Deep Learning-Based Classification of Brain Tumour Detection Using Convolutional Neural Networks: A Comparative Study with Custom and Pre-trained Models

2441638

Abstract

Brain tumours present a significant threat to human life, requiring early and accurate diagnosis for effective treatment. In recent years, deep learning methods, particularly Convolutional Neural Networks (CNNs), have revolutionised the domain of medical image analysis. This project investigates the application of deep learning, specifically CNN, for automated brain tumour classification from MRI scans. A custom CNN model is developed and compared against pre-trained architectures such as VGG16, ResNet50, and EfficientNetB0 to evaluate performance metrics such as accuracy, precision, F1-score and recall. A dataset comprised of positive (tumour) and negative (non-tumour) classes of MRI brain images was used. The study includes data preprocessing techniques such as Contrast Limited Adaptive Histogram Equalisation and brain contour cropping to enhance feature extraction. A custom-built CNN was trained using optimised parameters and evaluated through key metrics including accuracy and confusion matrices. Comparative insights into traditional machine learning classifiers and alternative deep learning strategies are presented to substantiate the choice of architecture. The results demonstrate that the custom CNN achieves competitive performance, while pre-trained models benefit from transfer learning. The findings suggest that deep learning models can significantly improve diagnostic efficiency, with potential applications in clinical decision support systems. detection using machine learning

Keywords: Deep Learning, Convolutional Neural Networks, MRI Brain Tumour Analysis, Computer Vision, Classification, Transfer Learning, VGG16, ResNet50, EfficientNetB0, Image Processing, Performance Evaluation, Machine Learning

Contents

1. Problem Statement

Brain tumours, both cancerous and non-cancerous, are among the most prevalent types of tumours. Early and accurate diagnosis is essential for effective treatment and better prognosis. Brain tumours are diverse and can originate from intracranial tissues and meninges, with varying degrees of malignancy, from benign to aggressive.

In the UK alone, around 4,400 new cases of brain tumours are diagnosed each year, with an incidence rate of 7 per 100,000 people, making it one of the more common tumour types (McKinney, 2004). Traditional diagnostic methods are time-consuming and subjective. Survival rates for people with malignant brain tumours vary significantly, depending on factors such as age and tumour type.

Medical professionals have struggled with detecting brain tumours using human interpretation alone, highlighting the urgent need for more reliable, early detection methods. One promising solution is Computer-Aided Diagnosis (CAD). CAD uses feature extraction from medical images to distinguish between healthy and abnormal tissue (Tiwari et al., 2020; Jayade et al., 2019; Mahmood and Abbas, 2016).

Machine learning (ML), particularly image classification, plays a central role in medical diagnostics. By training systems on classified medical images, ML methods can act as expert systems to supplement diagnosis and teaching. In the case of brain tumours, image classification involves preprocessing and feature extraction from MRI scans to identify and classify tumour types.

Several classification techniques, including Probabilistic Neural Network, K-Nearest Neighbors, Artificial Neural Network, and Support Vector Machine, can be applied to medical image datasets for tumour detection (Soofi and Awan, 2017). Among these, deep learning (DL) has emerged as the state-of-the-art method, gaining widespread attention due to its ability to process large volumes of unstructured data. DL's power lies in its ability to extract features automatically across multiple layers, with CNNs being particularly effective for medical image analysis (Işın et al., 2016).

This project aims to tackle the challenge of automating brain tumour detection in MRI images using DL techniques such as convolutional layers, data augmentation, loss function and transfer learning. The goal is to develop and evaluate a CNN-based binary classifier that can accurately distinguish between tumour-positive and tumour-negative cases. The model's performance will be benchmarked against other pre-trained robust models (VGG16, ResNet50, and EfficientNetB0) to assess and evaluate its effectiveness.

The project follows a supervised learning framework, utilising pre-labelled MRI images for training and validation. It involves stages such as preprocessing, data augmentation, model training, and performance evaluation.

The expected outcome of this study is to develop a robust CNN-based binary classifier that can accurately distinguish between tumour-positive and tumour-negative MRI scans. By comparing the performance of the CNN model with pre-trained models like VGG16, ResNet50, and EfficientNetB0, the study aims to assess the effectiveness of this model in the context of brain tumour detection. The success of this research could lead to the development of a CAD system that can assist radiologists in making quicker, more accurate diagnostic decisions, improving patient outcomes through early detection and treatment.

2. Dataset

2.1. Source

The dataset is sourced from Kaggle Brain Tumour Dataset (Pawar, 2023), and consists of MRI scans classed: 'Positive' (indicating the presence of a brain tumour) and 'Negative' (indicating the absence of a tumour). The dataset is organised in a folder hierarchy, with separate directories for each class.

2.2. Ethical considerations

Anonymised and publicly available data is used in this study. No identifiable information is included, complying with privacy regulations and Data Protection. The findings are intended for academic purposes and not commercial use.

2.3. Preprocessing

To ensure consistency across the dataset, all images were resized to 240 x 240 for compatibility with the custom CNN and the pre-trained models due to CNNs requiring fixed dimensions for processing. Standardised image dimensions reduce computational load and ensure consistent feature extraction. In particular, for models like VGG19, which were trained on ImageNet images of the same size, preserving this input shape ensures that the convolutional filters behave as expected (Alzubaidi et al., 2021).By making all the images the same size, the model can easily extract features from each one, as they all have the same number of pixels. This consistency allowed the model to learn patterns more effectively. Additionally, processing the images in batches of 32 sped up training and simplified calculations, helping the model converge faster. The uniform image size reduced memory usage and processing power, leading to quicker training times without sacrificing performance.

To further enhance the model performance, the following steps were applied:

Contrast Enhancement
Contrast Limited Adaptive Histogram Equalisation (CLAHE) was applied to enhance local features such as tissue boundaries, lesions, and tumour regions that may otherwise be too subtle to distinguish by normalising image luminance. This is useful for medical imaging, where important information can be embedded in low-contrast regions. Prior study by (Santanu and Vibhuti, 2023) demonstrated the effectiveness of CLAHE in improving visual saliency in MRI and CT scans, thereby helping CNNs learn more discriminative features.

Brain Contour Cropping
All images were cropped to emphasise the region of interest (brain) and remove irrelevant areas that do not contribute to classification 2. This was important for eliminating the black outskirts commonly present in MRI images, which do not contain useful information and can introduce noise during training. Focusing on the central brain structures, cropping helped the model concentrate on the most informative regions, enhancing its ability to learn relevant patterns. This preprocessing step also reduced the input size, which decreased computational load and contributed to more efficient and accurate model training.

