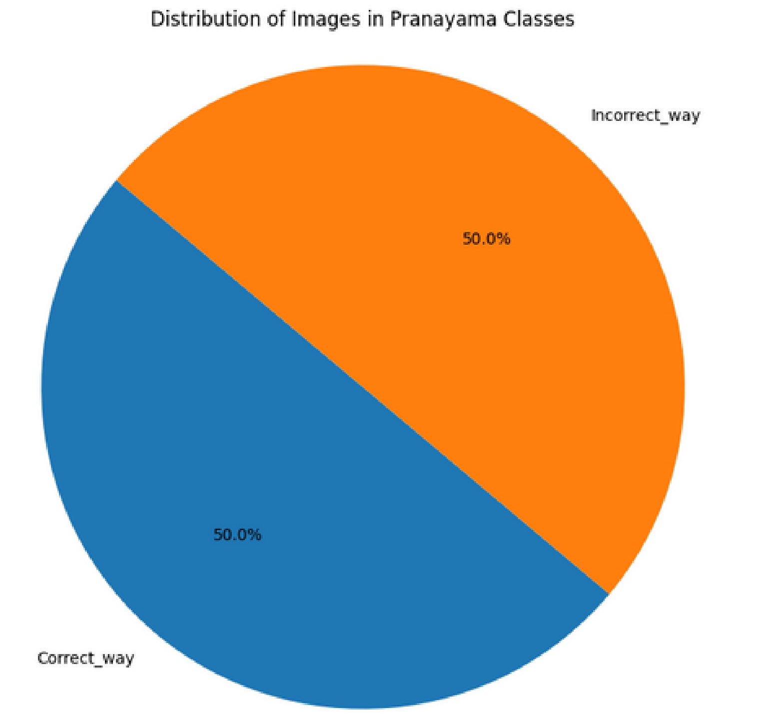
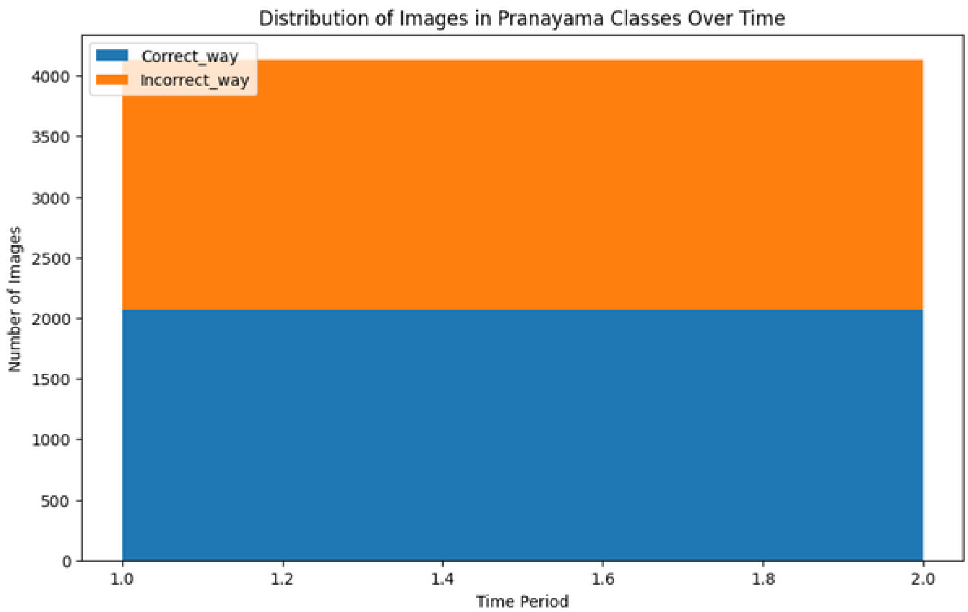
Pranayama Dataset: An Extensive Dataset of Correct and Incorrect Postures of Breathing Exercises

