Brain Tumour Detection Using MobileNetV2: Deep Learning For Brain Tumour Detection

March 23, 2025

Abstract

One of the most deadly health risks, brain tumors, need to be detected using effective and accurate detection methods to minimize the risks. Traditional MRI analysis takes time and needs specialised interpretation. Deep learning, particularly Convolutional Neural Networks (CNNs), has proven to be a good alternative for fast and accurate tumor detection. This article explores using pre-trained MobileNetV2 with transfer learning, for the automatic detection of brain tumors from MRI scans. The model is fine-tuned with custom layers for binary classification and achieved 90.62% accuracy on the test, had an AUC of 0.9909, and a test loss of 0.2775. The key outcomes are high classification accuracy (1.00) of the "YES" class and perfect recall (1.00) of the "NO" class, indicating outstanding classification ability with balanced precision, recall and F1 scores. The model's training and fine-tuning are optimized using learning rate scheduling, data augmentation and class weighting. The model's effectiveness makes it ideal for settings with limited resources. These results validate the potential of MobileNetV2 as an effective tool for early and precise diagnosis of brain tumors. Future work can involve a larger dataset, expanding the network to multi-class classification.

1 Introduction

One of the most deadly diseases is a brain tumor, specifically in children and those below 40 years of age, which makes the early diagnosis of brain tumors significant and increases the chances of living. For that reason, creating strategies to diagnose brain tumors early is necessary. Traditional brain tumor detection approaches mainly rely on manual inspection of Magnetic Resonance Imaging (MRI) information, which is time-consuming, has a high possibility of having a human mistake like wrong diagnosis (error-prone), and is sometimes unreliable.

To address these issues, the main objective of this report is to leverage MobileNetV2, to offer an effective brain tumor detection model. The primary task is to automate brain tumor detection and classification from MRI images.

Deep learning's prospects for recognizing intricate patterns and features from big data make it well-suited for image-based medical diagnosis. Improving Conventional Neural Networks (CNNs) fostered accuracy and reduced tumor diagnosis time, providing clinicians with a reliable and fast computer-aided diagnostic (CAD) system.

This study applies MobileNetV2 to automate brain tumor detection, achieving 90.62% accuracy, 0.9909 AUC, and 0.2775 loss. Results showed that the MobileNetV2 model, with good data augmentation and optimisation, offers a promising path toward improving the detection and classification of brain tumors. The result is good for positive cases and recall negatives, improved by data augmentation and transfer learning.

The report covers the MobileNetV2 model and compares different deep learning approaches in literature review, methodology, training of the model, achieved results from the given dataset, and future directions.

2 Related Work: Comparison of Deep Learning Approaches and Software Tools for Brain Tumor Detection

Deep learning has made significant strides in detecting brain tumors, using various convolutional neural network (CNN) architectures and software packages to deliver high accuracy and efficiency.

Traditional CNN architectures like AlexNet, VGGNet, and ResNet have proven excellent performance on brain tumor classification task [9]. VGG16 and ResNet, for instance, use deep layers to extract complex spatial features from MRI images and deliver high accuracy. Even though they are computationally costly, so they are less feasible to employ in real-time applications or low-resource settings [11],lightweight models like MobileNetV2 are rooted in the depth-wise separable convolution approach to drastically reduce the computational cost without compromising decent accuracy, to be optimal for deployment on cell phones or the edge [10], [3].

Similarly, Inception-v3 and DenseNet201 have also shown exemplary performance, with Inception-v3 achieving a maximum of 99.34% accuracy in some research.[8] However, due to their computational demand and huge model size, they are limited to actual implementation in resource-constrained settings [7]. Among software, TensorFlow and PyTorch are the foremost frameworks, with TensorFlow dominating deployment features and TensorFlow Lite and PyTorch offering flexibility and research ease with ease of usage [1].