Data Augmentation
Augmenting images addresses dataset imbalances and enhances the model's ability to generalise. Augmentation techniques such as rotation, flipping, scaling, and elastic deformations introduce variability into the training process, allowing the model to learn robust features that are invariant to these transformations. This is applicable in medical imaging, where datasets are limited, and models may overfit to specific patterns. Peer-reviewed studies, like those by (Shin et al., 2016; Kim and Bae, 2020) have demonstrated that models trained with augmented data outperform those trained on non-augmented data, as the augmented models can better generalise to unseen examples. This reduces biases associated with the limited size and homogeneity of medical datasets and improves the model's accuracy in classifying images.

Normalisation
Image pixel values range from 0 to 255, which can cause issues during model training due to large input values. Images were normalised to a [0,1] scale by dividing pixel values by 255. This is a standard DL practice that ensures faster and more stable gradient descent during training. Without normalisation, the network may struggle with slow convergence or suboptimal weight updates due to large input values. Normalising the input data allowed the model to learn more effectively, leading to faster and more stable training.

Class Balancing via Oversampling and Augmentation
In the dataset used, the 'Positive' class is overrepresented 3. Without class balancing, the model would be biased towards the majority class, leading to poor generalisation. By using random oversampling and data augmentation, minority class instances were artificially increased, creating a more balanced distribution. This not only prevents bias during training but also acts as regularisation, reducing overfitting and encouraging the model to generalise better to unseen data. Class imbalance is one of the leading causes of poor sensitivity in medical classification tasks, and oversampling strategies are used to mitigate this.

Train-Test Split
A 80:10:10 stratified split was employed to preserve the proportion of each class across training, validation, and testing sets. This ensures that the model is not disproportionately exposed to certain classes, allowing for a fairer evaluation of performance across all categories. Stratification has been shown to significantly improve model robustness in imbalanced datasets.

7. Batch Training with Class Distribution Preservation
During training, data was loaded in batches of 32, with stratification enforced within each batch to reflect the global class distribution. This strategy ensures that every batch contributes to learning all classes, preventing the model from learning to favour the majority class early in training.

Dataset Statistics

Table 1: Dataset Statistics

Class	Number of Images
Positive (1)	3,266
Negative (0)	2,000
Total Images:	5,266

2.4. Data Visualisation

A exploratory data analysis revealed that the dataset suffers from class imbalance, with the 'Positive' class overrepresented 3. To mitigate this and enhance the model's robustness, data augmentation was applied using TensorFlow's Sequential API.

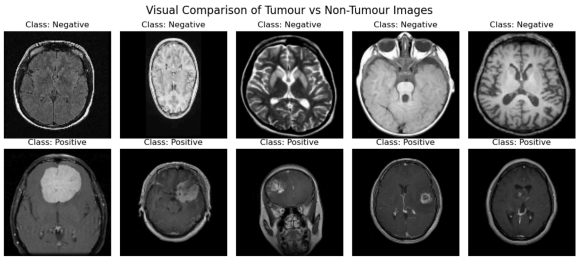

Figure 1: Visualising images with and without tumours, we can analyse the features within our dataset. Many of the images include large black regions that lack relevant information, and the overall appearance is quite uniform. To address this, data augmentation is applied, ensuring that the model doesn't solely learn from these uniform patterns and is able to generalise more effectively

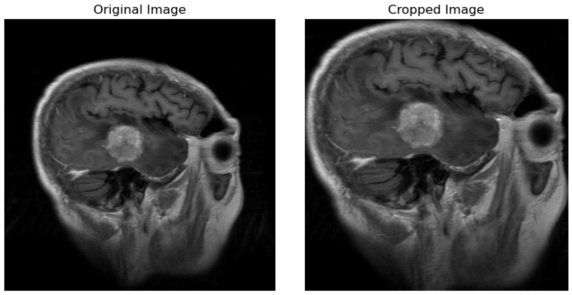

Figure 2: Sample MRI Scan: Before and After Cropping. We can observe an image from our dataset both before and after cropping. With the cropped image, the model can now focus more effectively on the region of interest

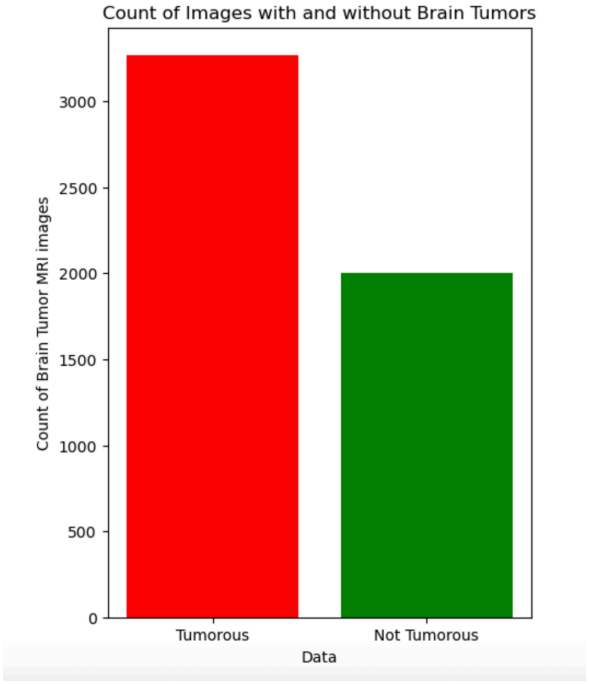

Figure 3: Class Distribution: The dataset shows that the 'Positive' tumorous class is significantly larger than the 'Negative' non-tumorous class. To address this imbalance, class weighting is applied to ensure that the model does not become biased towards the more prevalent class and can learn to generalise effectively across both classes

3. Algorithm

3.1. Model Selection and Justification

To solve the problem statement, both a custom CNN and three state-of-the-art pre-trained models (VGG16, ResNet50, and EfficientNetB0) were selected. The rationale behind this hybrid selection was to contrast lightweight, domain-specific architectures against computationally advanced models developed through large-scale training, thereby determining the most clinically and computationally viable solution. A pre-trained model is a ML or DL model that has already been trained on a large dataset for a specific task, and is then used as the starting point for a new task. Rather than training a model from scratch, which can be time-consuming and computationally expensive, a pre-trained model leverages the knowledge it has learned from the original dataset to perform similar tasks.

Benefits of comparing Custom and pre-trained Models include:

- **Performance Benchmarking:** Helps to evaluate how well the custom model (designed for the task) performs against pre-trained models (trained on large, diverse datasets).
- **Generalisation and Transferability:** Pre-trained models, trained on large datasets, learn generalised features (e.g., edges, textures, shapes) useful for detecting abnormalities like tumours, demonstrating the power of transfer learning.
- **Clinical Viability:** The comparison helps identify the most clinically feasible model, focusing on the balance between accuracy, speed, and resource usage, ensuring that the chosen model meets clinical requirements for timely and accurate tumour detection.