Tanishk N Shinde

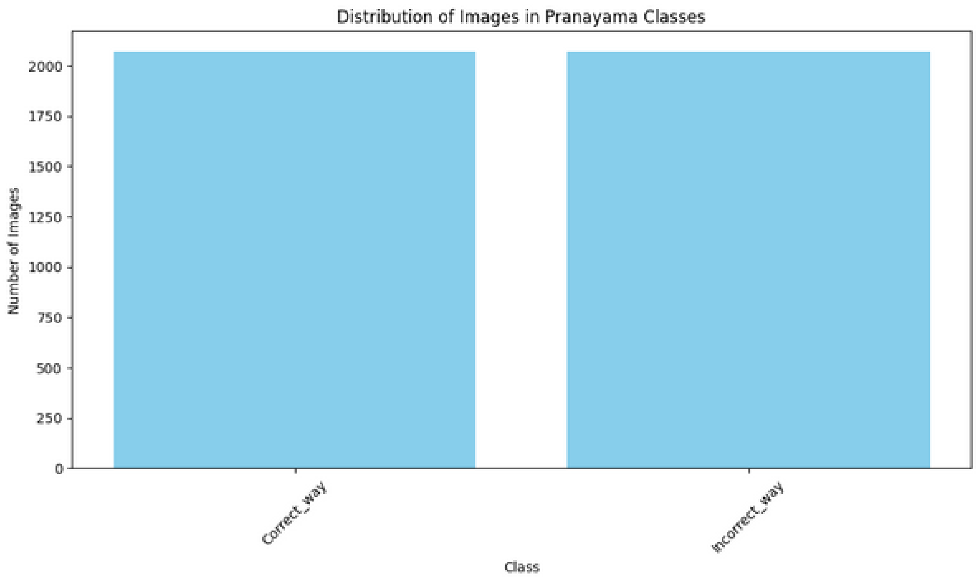
TABLE OF CONTENTS

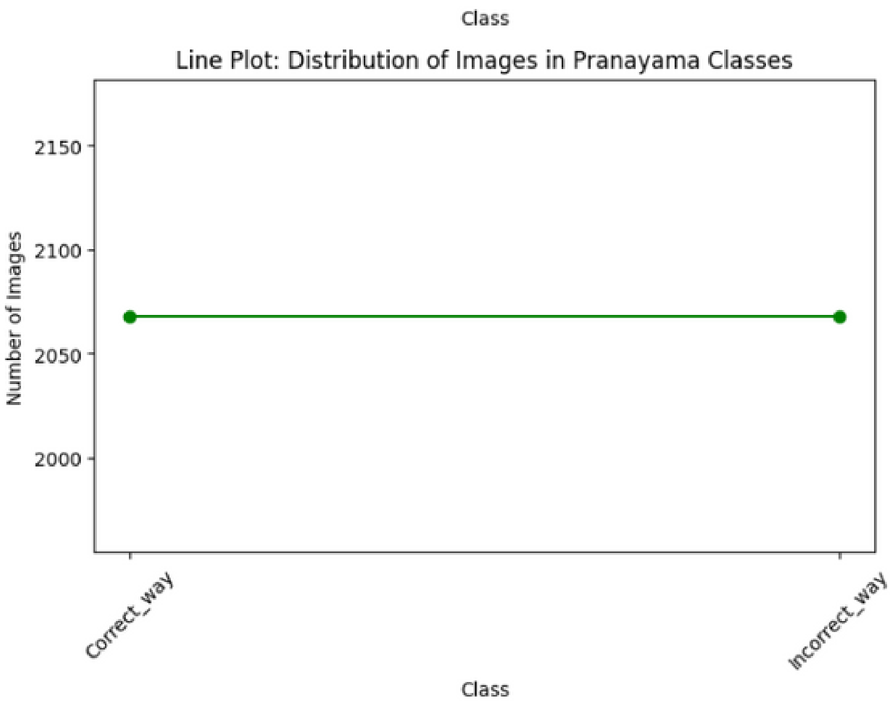
DATA AND INFORMATION VISUALIZATION



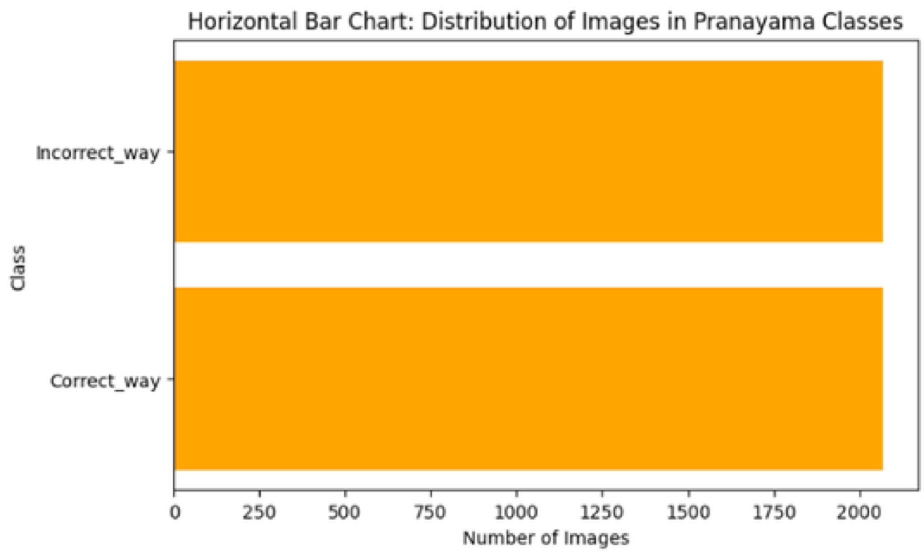


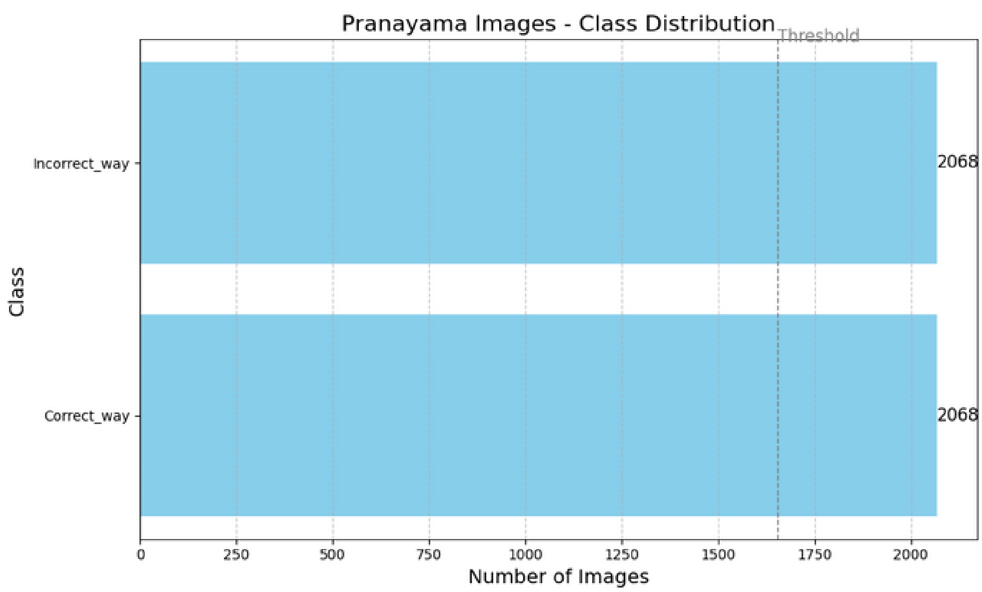
DATA AND INFORMATION VISUALIZATION





DATA AND INFORMATION VISUALIZATION





EFFICIENTNET-B0

EfficientNet-B0 is a convolutional neural network (CNN) architecture that has gained prominence in the field of deep learning for computer vision tasks. It was introduced by researchers in 2019 and is known for its efficiency and impressive performance in image classification tasks.

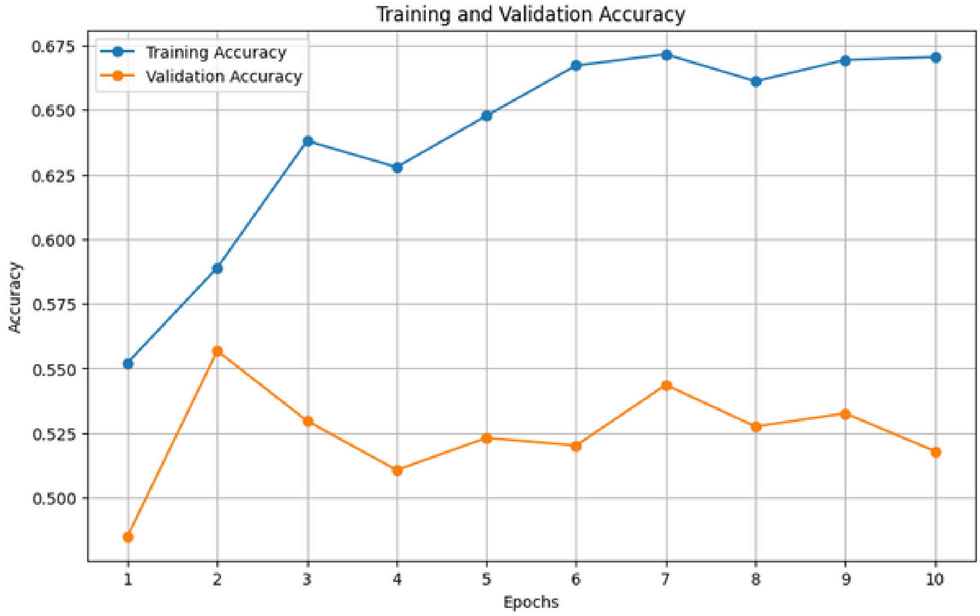
EfficientNet-B0 is part of a family of models, with 'B0' denoting the base model. The 'EfficientNet' series is designed to be highly efficient in terms of both computational resources and model size while delivering state-of-the-art performance. The key innovation behind EfficientNet is the compound scaling, which involves optimizing the network's depth, width, and resolution simultaneously. This scaling strategy allows EfficientNet to achieve a superior balance between accuracy and computational cost.

EfficientNet-B0 employs a combination of depthwise separable convolutions and a "swish" activation function, which contributes to its compactness and effectiveness. The model architecture consists of multiple stacked blocks, including inverted residual blocks and linear bottlenecks, which help it capture intricate features in the data.

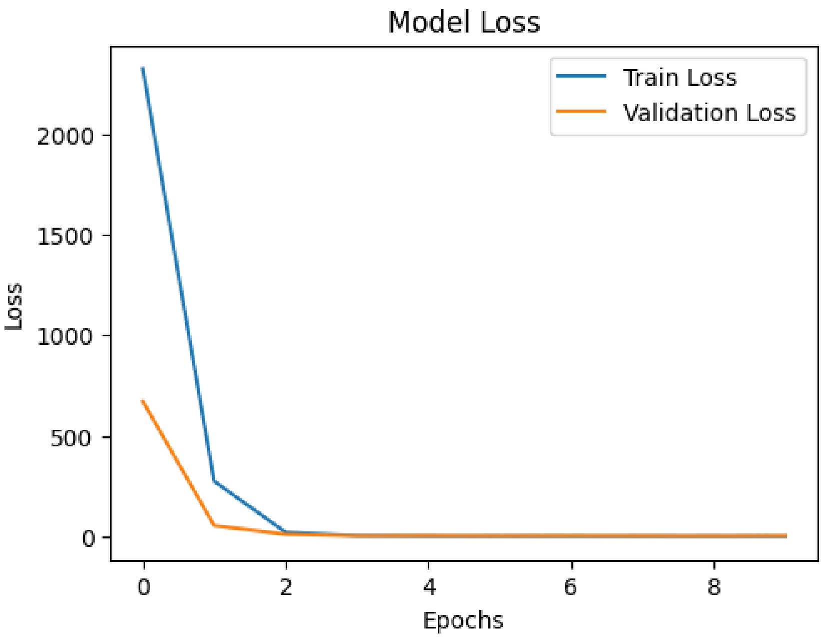
One of the remarkable aspects of EfficientNet-B0 is its transfer learning capability. By pre-training on large image datasets like ImageNet and then fine-tuning on specific tasks, it can adapt to a wide range of image classification challenges. This transfer learning approach has made EfficientNet-B0 a popular choice for various computer vision applications.