Keras, a deep learning API on top of TensorFlow, facilitates easy rapid prototyping and supports fundamental techniques like data augmentation and batch normalisation, required for model performance optimization, particularly for coping with small datasets [2]. Other packages, such as Fast.ai, Caffe, and MXNet, have unique strengths involving speed, scalability, and the ability to test different experiments [4], [6].

Data augmentation techniques like rotation, flip, scale, and batch normalization play critical roles in enhancing model stability and generalization to ensure good performance in brain tumor detection applications [12], [5]. Finally, the application's requirements determine the architecture and tools, balancing accuracy, computing complexity, and deployment demands.

3 Proposed Method

This paper evaluates a model designed to enhance the efficiency and accuracy of brain tumor detection. The model's workflow, as illustrated in Figure 1, outlines the steps taken to achieve this goal.

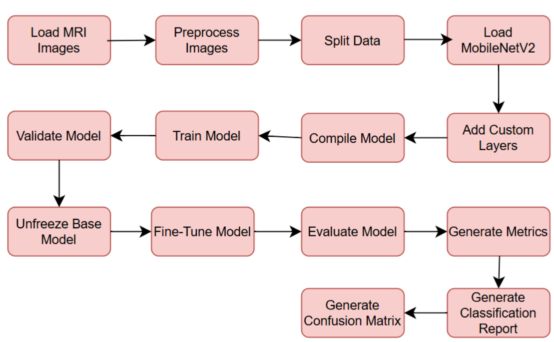

Figure 1: Model Workflow

In this section, a widespread neural network architecture, MobileNetV2 has been used to predict whether a patient has a brain tumor from brain MRI images from the dataset "brain_tumor_dataset." MobileNetV2 is chosen due to its strong feature extraction capabilities, making it ideal for medical imaging applications where labeled data is often limited.

3.1 Dataset Limitations

The dataset has 150 images labeled positive for the brain tumor(yes) and 73 images labeled as negative brain tumor(no), totaling 223 different images. This is a small dataset, which requires potential constraints and challenges such as the risk of overfitting, class imbalance and lower diversity. With the limited samples, the model can learn patterns specific to the training set instead of generalizing well to new data. This issue can affect model robustness when deployed in real-world applications. Apart from that, the difference between yes and no pictures leads to class imbalance and biased prediction towards the majority class (yes). To solve these issues, data augmentation techniques have been used, and class weights are computed and used during training so that the model pays bigger attention to the less frequent class, thereby increasing its ability to detect it.

3.2 Pre-processing Phase

Firstly, data is split into 70-15-15 train test value folders to ensure that the model is trained on large data. The Keras preprocessing is used in the model to preprocess MRI images, specifically through the "ImageDataGenerator" class. This includes rescaling to normalise pixel values to the range [0,1] by dividing by 25(rescale 1./255) and data augmentation. The data augmentation techniques have improved model generalisation and handling class imbalance.

3.3 Model Architecture

Unlike traditional CNNs, this lightweight convolutional neural network uses depthwise separable convolutions. The MobileNetV2 architecture is pre-trained on the ImageNet dataset. While the top classification layers are removed, some custom layers are added for binary classification.

Layer (type)	Output Shape	Param #
mobilenetv2_1.00_224 (Functional)	(None, 7, 7, 1280)	2,257,984
flatten (Flatten)	(None, 62720)	0
dense (Dense)	(None, 128)	8,028,288
batch_normalization (BatchNormalization)	(None, 128)	512
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 1)	129

Figure 2: Model Summary

As seen on the figure 2, the architecture consist of the following layers:

3.3.1 MobileNetV2

A pre-trained convolutional neural network using transfer learning.

3.3.2 Flatten Layer

The 2D feature map (7x7x1280) is converted into a 1D vector, preparing the data for fully connected layers.

3.3.3 Dense Layer

A fully connected layer with 128 units and ReLU activation, introducing non-linearity and learning complex patterns.

