodingai/mobilenet-fc34af9f58a5.

[8] https://medium.com/nocoding-ai/densenet121-760df192f12d.

[9] https://paperswithcode.com/method/efficientnet.

[10] https://chatgpt.com/.