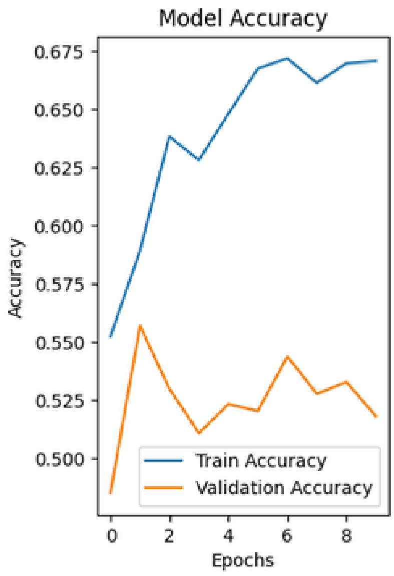
EfficientNet-B0 has served as a baseline model for other EfficientNet variants (B1, B2, B3, etc.) that are larger and more powerful. The choice of variant depends on the specific requirements of a task, with larger variants offering better accuracy but requiring more computational resources.

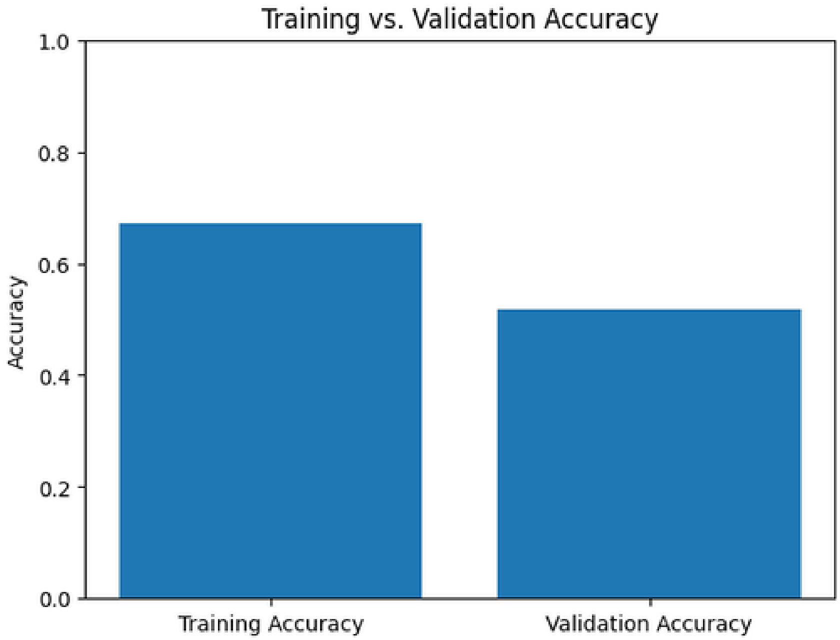
In summary, EfficientNet-B0 is a highly efficient and effective convolutional neural network architecture for image classification tasks. Its innovative compound scaling approach, compact design, and transfer learning capabilities make it a valuable tool in the deep learning community for a wide range of computer vision applications.

