Deep Learning for Brain Tumor MRI

Ross Bennfors

Student ID: 710035913

Abstract

Deep learning has revolutionised medical imaging, enabling early diagnosis and treatment planning. This project uses transfer learning and fine-tuning on a VGG19 model for tumor classification on MRI images. A baseline model using VGG19 as a feature extractor achieved 73.8% validation accuracy an incremental fine-tuning, progressively unfreezing layers, improved performance, with test accuracy reaching 89.3%.

Confusion matrices and accuracy/loss values were used to evaluate the models progression through fine-tuning stages. Analysis on these metrics reveal that controlled fine-tuning enhances feature learning but excessive unfreezing risks overfitting. The results highlight the trade-off between model flexibility and generalisation, providing useful insights for optimising deep learning for medical image analysis.

☐ I have used GenAI tools for developing ideas.

☐ I have used GenAI tools to assist with research or gathering information.

✓ I have used GenAI tools to help me understand key theories and concepts.

✓ I have used GenAI tools to identify trends and themes as part of my data analysis.

☐ I have used GenAI tools to suggest a plan or structure for my assessment.

☐ I have used GenAI tools to give me feedback on a draft.

☐ I have used GenAI tools to generate images, figures or diagrams.

✓ I have used GenAI tools to proofread and correct grammar or spelling errors.

☐ I have used GenAI tools to generate citations or references.

✓ Other: *To helps with code comment clarity*

1 Introduction

Brain tumor detection through medical imaging and deep learning has been a great advancement in healthcare in recent years. The ability to accurately detect brain tumors is important for early diagnosis and treatment. Traditional manual analysis of medical images can be time-consuming and subject to human error, making deep learning-based classification a valuable tool. Recent advances in deep learning, particularly convolutional neural networks (CNNs), have shown great success in medical image analysis, significantly improving accuracy when diagnosing many different health conditions[1].

1.1 Aim

This report presents the development and evaluation of a deep learning model for brain tumor classification using transfer learning and fine-tuning of a VGG19-based CNN model. The main goal is to build a model that can classify brain tumors accurately while also ensuring it generalises well to unseen data. This report discusses the fine-tuning strategies used and their impacts on the model, also assessing the trade-off between feature extraction, generalisation and mitigating overfitting.

1.2 The Problem

Brain tumors can vary a lot in shape, size and texture, making classification challenging. This project uses a dataset of medical images separated into tumorous or non-tumorous. Before training there are a few problems that need to be carefully navigated:

- Class size and imbalance: The provided dataset was imbalance and too small to effectively train a model. This required data augmentation to improve class representation
- Overfitting risk: Deep CNNs are prone to overfitting due to there large number of parameters, especially when fine-tuning on a small dataset.
- Generalisation: Making sure the trained model performs well on unseen test data, rather than memorising training data, is important for practical use.

1.2.1 Related Work

Deep learning, especially with CNNs, has shown great success in a variety of medical imaging tasks, such as tumor detection and disease classification[1]. There are a number of CNN architectures but this project utilised VGG19, introduced by Simonyan and Zisserman. VGG19 has been widely used due to its deep architecture, small receptive fields and strong feature extraction capabilities[2].

Developed for large-scale image recognition, VGG19 has feature extraction and transfers well to medical imaging tasks. Training a CNN from scratch on a relatively small dataset would likely lead to overfitting. Pre-trained models such as VGG19 provide a strong feature extractor that can be fine-tuned for this task thanks to having been trained on large datasets, such as ImageNet. Several studies have shown that fine-tuning pre-trained CNNs significantly improves performance with image classification tasks.

1.3 Achievements

This report documents the training and optimisation of a CNN for classification of brain tumors. Key points are:

- Baseline Model: an initial transfer learning model was created using VGG19 to have a baseline performance
- Fine-Tuning: Layers were progressively unfrozen to achieving increased learning and performance.

- Performance Analysis: Models were analysed using confusion matrices, accuracy/loss values, and validation vs test performance data.

The report covers the dataset, preprocessing, CNN architecture, hyper-parameters and training (Section 2), and the evaluation process and results (Section 3), and a summary and final conclusion (Section 4).

2 Methodology

2.1 Data Preprocessing

The dataset provided consisted of brain MRI images separated into tumorous and non-tumorous categories. This initial dataset was imbalanced, with a greater proportion of tumor images compared to non-tumor images (67%/33% split). The dataset was also significantly smaller than what is required to build a meaningful model. To improve the quality and size of the data a number of pre-processing techniques were used:

- Data Augmentation:
  - Rotation ($\pm 10°$): Helps in learning rotational invariance.
  - Horizontal and Vertical Flipping: Helps the model generalise to flipped orientations.
  - Brightness Adjustment: Simulates different lighting conditions.
  - Width and Height Shifts (10%): Helps model to learn from slightly shifted tumor locations.
  - Shear Transformations (10%): Applies small distortions to reduce overfitting.
  - Image Rescaling and Normalization: Images were resized to 240x240 pixels to ensure consistent input size. Pixel values were scaled to the [0,1] range by normalizing with 1/255 to improve convergence during training.
- Image Cropping: Each image was cropped to remove unnecessary background elements, such as empty space, reducing input size to enhance training efficiency and improve feature learning.
- Class balancing: Tumerous and non-tumerous images were augmented 6 and 13 times respectively in order to balance the dataset.