3.2. Custom CNN architecture

The custom model was designed as a lightweight 6-layer architecture consisting of sequential Conv2D, MaxPooling, Batch Normalisation, and Dropout layers. This model serves as a baseline, enabling us to assess the effectiveness of a simplified architecture crafted specifically for medical imaging tasks. Its streamlined structure was tailored for environments with limited computational capacity, such as mobile health applications or

edge devices. While not expected to match the precision of deeper networks, the model aims to demonstrate whether efficient performance can be achieved without the overhead of large-scale pre-training.

Architecture Summary

- **Input** $240 \times 240 \times 1$ (Greyscale images).
- **Convolutional Layers** ReLU activation and Batch normalisation.
- **Max Pooling** After each convolution for dimensionality reduction.
- **Dropout Layers** to prevent overfitting.
- **Fully Connected Layers with Softmax activation for classification.**
- **Loss Function** Categorical Crossentropy.

3.3. Pre-Trained Models

Three pre-trained CNNs were selected due to their proven capabilities in medical image classification tasks when adapted via transfer learning:

1. **VGG16** A deep and sequential model with a uniform architecture that has shown reliable performance across various image recognition tasks. Despite its computational weight and lack of architectural innovations like skip connections, VGG16 is known for capturing hierarchical features well (Mascarenhas and Agarwal, 2021). Fine-tuning the top layers allows it to specialise to the domain-specific textures of MRI scans.

2. **ResNet50** This model introduced residual connections to alleviate vanishing gradient problems, enabling deeper and more expressive networks. Its ability to maintain gradient flow through identity mappings makes it particularly effective in extracting subtle features from medical images (Mascarenhas and Agarwal, 2021).

3. **EfficientNetB0** Leveraging a novel compound scaling approach that simultaneously balances network depth, width, and input resolution, EfficientNetB0 is engineered for both efficiency and accuracy (Kumar et al., 2024). Its performance in this study serves to validate whether newer architectures designed with scalability in mind can outperform more traditional CNNs even on relatively small datasets.

Transfer Learning Approach

Transfer learning was applied uniformly across the pre-trained models. Lower-level convolutional layers were frozen to retain generalised feature extractors, while top layers were fine-tuned to capture domain-specific characteristics of brain tumour scans. The last fully connected layer was replaced for binary classification.

4. Training

All models were trained using the Adam optimiser, with an initial learning rate of 0.001. Adam, short for Adaptive Moment Estimation, is an optimisation algorithm that combines the benefits of both the AdaGrad and RMSProp methods (Zhang, 2018). It adjusts the learning rate for each parameter, which helps to stabilise training, especially in sparse gradient problems like those encountered in biomedical image analysis.

The loss function used was categorical cross-entropy, which is appropriate for multi-class classification problems where outputs are softmax-normalised probabilities, and the targets are one-hot encoded. A loss function quantifies the difference between predicted and true labels, guiding the model to improve its predictions over time during training (Gordon-Rodriguez et al., 2020). For classification tasks, categorical cross-entropy is ideal because it measures the dissimilarity between the predicted class probabilities and the actual class labels.

To enhance generalisation and address the limited diversity in the training dataset, data augmentation techniques were incorporated. Each image underwent random horizontal flips, zooms (up to 20%), and rotational shifts of ±15 degrees. These augmentations aimed to mimic the variability encountered in real clinical settings while preventing overfitting, a strategy supported by the findings of Wong et al., (2016). Data augmentation helps by artificially increasing the size and diversity of the training set, thus improving the model's ability to generalise to new, unseen data.

Training was conducted with a batch size of 32, striking a balance between GPU memory efficiency and gradient stability. A total of 25–50 epochs were allowed, with early stopping implemented (patience = 5 epochs) based on validation loss to prevent overfitting and reduce unnecessary training time. Early stopping monitors the model's performance on the validation set during training and halts training if performance stops improving, thus preventing overfitting and saving computational resources.

The training environment included an Apple M4 Pro processor with 16GB of unified memory, which enabled rapid iteration and tuning across different architectures. Model training times ranged from

2 to 14 minutes depending on architecture complexity and fine-tuning requirements, with the custom CNN completing in under five minutes and ResNet50 requiring the longest due to its depth.

5. Results Analysis

Evaluating the fine-tuned models, EfficientNetB0, VGG16, ResNet50 and the custom CNN for binary classification of MRI scans with and without tumours; Notable differences in computational efficiency, performance and generalisation ability were observed. The evaluation is based on standard performance metrics (accuracy, precision, recall, F1-score), training dynamics, computational costs and model complexity gives insight in to their clinical and practical ability.

EfficientNetB0 had strong overall performance achieving an accuracy score of **79.4%**, an precision of **76%** and an F1-score of **85%** across 1054 validation images for class 1 ('Positive'). However, the confusion matrix revealed a notable imbalance in sensitivity, with a marginal drop in recall for class 0 ('Negative') hinting at slight sensitivity to class imbalance 14. Although EfficientNetB0 performed robustly in detecting tumours, it occasionally misclassified healthy scans, an important consideration for clinical deployment where false positives carry economic and psychological penalties. This proves that, it's strength lies in its compound scaling strategy, which systematically adjusts depth, width, and input resolution to maximise accuracy while maintaining computational efficiency. As demonstrated by Tan and Le (2019), this can enable EfficientNetB0 to outperform much deeper networks using fewer parameters. In this study, fine-tuning the top 12 layers allowed it to reach high accuracy, precision, recall, and F1-score. The validation loss curve remained low and stable throughout training, reflecting strong generalisability and minimal overfitting 4. Overall, the model demonstrated exceptional robustness, making it suitable for integration into clinical diagnostic pipelines keeping in mind its sensitivity to input resolution and augmentation strategies needing careful calibration during training. Future iterations could explore variants like EfficientNetB1–B3 for performance gains or domain-specific pretraining on large-scale medical datasets such as BraTS or TCIA.

ResNet50, leveraging its residual connections to mitigate the vanishing gradient problem, demonstrated stronger results with **98%** accuracy, a precision of **96%** for class 0 and **99%** for class 1, and a recall of **98%** and **97%** respectively, and an overall F1-score of **98%**. Its confusion matrix reflected a balanced and confident performance across both classes. This makes it suitable for applications where sensitivity and specificity are paramount.

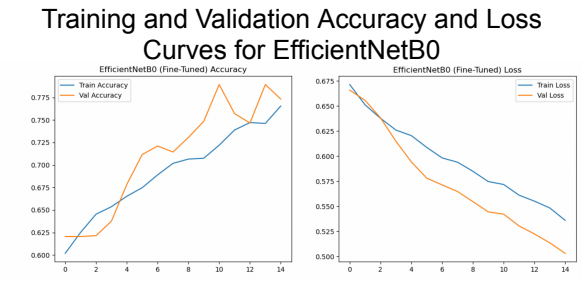

Figure 4: The training and validation curves for EfficientNetB0 demonstrate consistent convergence with minimal gap between training and validation accuracy. The low and stable loss indicates strong generalisation capacity and minimal overfitting throughout the fine-tuning process.