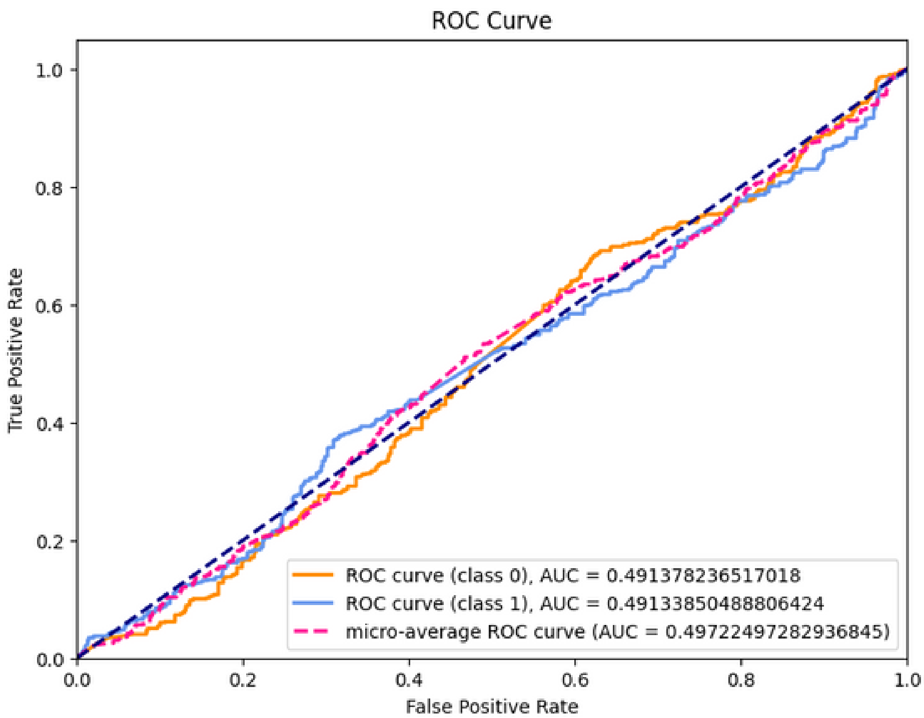


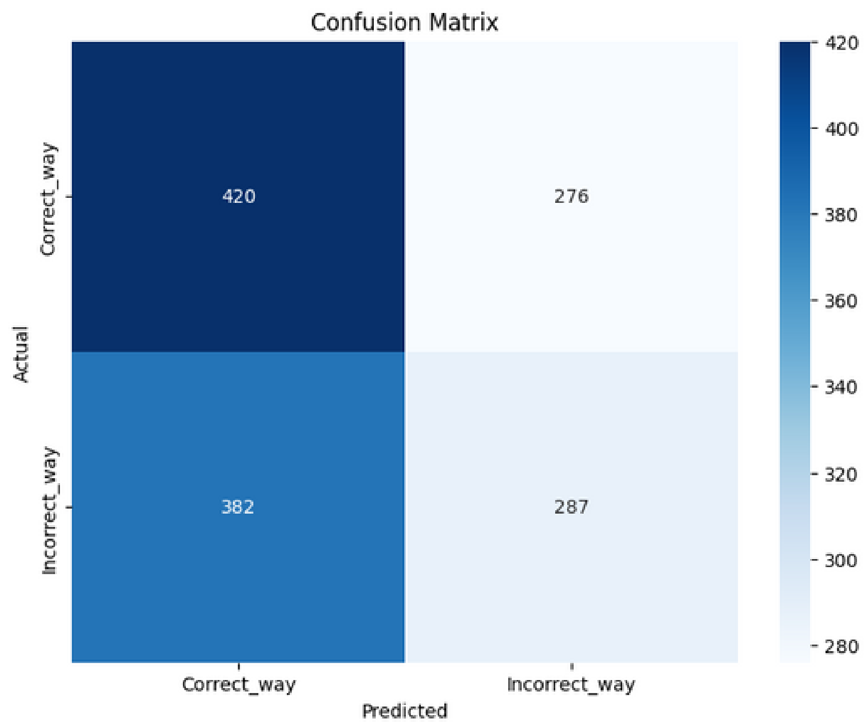
Final Training Accuracy: 0.6705
Fina

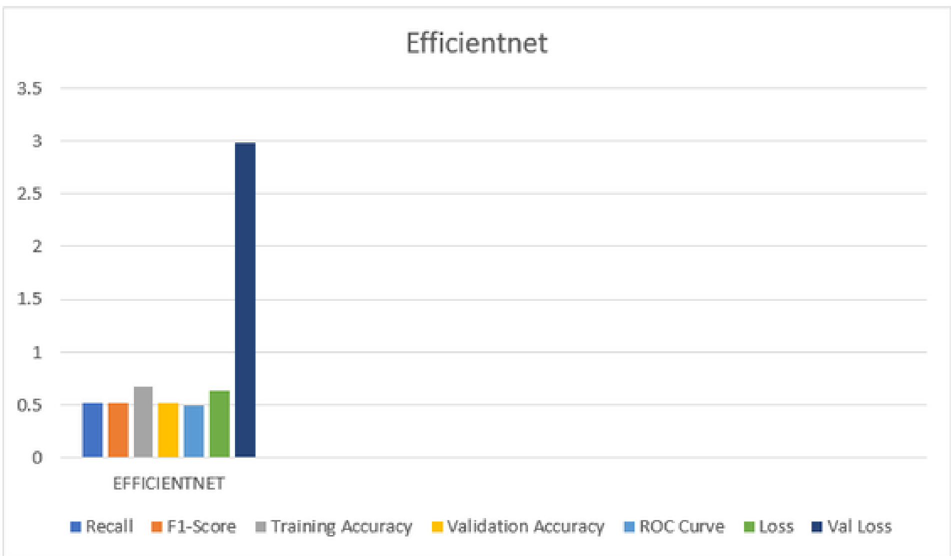





Training vs. Validation Accuracy
Accuracy
1.0
0.8
0.6
0.4
0.2
0.0
Training Accuracy
Validation Accuracy


ROC Curve
True Positive Rate
False Positive Rate
1.0
0.8
0.6
0.4
0.2
0.0
ROC curve (class 0), AUC = 0.491378236517018
ROC curve (class 1), AUC = 0.49133850488806424
micro-average ROC curve (AUC = 0.49722497282936845)


Confusion Matrix
Actual
Correct_way
Incorrect_way
420
276
382
287
420
400
380
360
340
320
300
280
Correct_way
Incorrect_way
Predicted


Efficientnet
3.5
3
2.5
2
1.5
1
0.5
0
EFFICIENTNET
Recall
F1-Score
Training Accuracy
Validation Accuracy
ROC Curve
Loss
Val Loss

MOBILENETV2

MobileNetV2 is a highly efficient convolutional neural network (CNN) architecture designed for mobile and embedded devices. It was developed by Google as a successor to the original MobileNet, with the primary goal of enabling state-of-the-art deep learning performance on resource-constrained platforms. Here's an overview in 300 words:

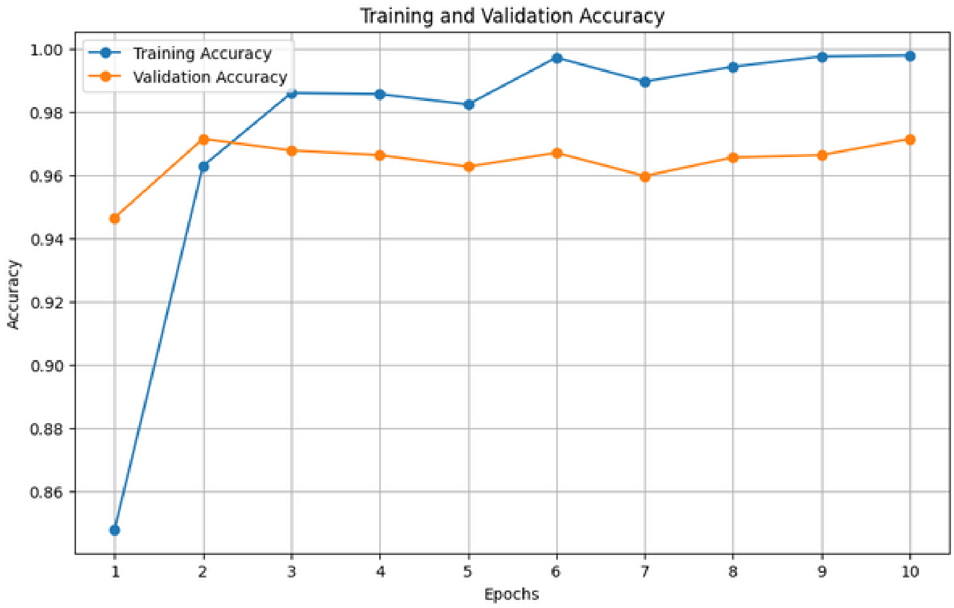
MobileNetV2 is characterized by its lightweight and compact design, making it ideal for applications like real-time object recognition on smartphones, IoT devices, and other edge computing environments. The architecture is known for its exceptional speed and accuracy, making it a popular choice for tasks such as image classification, object detection, and more.

The key innovation in MobileNetV2 is the use of inverted residual blocks, which include a bottleneck layer with fewer filters followed by expansion and projection layers. This design reduces the computational load while maintaining high representation capacity. MobileNetV2 also employs depthwise separable convolutions, which split standard convolutions into depthwise and pointwise convolutions. This separation further reduces the number of parameters and computational requirements.

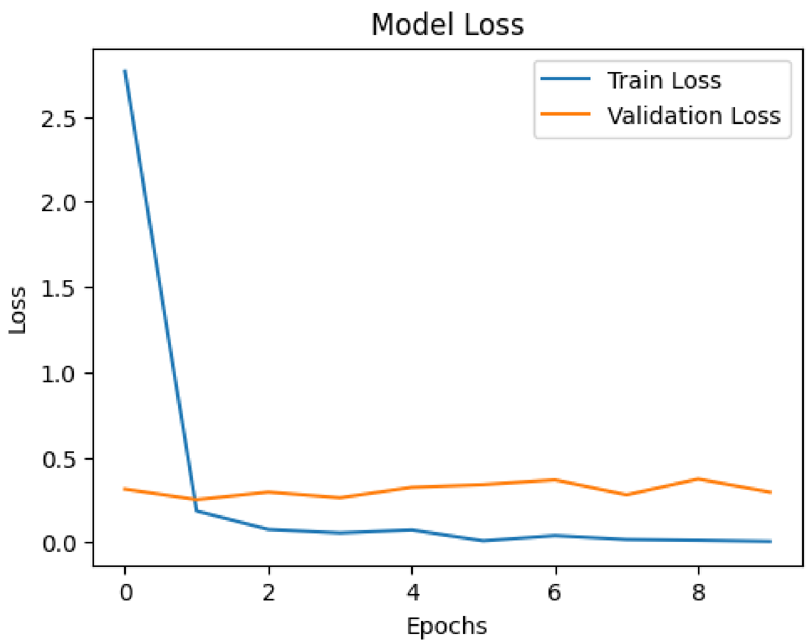
In addition to its efficiency, MobileNetV2 includes a feature called "linear bottlenecks," which helps prevent information loss during the network's forward pass. It employs skip connections to facilitate the flow of gradients during training, promoting faster convergence and better training stability.

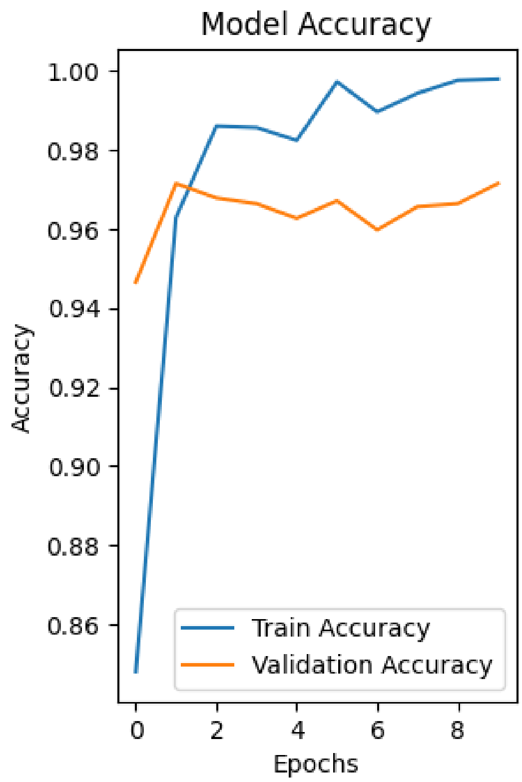
MobileNetV2 comes in various pre-trained models with different complexities, allowing developers to choose the right trade-off between speed and accuracy for their specific application. This adaptability is one of its significant advantages.