Once data augmentation had been complete, the augmented dataset was split into training, validation, and test sets at 80%, 10% and 10% respectively. The training set was used for model training, the validation set for hyperparameter tuning and fine-tuning, and the test set for model evaluation.

2.2 Model Architecture

A VGG19-based convolutional neural network (CNN) was chosen for classification due to its, previously mentioned, strong feature extraction capabilities.

Transfer Learning: The base VGG19 model was loaded without its fully connected (FC) layers, keeping only the convolutional layers. Initially, all layers were frozen, meaning they retained their pre-trained weights without updating. This approach allowed the model to use pre-learned features from ImageNet to help reduce training time and overfitting risks.

Custom Classifier: A custom classification head was added to the VGG19 base model:

- Flatten Layer: Converts the feature maps into a 1D vector for dense layers.
- Dense Layer (1024 neurons, ReLU activation): Captures high-level features from the extracted representations.
- Dropout Layer (30%): Reduces overfitting by randomly deactivating neurons during training.

- Dense Layer (256 neurons, ReLU activation): Further refines learned features before classification.
- Final Output Layer (Softmax, 2 neurons): Generates tumor vs. non-tumor predictions.

Fine-Tuning: to maximise feature learning, layers were gradually unfrozen:

- Baseline Model: Only FC layers were trained (VGG19 layers frozen).
- Fine-Tuning Stage 1: Last two convolutional layers (Block 5 Conv3 and Conv4) were unfrozen for targeted feature learning.
- Fine-Tuning Stage 2: All of Block 5 was unfrozen, allowing deeper fine-tuning of high-level feature extraction
- Fine-Tuning Stage 3: Blocks 4 and 5 were unfrozen, targeting mid-to-high-level feature learning.
- Final Fine-Tuning: The entire VGG19 model was unfrozen, allowing full retraining for maximum feature adaptability.

After each fine-tuning stage validation loss and accuracy was monitored to make sure additional tuning was improving generalisation rather than overfitting.

2.3 Training and Optimisation

The initial model was trained using the Adam optimiser with a learning rate of 0.0003, and due this being a binary classification problem, categorical cross-entropy loss was used. After, fine-tuning was introduced by progressively unfreezing deeper layers, reducing the learning rate to prevent forgetting.

Hyperparameters: To optimise model performance, the following hyperparameters were used:

- Batch Size = 32: Balanced to ensure efficient training while maintaining GPU memory constraints.
- Epochs = 10-20: Varied per fine-tuning stage to prevent overfitting while allowing sufficient learning.
- Optimiser = Adam: Chosen for its adaptive learning rate capabilities, helping the model converge faster.
- Learning Rate:
  - Baseline Model: 0.0003 for initial training.
  - Fine-Tuning Stages: Progressively reduced from 0.0001 to 0.00001, over 4 fine-tuning stages, to refine feature learning without destabilizing training.
- Loss Function = Categorical Cross-Entropy: Used due labels being stored in one-hot encoding format.
- Evaluation Metrics: Accuracy, Loss and Confusion Matrix.

Callbacks:

- Early Stopping: Stops training if validation loss does not improve for 4 consecutive epochs, preventing overfitting.
- Model Checkpointing: Automatically saves the best-performing model at each fine-tuning stage (models were saved to separate files for analysis across the whole training process).
- Learning Rate Reduction on Plateau: Reduces the learning rate by 50% if validation accuracy stagnates for 3 epochs, allowing better convergence.

3 Results

The model was evaluated after each fine-tuning stage using both the validation and test datasets. Evaluation was done using validation accuracy and loss to assess overfitting, test accuracy and loss was used to measure generalisation performance and confusion matrices were displayed to provide an insight into performance of tumor vs. non-tumor classification.

3.1 Baseline Training

The initial model displayed slow convergence, beginning with an accuracy of 49.3% and gradually improving to 75.0%. Validation accuracy steadily increased to 76.7%, however, a plateau in validation loss could be seen in later epochs. Despite the lack of fine-tuning, the model demonstrated strong generalisation, showing that the VGG19 feature extractor was effective for tumor classification.

3.2 Fine-Tuning Stage 1

In the first step in fine-tuning the last two layers of Block 5 were unfrozen which produced a great improvement, with validation accuracy reaching 81.1%. The similarity between training and validation accuracy suggests that the model was learning increasingly well without overfitting. Additionally, the drop in validation loss (see Figure 2.a) indicates that fine-tuning these deeper layers allowed the model to extract more relevant features, improving its generalisation on unseen data.