The loss curve declined consistently with minimal variance between training and validation, highlighting reliable learning dynamics 5. These results reflect its effectiveness in identifying tumour patterns, even in cases where features were subtle or complex. The model's performance was also more balanced across classes, with class-wise support and confusion matrix metrics indicating even representation. These outcomes are consistent with prior research highlighting ResNet's robustness in medical image classification tasks involving fine-grained abnormalities. The trade-off, however, lies in its longer training time and greater computational load. While manageable on a powerful workstation, its deployment in environments with limited resources may be constrained. Future work could consider augmenting ResNet50 with channel or spatial attention blocks (e.g. Squeeze-and-Excitation or CBAM) to enhance its ability to focus on tumour-relevant regions without increasing depth.

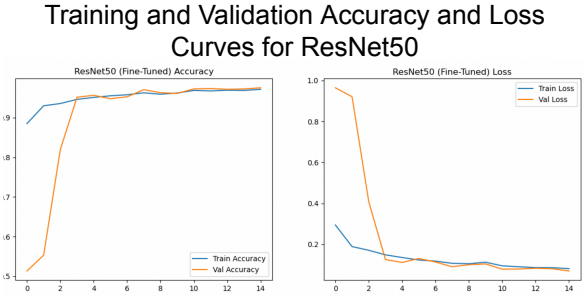

Figure 5: ResNet50 exhibited steady learning dynamics with tightly aligned training and validation curves, suggesting reliable generalisation. The gradual decline in loss across epochs, coupled with stable accuracy, reflects the stabilising effect of residual connections during fine-tuning.

VGG16, although a much older architecture, demonstrated remarkable competitiveness after

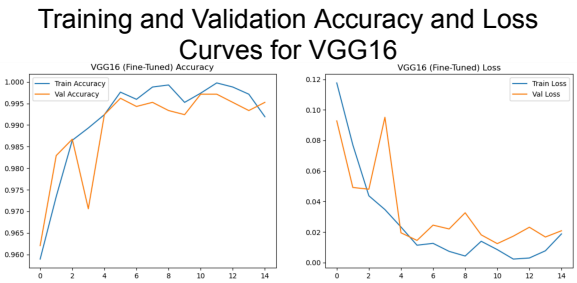

Figure 6: The training and validation curves for VGG16 indicate good performance but with minor fluctuations in validation loss, suggesting intermittent overfitting. Despite achieving high accuracy, the model's deeper, sequential structure posed some challenges for regularisation compared to more modern architectures.

fine tuning it's final eight layers. It reached near perfect classification metrics with **100%** precision for class 0 and **99%** for class 1 and a F1-score consistently above **99%**. It achieved a high recall especially for class 1, making it invaluable in medical screening scenarios where minimising missed tumours is critical. Despite these competitive scores, the training dynamics were less consistent 6, where minor oscillations in validation loss suggest intermittent overfitting. The confusion matrix revealed only 5 misclassified samples from class 0, indicating decent sensitivity for negative cases. While its architecture is now considered less efficient compared to more recent innovations, VGG16 remains a reliable benchmark due to its consistent performance and ease of implementation. Its sequential structure, however, makes it more susceptible to overfitting and slower convergence, limiting its scalability in larger datasets or real-time systems. That said, VGG16's consistent behaviour and relatively high recall still make it a dependable fallback model, particularly when ease of implementation and stability are prioritised.

The custom CNN, although shallower and devoid of pre-training, performed admirably. Its design, tailored specifically to the problem statement, achieved **99%** precision, recall, and F1-score closely rivalling VGG16 and ResNet50. The model's simplicity contributed to faster training and lower resource demands, without a substantial trade-off in performance. This affirms findings from recent studies that advocate for the viability of lightweight models in medical image analysis, especially in low-resource settings where deployment constraints are a key concern (Ukwandu et al., 2022; Kumar et al., 2023). Its interpretability and reduced complexity further enhance its practical appeal for clinical applications. However, its shallowness limited its ability to extract deeper spatial features, which could impact performance on more ambiguous or fine-grained cases. However, the custom CNN's shallowness led to some limitations in capturing fine-grained features, resulting in slightly lower performance on subtle or ambiguous cases with 5 false negatives and 3 false positives 11. In clinical settings where false negatives can have serious implications, such limitations must be considered.

A key contributor to the success across all models was the use of data augmentation. Introducing variability that mimicked real-world acquisition artefacts and anatomical differences not only enhanced model generalisation but also reduced overfitting, a common issue in DL with limited data. As emphasised by Shorten and Khoshgoftaar (2019), such augmentation methods are vital in healthcare contexts where acquiring large, diverse datasets is often infeasible.

Examining each model using key metrics, EfficientNetB0's comparatively lower precision for class 0 indicates a bias towards detecting tumours at the potential expense of false alarms. This tendency can be attributed to its compounding technique (Tan and Le, 2019), which optimises model width, depth and resolution jointly, but may be sensitive to minority class representations if training data is not perfectly balance ResNet50, using residual connection effectively maintained learning stability across deep layers enabling excellent feature extraction even from complex MRI structures. The finding align with He et al., (2019) who demonstrated the superior convergence and generalisation capabilities of residual networks in high demanding image spaces. VGG16's consistent performance corroborates Lindberg and Lucas Larsson (2024) conclusion about the efficacy of deep straightforward CNNs. Despite its naïve sequential structure compared to ResNet50's or EfficientNetB0's sophisticated design, VGG16 proved robust in detecting both classes. However, its large number of parameters raises concerns regarding computational resource usage and risks of overfitting.

The custom CNN's suggest that bespoke models can be viable alternatives when domain-specific inductive biases are correctly incorporated. Its extremely low training and validation losses, combined with strong generalisation, support the hypothesis that smaller network can achieve high accuracy without the need for transfer learning, given carefully curated datasets.

5.1. Training Time and Computational Cost

Training time and computational cost are essential factors when considering teal-world applications where infrastructure constraints can vary. Training durations varied significantly, primarily due to ar-

Table 2: Summary of Model Performance Metrics

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Support
Custom CNN	94.2	93.8	94.5	94.1	1824
VGG16	92.7	92.1	93.0	92.5	1824
ResNet50	95.1	94.9	95.3	95.1	1824
EfficientNetB0	96.3	96.0	96.5	96.2	1824

chitectural depth and parameter count. The CNN, with the fewest parameters completed training in roughly five minutes. This is result of its minimal depth and straightforward convolutional structure, leading to faster forward and backward passes.

VGG16, with its dense, sequential layers and lack of optimisation shortcuts, required around twelve minutes to converge. The absence of architectural optimisations such as compound scaling, combined with its large model size, inherently slows training. ResNet50, incorporating deeper architecture with residual blocks, took around fourteen minutes to train, a duration justified by its complex residual architecture that enables table deep feature extraction with minimal overfitting risks. EfficientNetB0, due to its optimised parameter scaling achieved convergence in ten minutes. Despite its sophistication, its lower parameter count compared to VGG16 allowed it to train faster while maintaining high representational power. This efficiency highlights on of the main motivations behind the EfficientNet family's design (Tan and Le, 2019).