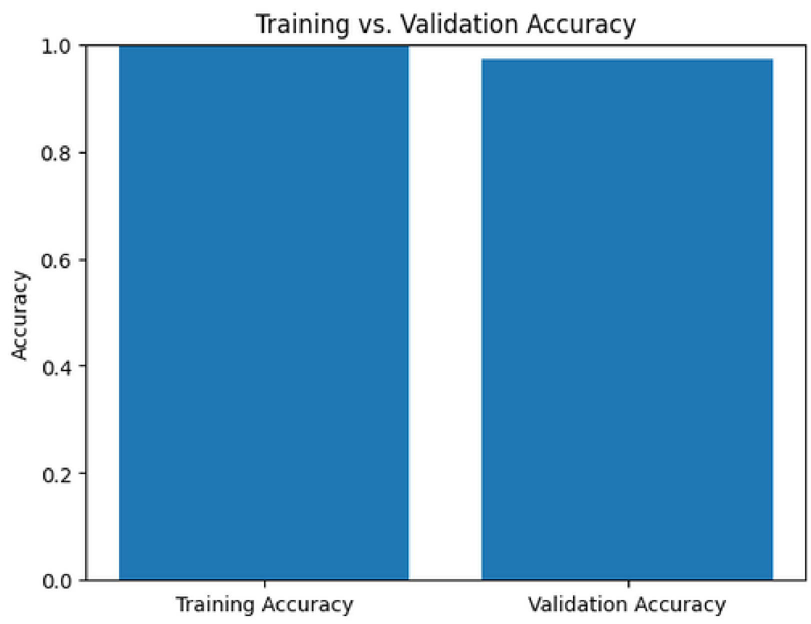
MobileNetV2 has achieved impressive results in benchmarking competitions, and it's often used as a backbone architecture in larger models for tasks like object detection and semantic segmentation. The network has a wide range of practical applications, including image and video processing, autonomous vehicles, and even augmented reality.

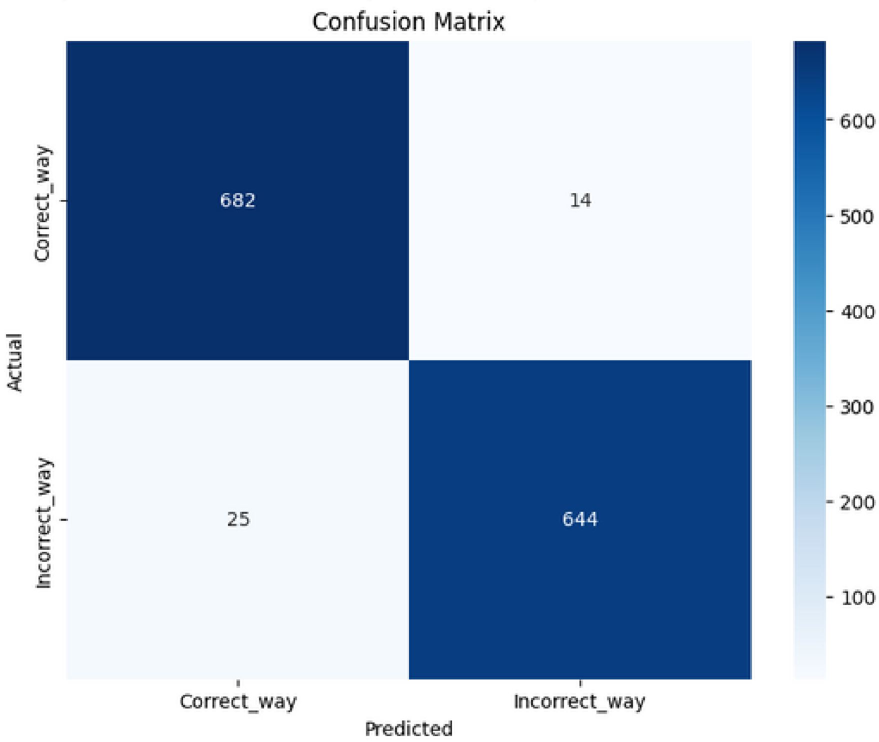


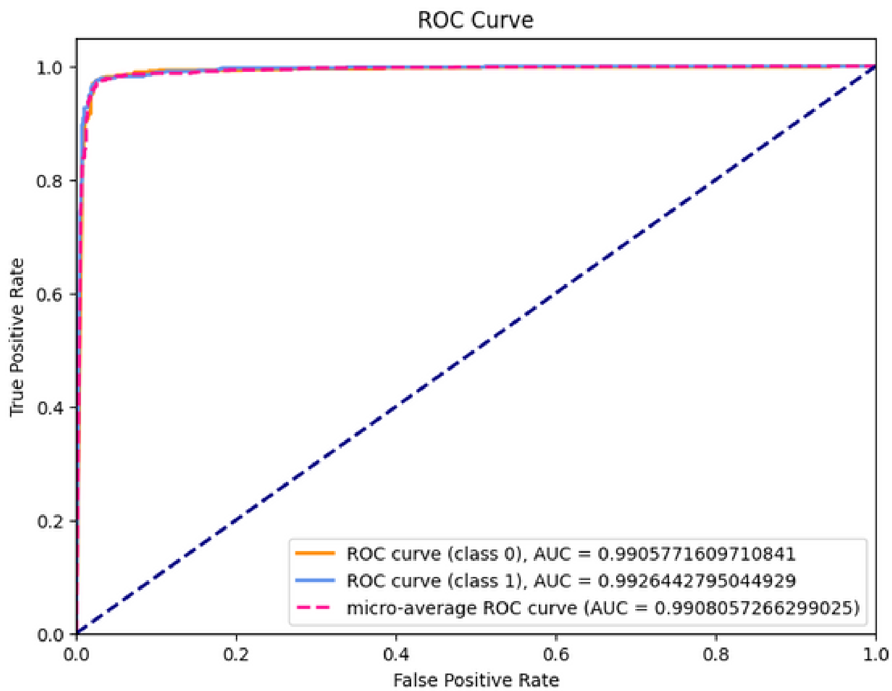

```
Final Training Accuracy: 0.9978
Final Validation Accuracy: 0.9714
```

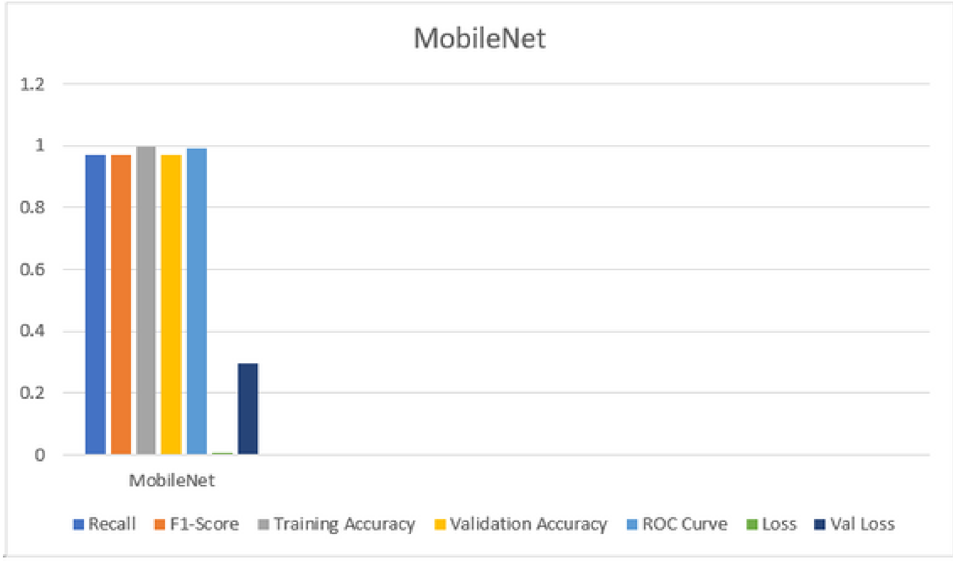



Model Accuracy
1.00
0.98
0.96
0.94
0.92
0.90
0.88
0.86
Accuracy
Train Accuracy
Validation Accuracy
0
2
4
6
8
Epochs