3.3.4 Batch Normalisation

Normalizes the outputs of the dense layer to stabilize and accelerate training.

3.3.5 Dropout Layer

Randomly drops 50% of the neurons during training to prevent overfitting.

3.3.6 Dense Layer

This final layer has a single unit and sigmoid activation. The dense layer produces a binary output for classifying tumors.

3.4 Training Phase

The training phase plays a crucial role, the choice of loss function and hyperparameters is critical in improving the model's performance. In this model, the binary cross-entropy loss function is used for the model's binary classification.

The "Adam" (Adaptive Moment Estimation) optimizer is used to get the most optimal loss function. Adam combines two optimization algorithms with their benefits: "AdaGrad" and "RMSProp." It adapts the learning rate for each epoch parameter individually, leading to faster convergence and improved performance.

Key hyperparameters, such as the initial learning rate (0.0001) and gradient clipping, were carefully tuned to prevent exploding gradients and ensure efficient training. The learning rate was set to 0.0001, a common choice for fine-tuning pre-trained models, as it allows for gradual weight updates without causing instability. In addition, a learning rate scheduler, was implemented dynamically. This adjusted the learning rate during training to adjust the learning rate based on validation loss, preventing the model from getting stuck in local minima.

The batch size was increased to 32 to balance computational efficiency and model stability during training. Data augmentation techniques, such as rotation, width and height shifts, shear, zoom, and horizontal flip, were applied to handle class imbalance and improve the model's ability to generalize to unseen data. The model also incorporated batch normalization and dropout (0.5) to reduce overfitting and stabilize the learning process.

The class weighting technique is applied during alongside the data augmentation to address the class imbalance in the dataset. With the "balanced" parameter from "sklearn.util.class_weight," data was inversely weighted based on their frequency.

4 Experimental Results

The model is deployed on the dataset. The proposed fine-tuned model, which is the base of MobileNetV2, is executed on JupyterNotebook using Python frameworks such as Keras and TensorFlow.

The model has been trained using hyperparameters to improve accuracy and performance. Epochs are 20 for training, followed by 10 epochs after fine-tuning.

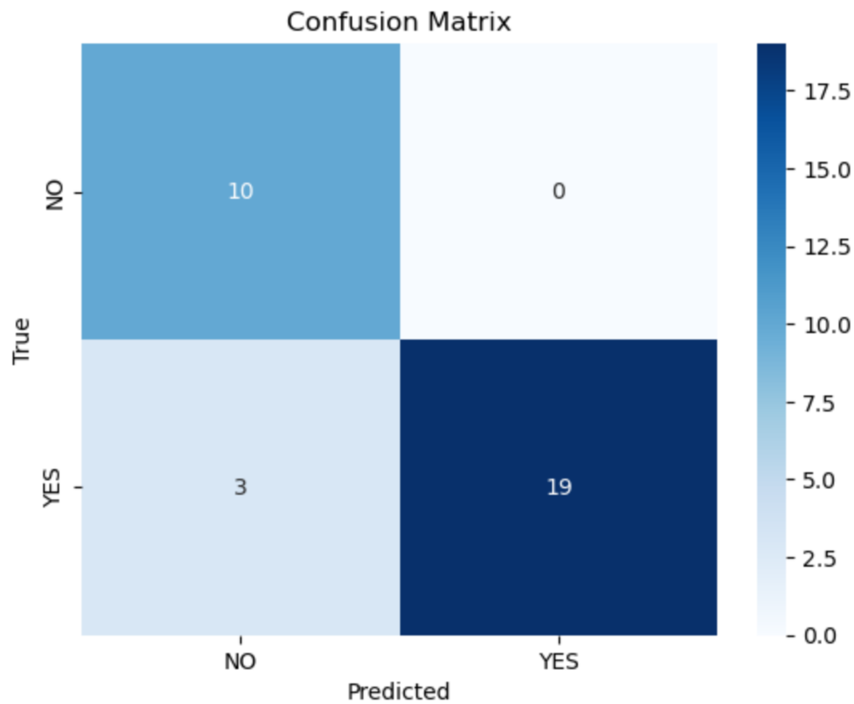

Figure 3: Confusion Matrix

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 998ms/step
              precision    recall  f1-score   support