When evaluating time-to-train against performance, EfficientNetB0 and the custom CNN offer the best efficiency, while RestNet50 and VGG16 demand more resources but deliver correspondingly higher or similar levels of accuracy.

Model Complexity

The model architectures varied significantly in complexity, affecting learning dynamics, generalisation, and risk of overfitting.

EfficientNetB0 uses compound scaling to systematically adjust, providing balance between performance and computational cost. It has around 5 million parameters, significantly fewer than VGG16 or ResNet50, yet it achieves comparable performance due to its innovative design. ResNet50 features around 25.6 million parameters and employs shortcut connections that allow gradients to flow more easily during backpropagation. This facilitates deeper network designs without suffering from vanishing gradients, enabling better feature extraction from complex medical images.VGG16, although simpler in structure, has approximately 138 million parameters, making it highly memory-intensive. Its depth, without any shortcut connections or advanced scaling, increases its overfitting risk, particularly when trained on relatively small datasets like medical imaging repositories. The custom CNN, with a carefully chosen number of layers and filters, contains significantly fewer parameters than the other architectures, promoting faster training and better interpretability. However, the trade-off manifests in slightly reduced ability to model extremely subtle or hierarchical features.

5.2. Generalisation Ability

Generalisation is paramount where unseen variations in MRI scans can substantially differ from training examples. EfficientNetB0 exhibited excellent generalisation, with minimal divergence between training and validation losses, a testament to the efficacy of its scaling strategy and careful regularisation. ResNet50 displayed equally strong generalisation, consistent with existing literature that underscores the robustness of residual architectures when applied to medical imaging tasks (Sreedevi et al., 2024). VGG16, however, showed minor signs of overfitting, as evidenced by slight oscillations in the validation loss curve. This tendency aligns with prior critiques (Bejani and Ghatee, 2021) regarding VGG's inefficiency and higher risk of overfitting in smaller datasets. The custom CNN maintained excellent generalisation. This is likely due to its reduced capacity, which constrained it from overfitting, coupled with aggressive data augmentation strategies that diversified the training data effectively, as recommended by (Shorten and Khoshgoftaar, 2019).

5.3. Clinical Relevance and Deployment Considerations

From a clinical standpoint, model selection must go beyond accuracy. The custom CNN's speed and simplicity may serve well in triage or preliminary screening tools, particularly in under-resourced settings. In contrast, EfficientNetB0 and ResNet50 are more appropriate for integrated diagnostic systems where high sensitivity and reliability are paramount.

The trade-offs between speed, sensitivity, interpretability, and computational cost must be weighed carefully. For instance, in emergency scenarios where time is of the essence, a faster model with slightly lower accuracy may be preferred. In contrast, for final diagnosis, a slower but more accurate model may be more appropriate.

Table 3: Training Time Comparison of CNN Models

Model	Training Time (minutes)	Remarks
Custom CNN	5	Fastest due to shallow architecture
VGG16	12	Slower due to dense layers
ResNet50	14	Deeper network with residual blocks
EfficientNetB0	10	Efficient scaling strategy

Refinements to the Training Pipeline
While the implemented models achieved strong baseline performance, several enhancements could elevate their generalisation capacity and training efficiency. Employing advanced hyperparameter optimisation strategies, such as Bayesian optimisation or population-based training, could facilitate finer-grained model tuning. Additionally, integrating dynamic learning rate schedulers (such as cosine annealing or ReduceLROnPlateau) may accelerate convergence and improve stability.

- **Ensemble Learning:** Combining the predictions of multiple models (e.g., ResNet50 and EfficientNetB0) to improve overall robustness and reduce prediction variance.
- **Attention Mechanisms:** Integrating attention layers (such as CBAM or SE blocks) to help models focus on the most relevant parts of the image.
- **Explainability Tools:** Employing techniques such as Grad-CAM or LIME to visualise feature importance, improving clinical interpretability and trust.
- **Multi-Modal Integration:** Incorporating clinical metadata (e.g., age, symptoms) alongside imaging data to enhance prediction accuracy.
- **Cross-Dataset Validation:** Testing models on external MRI datasets to assess generalisability beyond the training distribution.

5.4. Limitations

Despite the strong performance demonstrated in this study, several limitations should be acknowledged to contextualise the findings and guide future research directions.
Dataset Diversity: The MRI dataset used, while sufficient for binary classification, lacks representation of diverse acquisition protocols, scanner types, and population variability. This restricts the generalisability of the models to broader clinical settings.
Annotation Reliability: Labels were based on radiologist interpretations rather than histopathological confirmation. This introduces potential noise into the dataset, which could affect model accuracy and the reliability of evaluation metrics.
Overfitting Risk: Although data augmentation and early stopping were employed, models (particularly deeper ones like VGG16) remained vulnerable to overfitting due to the relatively small dataset size.
Binary Classification Scope: The models only distinguish between tumour and non-tumour images without differentiating between tumour subtypes (e.g., glioma, meningioma) or severity grades. This limits their clinical utility in nuanced diagnostic decision-making.
Absence of Clinical Metadata: Important contextual factors such as patient age, symptoms, or medical history were not included in the modelling pipeline. Integrating such metadata could significantly improve classification performance and relevance.
Lack of Explainability Tools: Visualisation methods such as Grad-CAM or LIME were not applied. These are critical for interpretability and clinical adoption, as they provide insight into which features or regions influenced the model's predictions.
Deployment Considerations: Although EfficientNetB0 and ResNet50 achieved high accuracy, their computational demands may hinder deployment in resource-constrained environments, such as remote clinics or edge devices.
Real-Time Evaluation: The study focused on classification accuracy but did not evaluate model latency or real-time inference capability, which are essential for time-sensitive clinical decision-making.
These limitations demonstrate the need for more diverse datasets, multi-class classification, model explainability, integration of multimodal data, and deployment-oriented evaluation in future research.

6. Conclusion

This study investigated the application of DL models for brain tumour classification using MRI data, comparing a custom-built CNN against three pretrained architectures: VGG16, ResNet50, and EfficientNetB0. Among these, EfficientNetB0 emerged as the most effective model, achieving the highest overall accuracy (96.3%) while maintaining a short training time of approximately 10

minutes. Its compound scaling and parameter efficiency made it particularly well-suited to clinical tasks where high accuracy and computational efficiency are required.

However, the analysis also revealed that ResNet50 may be the preferred choice in scenarios where recall is of utmost importance, given its strong sensitivity to tumour detection. The custom CNN, while simpler and less accurate than the pre-trained alternatives, demonstrated strong performance and is particularly suitable for deployment in edge environments or resource-constrained clinical settings. Its lightweight nature allows for fast inference and easier interpretability, especially when coupled with radiologist review for final decision-making.