Training vs. Validation Accuracy
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Training Accuracy
Validation Accuracy


Confusion Matrix
682
14
25
644
Correct_way
Incorrect_way
Actual
Correct_way
Incorrect_way
Predicted
600
500
400
300
200
100


ROC Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (class 0), AUC = 0.9905771609710841
ROC curve (class 1), AUC = 0.9926442795044929
micro-average ROC curve (AUC = 0.9908057266299025)


MobileNet
1.2
1
0.8
0.6
0.4
0.2
0
MobileNet
Recall
F1-Score
Training Accuracy
Validation Accuracy
ROC Curve
Loss
Val Loss

VGG-16

VGG16 is a convolutional neural network architecture proposed by the Visual Geometry Group (VGG) at the University of Oxford. It was introduced in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition" by Karen Simonyan and Andrew Zisserman in 2014.

Key characteristics of VGG16:

1. **Architecture**: VGG16 is composed of 16 weight layers, including 13 convolutional layers and 3 fully connected layers. It follows a sequential architecture with alternating convolutional and max-pooling layers, followed by fully connected layers at the end.

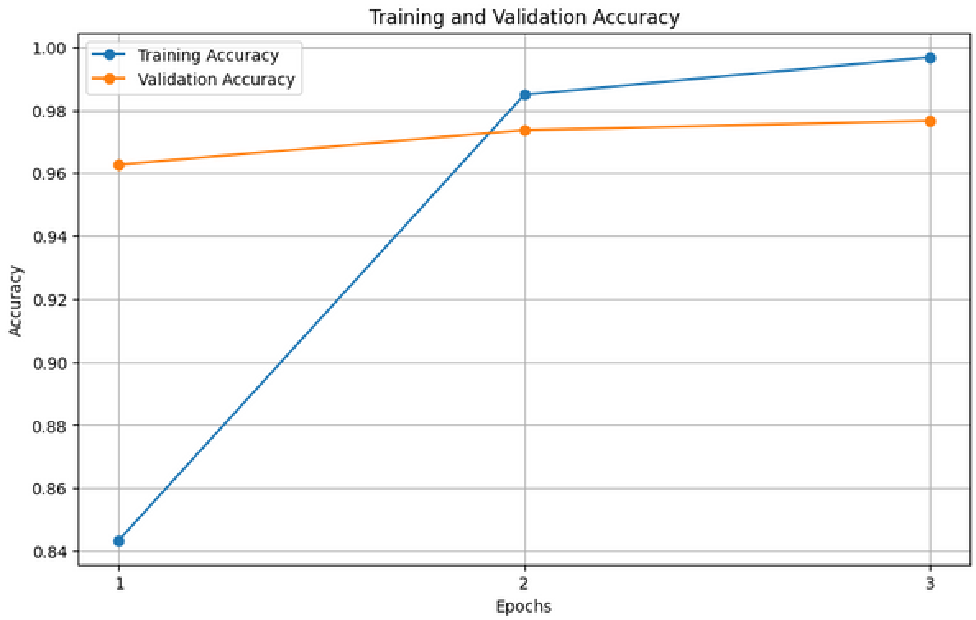
2. **Convolutional Layers**: The convolutional layers in VGG16 use small receptive fields (3x3 filters) with a stride of 1. Each convolutional layer is followed by a rectified linear unit (ReLU) activation function.

3. **Max Pooling**: VGG16 employs max-pooling layers with 2x2 filters and a stride of 2 after certain convolutional blocks. Max pooling helps in downsampling the feature maps and reducing spatial dimensions.

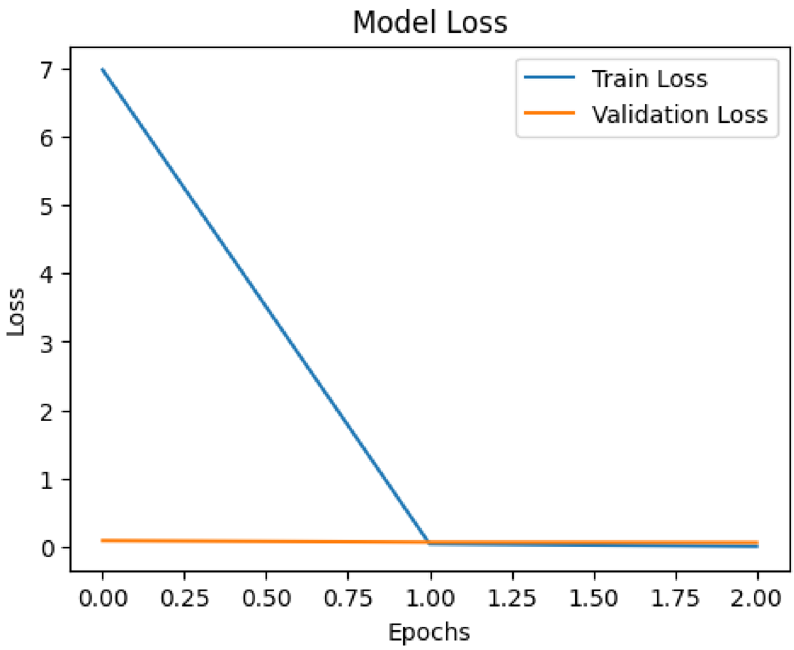
4. **Fully Connected Layers**: The convolutional layers are followed by three fully connected layers, each with 4096 units. The final layer consists of softmax units for classification.

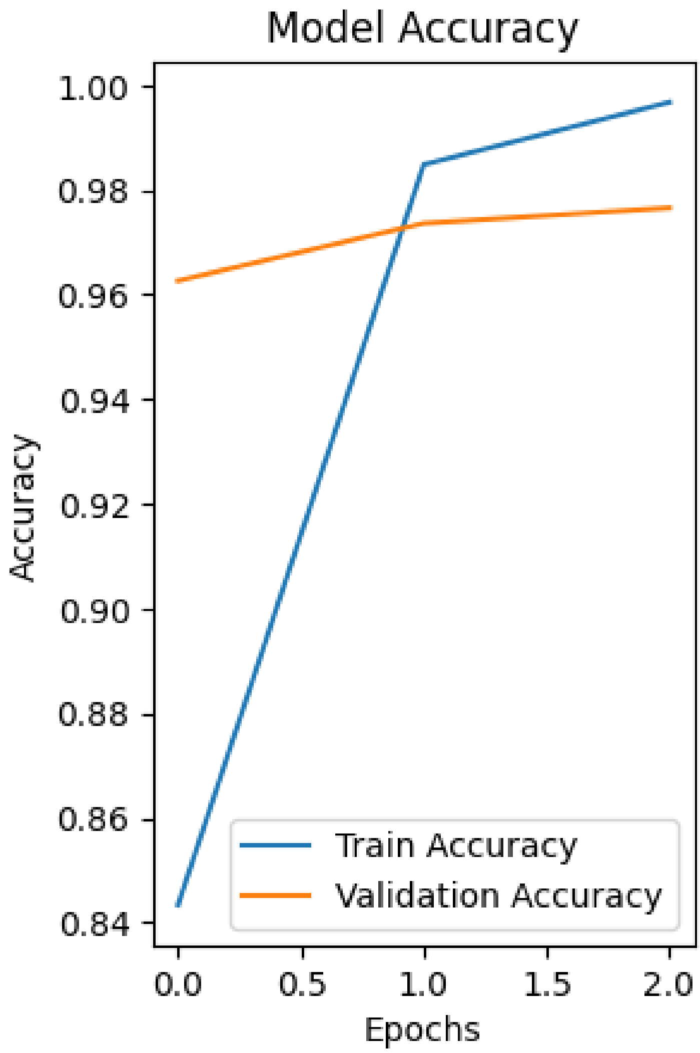
5. **Preprocessing**: VGG16 expects input images of size 224x224 pixels with three color channels (RGB). It is common to preprocess the input images by subtracting the mean RGB value computed on the training dataset from each pixel.