          NO       0.77      1.00      0.87        10
         YES       1.00      0.86      0.93        22

    accuracy                           0.91        32
   macro avg       0.88      0.93      0.90        32
weighted avg       0.93      0.91      0.91        32
```

Figure 4: Classification Report

The results obtained from the classification reports show good predictive accuracy in determining if the MRI has a brain tumor, with a 90.62% test accuracy rate. The model's ability to distinguish between classes further supports its strong classification power, as evidenced by its Area Under the Curve (AUC) of 0.9909. A test loss measure of 0.2775 indicates that the model's prediction is exceptionally close to the actual labels, assuming low error. All these findings combined show that the model generalizes well to new data, a critical factor in medical real-world applications requiring reliable and accurate tumor detection.

Class-specific results provide more detail on the performance of the model. The "no" class had a precision of 77%, indicating some false positives, while the "yes" class had a flawless precision of 1.00, indicating that all positive predictions were correct. About recall, the "no" class reached 1.00, which means there were no false negatives, and the "yes" class had a slightly reduced recall of 0.86, meaning 14% of positive true cases were labeled as negative. The F1 measures, which balance precision and recall, were 0.86 for the "no" class and 0.93 for the "yes" class, demonstrating that the model performing highly well in general.

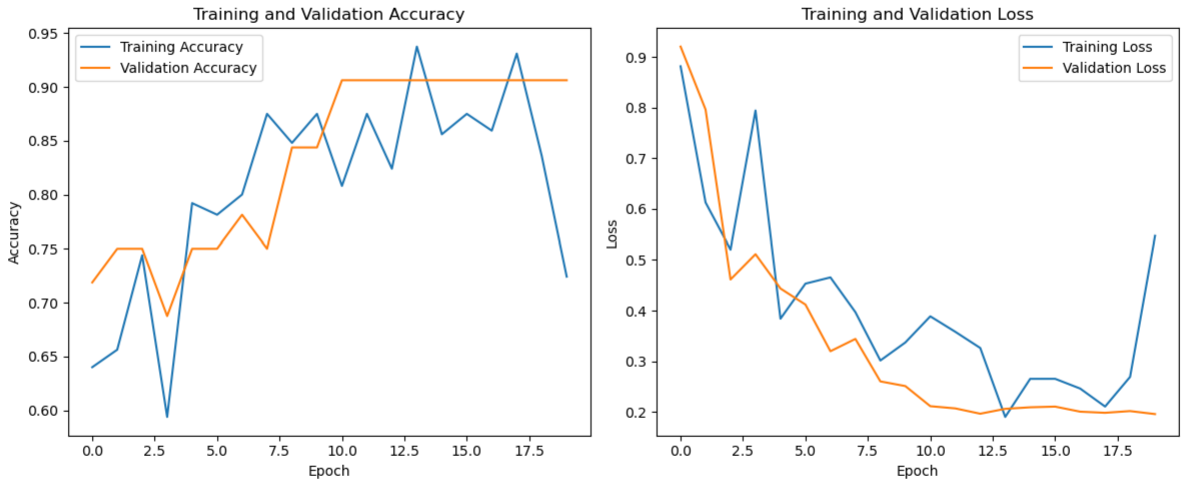

Figure 5: Training and Validation Accuracy and Loss Over Epochs