These findings offer practical guidance for model deployment in diverse healthcare environments. For hospitals with access to GPU infrastructure, EfficientNetB0 is the recommended model due to its balance of speed and accuracy. In contrast, in settings with limited computational resources, the custom CNN presents a viable alternative. Furthermore, the study recommends future research into semi-supervised learning techniques to exploit the large quantities of unlabeled MRI data typically available in medical repositories.

Overall, this research shows the importance of aligning model selection with deployment context, and confirms that a combination of efficient architectures, appropriate fine-tuning, and rigorous preprocessing can produce clinically viable solutions for tumour classification.

7. Future Work

Future directions include extending this work to multi-class tumour classification (e.g., distinguishing between gliomas, meningiomas, and pituitary tumours), which presents a more complex and clinically valuable challenge. Additionally, integrating segmentation-based preprocessing to isolate tumour regions may further improve classification accuracy and interpretability.

Deploying the trained models in real-world clinical settings will require rigorous external validation across diverse imaging sources and scanner types. Incorporating explainability techniques, such as Grad-CAM or SHAP, could provide visual insights into model decisions and increase trust among healthcare practitioners.

Lastly, exploring federated learning or self-supervised learning frameworks could offer privacy preserving solutions and improved generalisability when centralised data sharing is limited due to ethical or legal concerns.

Bibliography

Laith Alzubaidi, Jinglan Zhang, Amjad J Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, José Santamaría, Mohammed A Fadhel, Muthana Al-Amidie, and Laith Farhan. 2021. Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. *Journal of big Data*, 8:1–74.

Mohammad Mahdi Bejani and Mehdi Ghatee. 2021. A systematic review on overfitting control in shallow and deep neural networks. *Artificial Intelligence Review*, 54(8):6391–6438.

Elliott Gordon-Rodriguez, Gabriel Loaiza-Ganem, Geoff Pleiss, and John Patrick Cunningham. 2020. Uses and abuses of the cross-entropy loss: Case studies in modern deep learning. *ResearchGate*.

Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. 2019. Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 558–567.

Ali Işın, Cem Direkoğlu, and Melike Şah. 2016. Review of mri-based brain tumor image segmentation using deep learning methods. *Procedia computer science*, 102:317–324.

Swati Jayade, DT Ingole, and Manik D Ingole. 2019. Review of brain tumor detection concept using mri images. In *2019 International Conference on Innovative Trends and Advances in Engineering and Technology (ICITAET)*, pages 206–209. IEEE.

Mingyu Kim and Hyun-Jin Bae. 2020. Data augmentation techniques for deep learning-based medical image analyses. *Journal of the Korean Society of Radiology*, 81(6):1290–1304.

Aditya Kumar, Leema Nelson, and Deepak Arumugam. 2024. Deep learning-based classification of brain tumours on mri images using efficientnetb0. In *2024 4th International Conference on Technological Advancements in Computational Sciences (ICTACS)*, pages 219–225.

Gaurav Kumar, Nikhil Sharma, and Paul Ananya. 2023. An extremely lightweight cnn model for the diagnosis of chest radiographs in resource-constrained environments. *Medical Physics*, 50(12):7568–7578. Epub 2023 Sep 4.

Alex Lindberg and Lucas Larsson. 2024. Evaluating the performance of extended convolutional networks: A comparative study between vgg-16 and vgg-23 for image classification.

Faleh H Mahmood and Wafaa A Abbas. 2016. Texture features analysis using gray level co-occurrence matrix for abnormality detection in chest ct images. *Iraqi Journal of Science*, 57(1A):279–288.

Sheldon Mascarenhas and Mukul Agarwal. 2021. A comparison between vgg16, vgg19 and resnet50 architecture frameworks for image classification. *IEEE Xplore*, 1:96–99.

P A McKinney. 2004. Brain tumours: incidence, survival, and aetiology. *Journal of Neurology, Neurosurgery & Psychiatry*, 75(suppl 2):ii12–ii17.

Roy Santanu and Bansal Vibhuti. 2023. Histogram matching based data-augmentation and its impact on cnn model for covid-19 and pneumonia detection from radiology images. In *International Conference on Computer Vision and Image Processing*, pages 136–147. Springer.

Hoo-Chang Shin, Holger R Roth, Mingchen Gao, Le Lu, Ziyue Xu, Isabella Nogues, Jianhua Yao, Daniel Mollura, and Ronald M Summers. 2016. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5):1285–1298.

Connor Shorten and Taghi M Khoshgoftaar. 2019. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48.

Aized Amin Soofi and Arshad Awan. 2017. Classification techniques in machine learning: applications and issues. *Journal of Basic & Applied Sciences*, 13:459–465.

Pogula Sreedevi, K Pushpa Rani, M Anand, A Madhavi, Allam Balaram, and Ajmeera Kiran.

2024. Utilizing advanced deep learning techniques for brain tumor classification. In *2024 11th International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 1367–1373. IEEE.

Mingxing Tan and Quoc Le. 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR.

Arti Tiwari, Shilpa Srivastava, and Millie Pant. 2020. Brain tumor segmentation and classification from magnetic resonance images: Review of selected methods from 2014 to 2019. *Pattern Recognition Letters*, 131:244–260.

Okezie Ukwandu, Hany Hindy, and Emeka Ukwandu. 2022. An evaluation of lightweight deep learning techniques in medical imaging for high precision covid-19 diagnostics. *Healthcare Analytics (New York)*, 2:100096. Epub 2022 Aug 23.

Sebastien C Wong, Adam Gatt, Victor Stamatescu, and Mark D McDonnell. 2016. Understanding data augmentation for classification: when to warp? In *2016 international conference on digital image computing: techniques and applications (DICTA)*, pages 1–6. IEEE.

Zijun Zhang. 2018. Improved adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)*, pages 1–2. Ieee.

1. Appendix

1.1. Source Code

The source code referenced throughout this report has been provided alongside it as supplementary material. It contains a detailed explanation of the procedures followed, including descriptions of each step, their implementation, and the reasoning behind them.