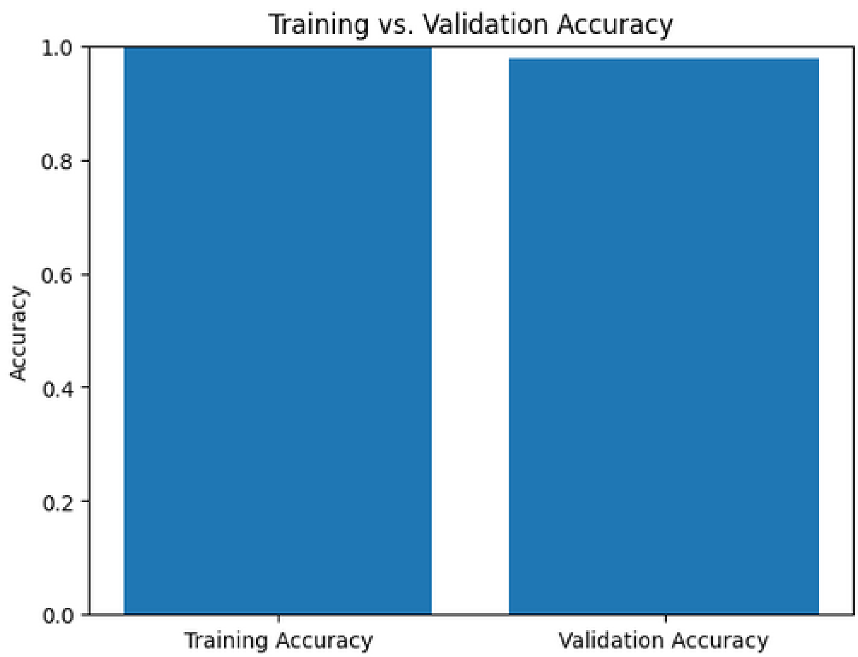
6. **Training**: VGG16 is typically trained on large-scale image classification datasets like ImageNet, which contains millions of images belonging to thousands of categories. It is often pre-trained on ImageNet and then fine-tuned for specific tasks or datasets.

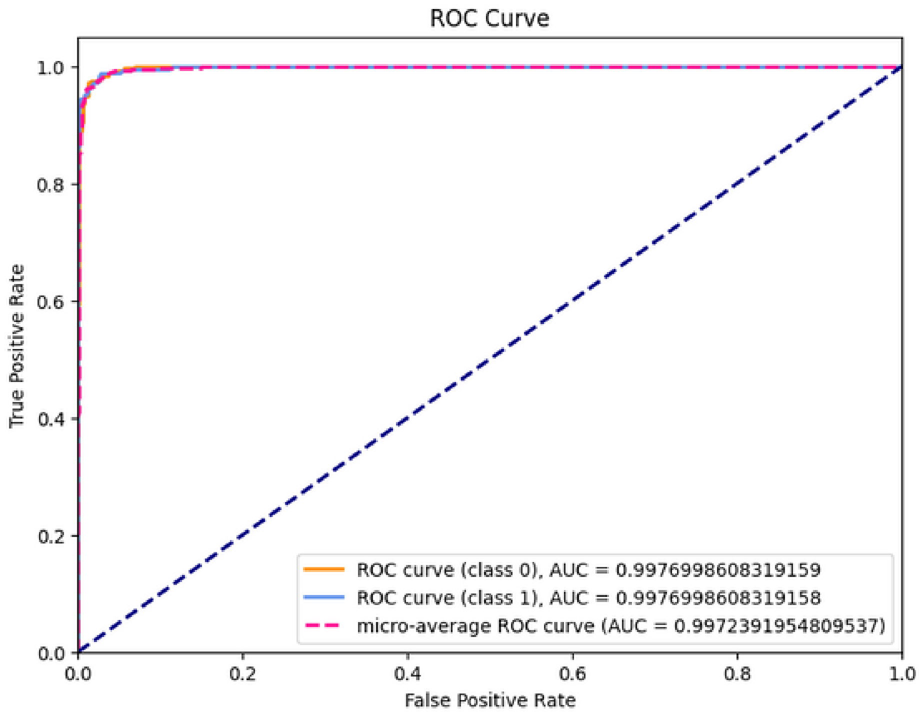



```
Final Training Accuracy: 0.9968
Final Validation Accuracy: 0.9766
```

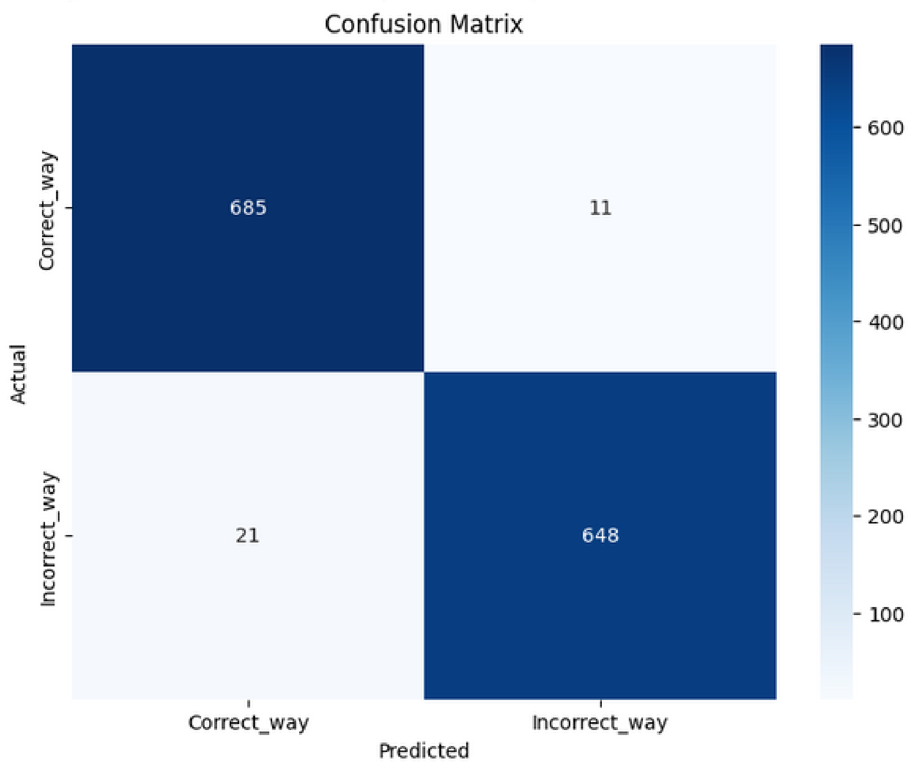



Model Accuracy
1.00
0.98
0.96
0.94
0.92
0.90
0.88
0.86
0.84
Accuracy
0.0
0.5
1.0
1.5
2.0
Epochs
Train Accuracy
Validation Accuracy