Furthermore, as seen in Figure 5, the training and validation curve accuracy and loss indicate that the brain tumor detection model performs well, with high accuracy and consistent improvement through epochs. The model also generalizes well to unseen data, and no compelling evidence of overfitting exists.

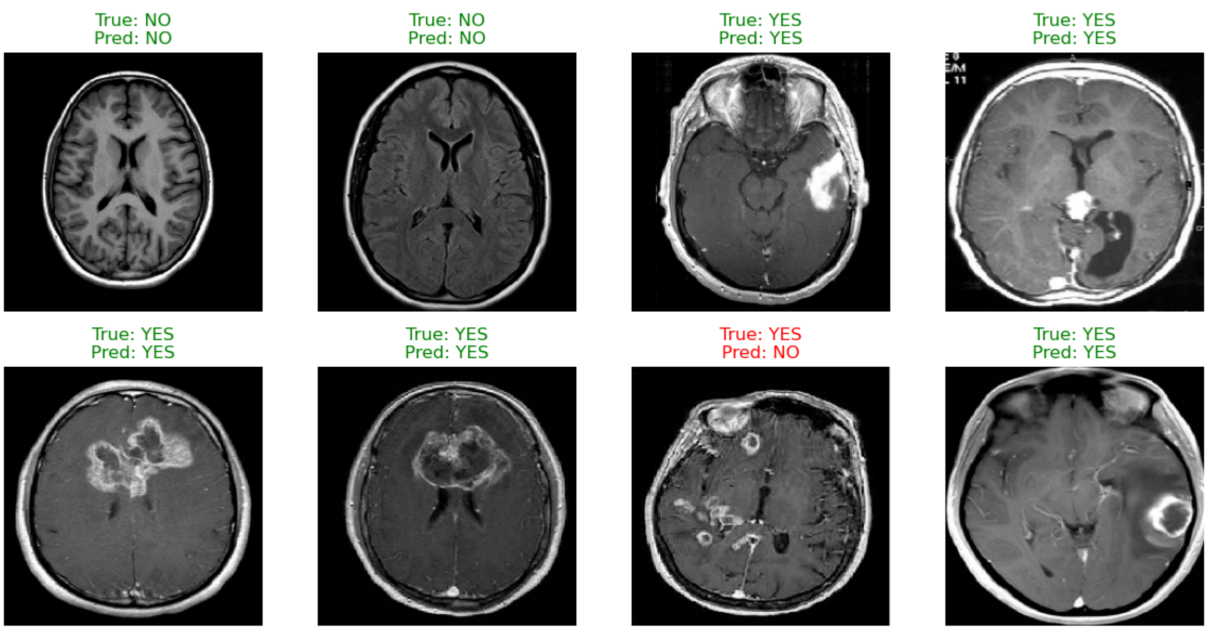

Figure 6: Labaled Data

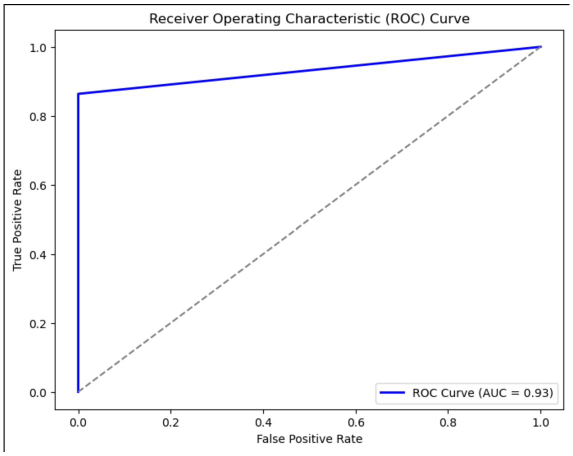

Figure 7: ROC curve

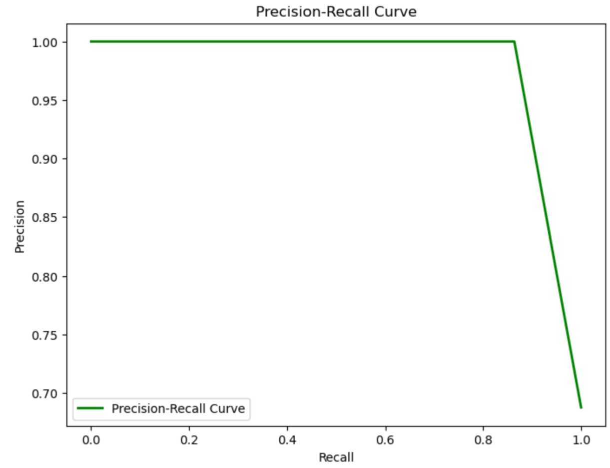

Figure 8: Precision-Recall Curve

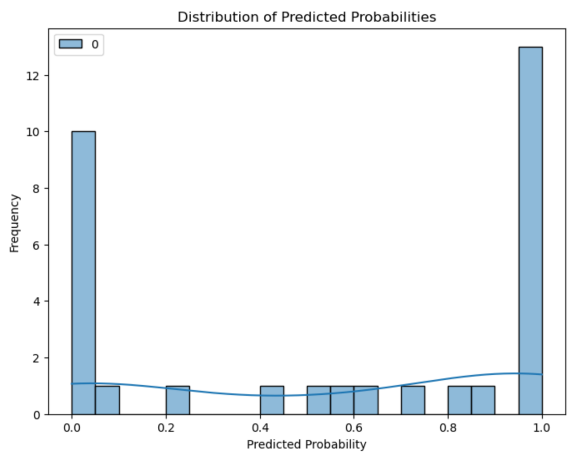

Figure 9: Distribution of predicted probabilities

The ROC curve shows excellent model performance with an AUC of 0.93, indicating strong class separation. The Precision-Recall curve highlights a good balance between precision and recall, which is crucial for imbalanced data. The predicted probabilities distribution confirms reasonable confidence in predictions. Overall, the model performs well for brain tumor detection.

5 Summary

This work analyses the automated brain tumor detection from MRI images using MobileNetV2. Indicating rather good performance in separating tumor and non-tumor cases, the model was found to have a test accuracy of 90.62%, an AUC of 0.9909, and a test loss of 0.2775. Strong classification abilities are shown by the key results, which show perfect recall (1.00) for the "NO" class and great accuracy (1.00) for the "YES" class. Together with data augmentation and transfer learning, the model's efficiency qualifies it as a perfect model for settings with limited resources. These results show MobileNetV2's promise as a consistent tool for exact early brain tumor diagnosis.