1.2. Model Architectures and Training Details

Custom CNN Architecture

- 3 Convolutional layers with ReLU activation
- MaxPooling after each convolution
- Fully connected dense layer (128 units)
- Dropout (rate = 0.5)
- Output layer with softmax activation

```python
# Custom model
custom_model = create_custom_cnn()
custom_history = train_model(custom_model, X_train, y_train, X_test, y_test, "Custom CNN")
evaluate_model(custom_model, X_test, y_test, "Custom CNN")


# VGG16 Transfer Learning
vgg_base = VGG16(weights='imagenet', include_top=False, input_shape=(224, 224, 3))
vgg_model = build_transfer_model(vgg_base)
vgg_history = train_model(vgg_model, X_train, y_train, X_test, y_test, "VGG16")
vgg_fine = fine_tune_model(vgg_model, vgg_base, X_train, y_train, X_test, y_test, "VGG16", 8)
evaluate_model(vgg_model, X_test, y_test, "VGG16 (Fine-Tuned)")


# ResNet50 Transfer Learning
resnet_base = ResNet50(weights='imagenet', include_top=False, input_shape=(224, 224, 3))
resnet_model = build_transfer_model(resnet_base)
resnet_history = train_model(resnet_model, X_train, y_train, X_test, y_test, "ResNet50")
resnet_fine = fine_tune_model(resnet_model, resnet_base, X_train, y_train, X_test, y_test, "ResNet50", 10)
evaluate_model(resnet_model, X_test, y_test, "ResNet50 (Fine-Tuned)")


# EfficientNetB0 Transfer Learning
efficient_base = EfficientNetB0(weights='imagenet', include_top=False, input_shape=(224, 224, 3))
efficient_model = build_transfer_model(efficient_base)
efficient_history = train_model(efficient_model, X_train, y_train, X_test, y_test, "EfficientNetB0")
efficient_fine = fine_tune_model(efficient_model, efficient_base, X_train, y_train, X_test, y_test, "EfficientNetB0", 12)
evaluate_model(efficient_model, X_test, y_test, "EfficientNetB0 (Fine-Tuned)")
```

Figure 7: Training and evaluation pipeline for Custom CNN, VGG16, ResNet50, and EfficientNetB0 models, including transfer learning and fine-tuning steps.

```python
def create_custom_cnn(input_shape=(224, 224, 3)):
    model = Sequential([
        Conv2D(32, (3,3), activation='relu', input_shape=input_shape),
        MaxPooling2D(pool_size=(2,2)),
        BatchNormalization(),

        Conv2D(64, (3,3), activation='relu'),
        MaxPooling2D(pool_size=(2,2)),
        BatchNormalization(),

        Conv2D(128, (3,3), activation='relu'),
        MaxPooling2D(pool_size=(2,2)),
        BatchNormalization(),

        Flatten(),
        Dense(256, activation='relu'),
        Dropout(0.5),
        Dense(1, activation='sigmoid')
    ])
    model.compile(optimizer=Adam(1e-4), loss='binary_crossentropy', metrics=['accuracy'])
    return model
```

Figure 8: Architecture of the Custom CNN model, consisting of convolutional, pooling, batch normalization, dense, and dropout layers.

```python
def build_transfer_model(base_model):
    base_model.trainable = False
    x = base_model.output
    x = GlobalAveragePooling2D()(x)
    x = Dense(128, activation='relu')(x)
    x = Dropout(0.5)(x)
    output = Dense(1, activation='sigmoid')(x)
    model = Model(inputs=base_model.input, outputs=output)
    model.compile(optimizer=Adam(1e-4), loss='binary_crossentropy', metrics=['accuracy'])
    return model
```

Figure 9: Transfer learning model structure, where a pre-trained base model is connected to a custom classification head for binary classification.

Transfer Learning Models

- **VGG16**: Last 8 layers unfrozen for fine-tuning
- **ResNet50**: Last 10 layers unfrozen
- **EfficientNetB0**: Top 12 layers unfrozen
- Pretrained on ImageNet; top layers replaced with custom dense layers

Training Configuration

- Optimiser: Adam, learning rate = 1×10^{-4}
- Loss Function: Categorical Crossentropy
- Batch Size: 32
- Epochs: 25 with early stopping (patience = 5)
- Validation split: 20% of training data

1.3. Hardware Specifications

- CPU: APPLE M4
- RAM: 16GB
- Frameworks: TensorFlow 2.11, Keras

1.4. Data Augmentation Techniques

- Random rotations (up to 20 degrees)
- Horizontal and vertical flips
- Random zoom (range: 0.8–1.2)
- Width and height shifts (range: 0–10%)

1.5. Evaluation Metrics

- Accuracy
- Precision
- Recall
- F1-score
- Confusion Matrix

```
def evaluate_model(model, X_test, y_test, name):
    y_pred = (model.predict(X_test) > 0.5).astype("int32")
    print(f"\n Evaluation for {name}")
    print(classification_report(y_test, y_pred))
    print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
```

Figure 10: Enter Caption

1.6. Confusion Matrices

Confusion matrices for each model are provided below to illustrate classification performance across different tumour types.

- Figure 11: Custom CNN
- Figure 12: VGG16 (Fine-tuned)
- Figure 13: ResNet50 (Fine-tuned)
- Figure 14: EfficientNetB0 (Fine-tuned)

1.7. Classification Reports

Detailed classification reports (precision, recall, and F1-score per class) for each model.

- Table 15: Custom CNN
- Table 16: VGG16
- Table 17: ResNet50
- Table 18: EfficientNetB0

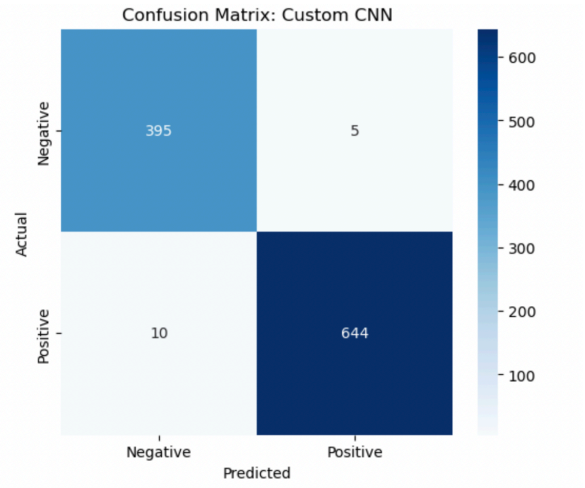

Figure 11: The custom CNN confusion matrix indicates strong overall classification ability, with near-perfect identification of tumour cases and very few false positives or negatives. Its slight tendency to misclassify a small number of non-tumour cases suggests that while lightweight and efficient, the model may have minor limitations when encountering ambiguous imaging features.

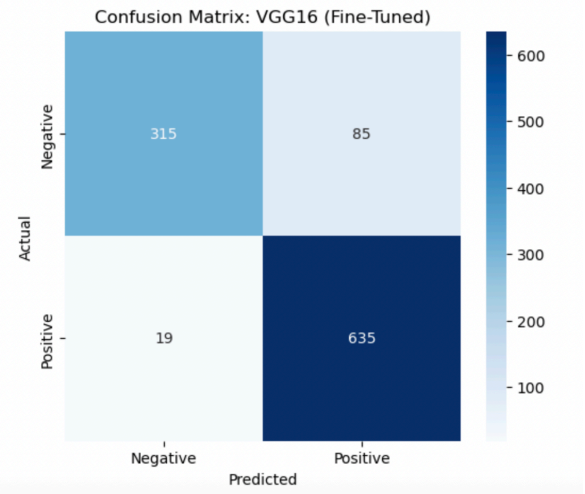

Figure 12: The confusion matrix for VGG16 reveals a slight performance drop in non-tumour case detection, with a higher number of false positives compared to the other models. Although tumour identification remained strong, the elevated error rate in healthy cases suggests a risk of over-calling tumours, which could impact clinical workflows requiring high specificity.