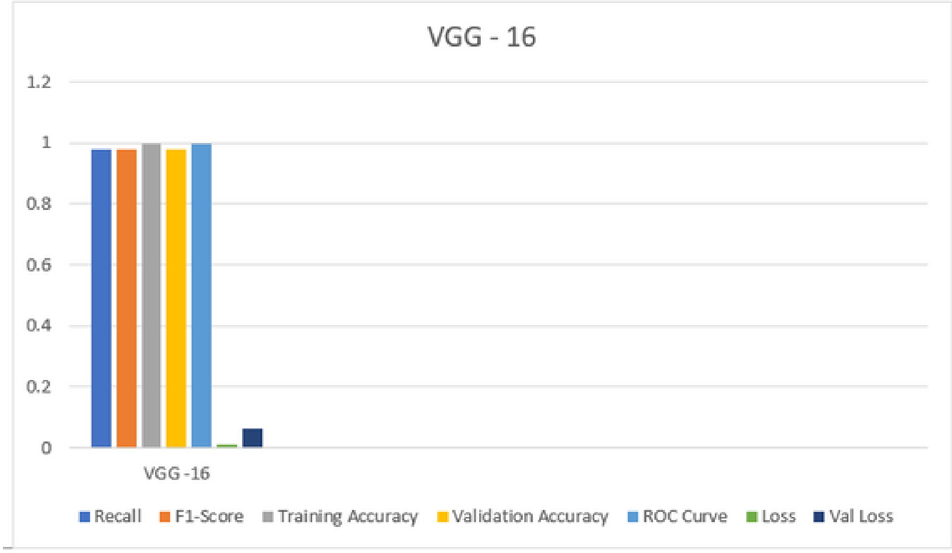

Training vs. Validation Accuracy
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Training Accuracy
Validation Accuracy



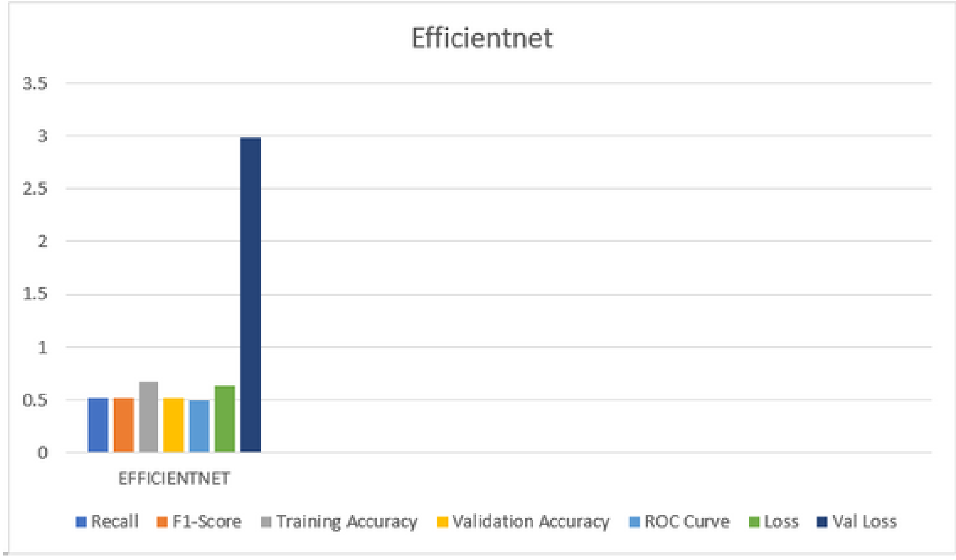
ROC Curve
1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
ROC curve (class 0), AUC = 0.9976998608319159
ROC curve (class 1), AUC = 0.9976998608319158
micro-average ROC curve (AUC = 0.9972391954809537)



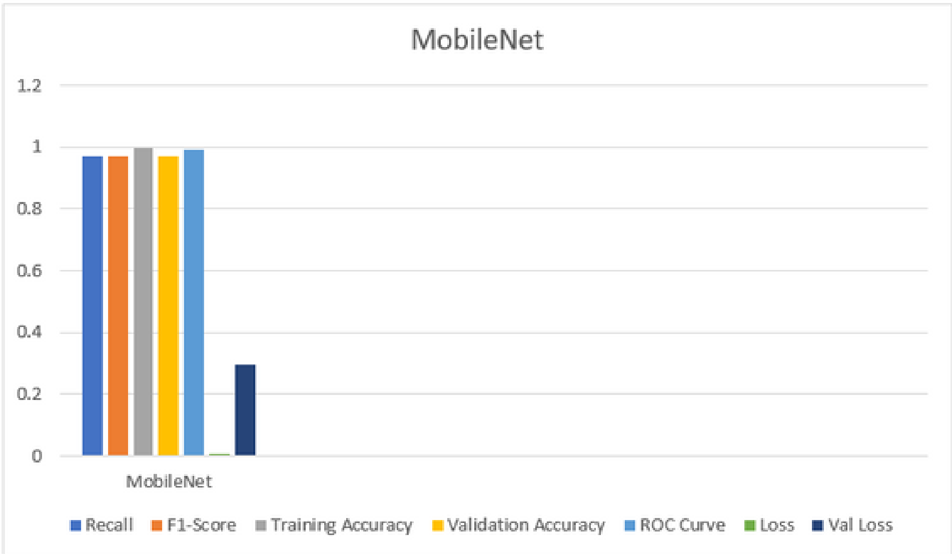
Confusion Matrix
Correct_way
Incorrect_way
Actual
685
11
21
648
Correct_way
Incorrect_way
Predicted
600
500
400
300
200
100


VGG - 16
1.2
1
0.8
0.6
0.4
0.2
0
VGG -16
Recall
F1-Score
Training Accuracy
Validation Accuracy
ROC Curve
Loss
Val Loss

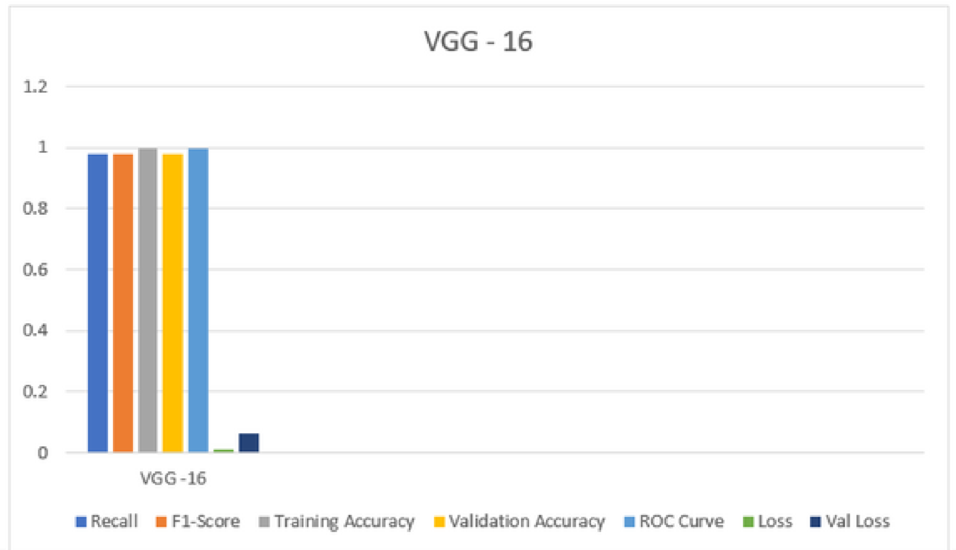
Comparison



Efficientnet
3.5
3
2.5
2
1.5
1
0.5
0
EFFICIENTNET
Recall
F1-Score
Training Accuracy
Validation Accuracy
ROC Curve
Loss
Val Loss



MobileNet
1.2
1
0.8
0.6
0.4
0.2
0
MobileNet
Recall
F1-Score
Training Accuracy
Validation Accuracy
ROC Curve
Loss
Val Loss



VGG - 16
1.2
1
0.8
0.6
0.4
0.2
0
VGG -16
Recall
F1-Score
Training Accuracy
Validation Accuracy
ROC Curve
Loss
Val Loss