To enhance the models performance on generalisability and resilience user larger dataset with various MRI images could be beneficial. Edge or mobile platform model optimisation for real-time applications could make it deployable on systems with constrained resources. Further improvements of performance could come from more hyperparameter optimization and researching advanced regularising methods.

References

[1] Abadi, M., et al. Tensorflow: A system for large-scale machine learning. *OSDI* (2016).

[2] Chollet, F. Keras: Deep learning for humans. *GitHub Repository* (2015).

[3] Howard, A. G., et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* (2017).

[4] Howard, J., and Gugger, S. Fast.ai: A layered api for deep learning. *arXiv preprint arXiv:2002.04688* (2020).

[5] Ioffe, S., and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ICML* (2015).

[6] Jia, Y., et al. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093* (2014).

[7] Noreen, N., et al. Multi-level feature extraction and concatenation for brain tumor classification using inception-v3 and densenet201. *IJRITCC* (2021).

[8] Paszke, A., et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS* (2019).

[9] Rehman, A., et al. A deep learning-based framework for automatic brain tumor classification using mri images. *Journal of Medical Systems* (2020).

[10] Sandler, M., et al. Mobilenetv2: Inverted residuals and linear bottlenecks. *CVPR* (2018).

[11] Senan, E. M., et al. Brain tumor classification using resnet-18 with svm. *IEEE Access* (2021).

[12] Shorten, C., and Khoshgoftaar, T. M. A survey on image data augmentation for deep learning. *Journal of Big Data* (2019).