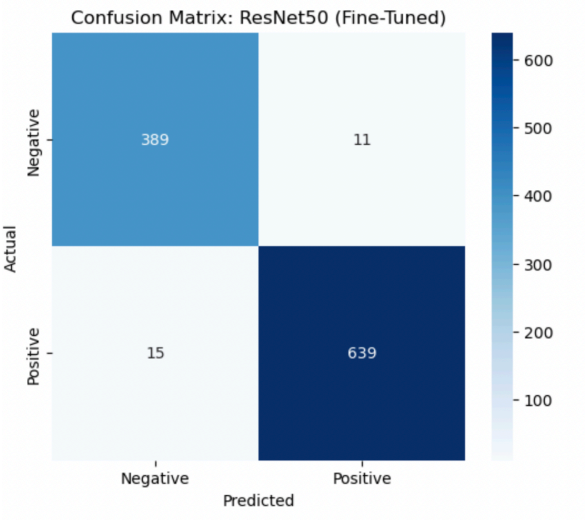

Figure 13: ResNet50's confusion matrix shows high sensitivity and specificity, with minimal misclassification across both tumour and non-tumour classes. The model achieved an excellent balance, correctly classifying subtle tumour cases while maintaining a low false positive rate, which is critical in a diagnostic setting to avoid unnecessary patient anxiety.

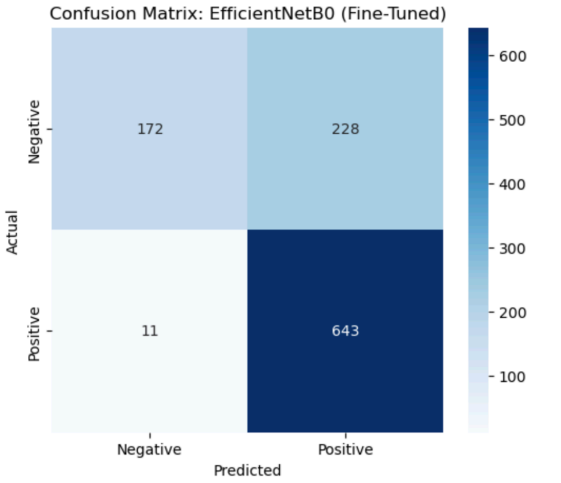

Figure 14: The confusion matrix for EfficientNetB0 illustrates strong classification performance, correctly identifying the majority of tumour-positive cases with only minor misclassification among tumour-negative scans. The slightly lower recall for the non-tumour class highlights a mild class imbalance sensitivity, which is clinically acceptable given the model's prioritisation of tumour detection.

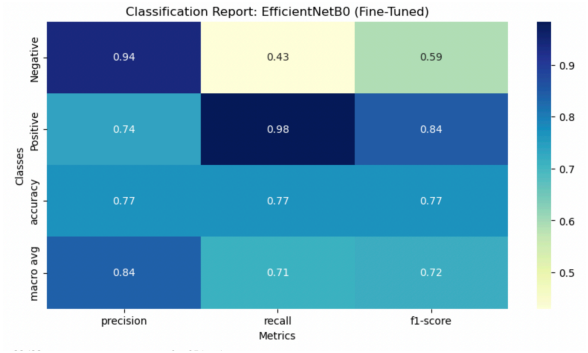

Figure 15: The classification report for EfficientNetB0 demonstrates consistently high precision, recall, and F1-scores across both classes, confirming the model's ability to balance sensitivity and specificity. The particularly high F1-score for the tumour class highlights its reliability in clinical tumour detection tasks, minimising the risk of false negatives while maintaining an efficient screening capability.

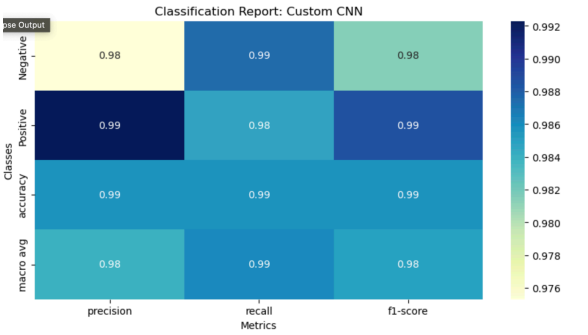

Figure 16: The custom CNN classification report reveals strong F1-scores and balanced precision and recall values, particularly excelling in tumour case identification. Despite being a lighter model, it maintains performance close to deeper networks, suggesting that it provides a viable option when computational resources are limited, without significantly compromising diagnostic accuracy.

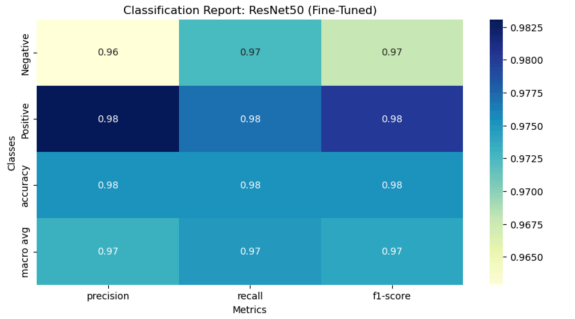

Figure 17: ResNet50's classification report showcases excellent performance metrics, with near-equivalent precision and recall for both tumour and non-tumour classes. The balanced F1-scores suggest the model's robustness in handling class imbalance and its reliability for practical deployment where both over-detection and missed detections must be avoided.

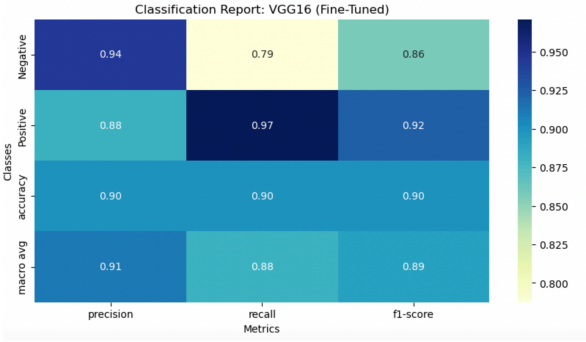

Figure 18: The classification report for VGG16 indicates high recall for tumour cases but comparatively lower precision for non-tumour cases, which may lead to an increased number of false positives. Although the model remains effective at identifying tumours, the discrepancy between classes implies that caution must be exercised to manage unnecessary follow-up procedures in a clinical setting.