3.3 Fine-Tuning Stage 2

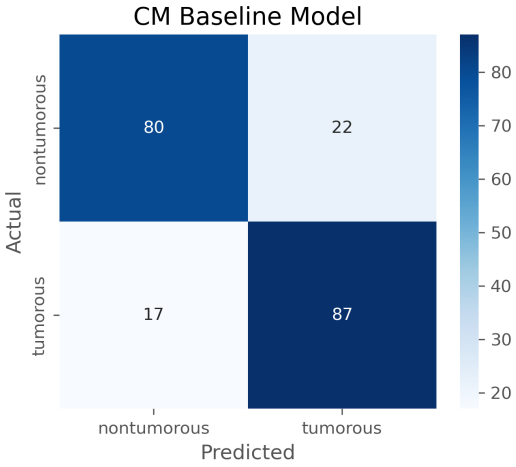
This next fine-tuning stage involved unfreezing the whole of Block 5 of VGG19. This lead to a slight improvement of validation accuracy to 82.0%, with test accuracy reaching 83.5%. This stage, however, did introduce a small class imbalance as shown in Figure 1.c. While non-tumorous accuracy improved a little bit there was a large drop in tumorous accuracy suggesting that the model may have started favouring non-tumorous classifications. Additionally, while training loss decreased, validation loss fluctuated slightly, this indicates that the model might have begun overfitting to the training data, potentially reducing its ability to generalise further.

3.4 Fine-Tuning Stage 3

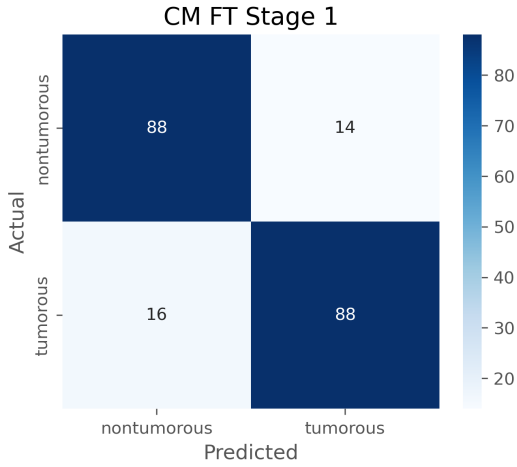
In fine-tuning stage 3, Blocks 4 and 5 of the VGG19 model were unfrozen, allowing deeper feature refinement. This resulted with improvements of validation accuracy reaching 85.4% and test accuracy increasing to 87.3%. Unlike the previous stage, where overfitting became a concern, this stage maintained a stable validation loss while training loss continued to decrease (shown in Figure 2.b), suggesting an more ideal trade-off between model flexibility and generalisation. Additionally, tumorous prediction accuracy recovered (see Figure 1.d), leading to a more balanced performance across both classes.

3.5 Final Fine-Tuning Stage

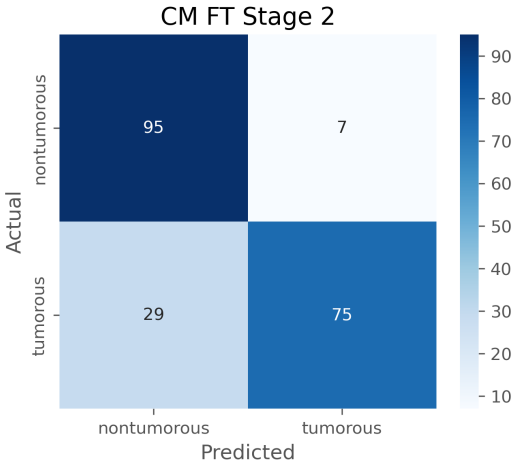
In the final fine-tuning stage, the whole VGG19 model was unfrozen which resulted in the highest model test accuracy of 89.3%, however, validation accuracy plateaued at 85.4%. This indicated that while the model performed very well on test data, its ability to generalise to unseen validation data did not significantly improve. Additionally, non-tumorous prediction accuracy decreased slightly (see Figure 1.e), suggesting a potential trade-off in class-wise performance. Figure 2.d also shows validation loss had no significant improvement, adding to the idea that although this fine-tuning increased test performance it has not improved the models ability to generalise towards unseen data.



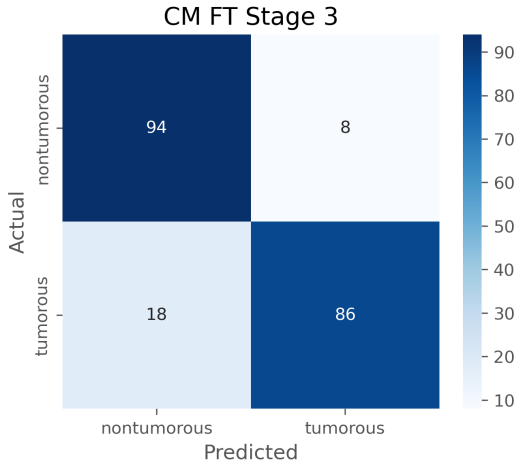
(a) Baseline Model



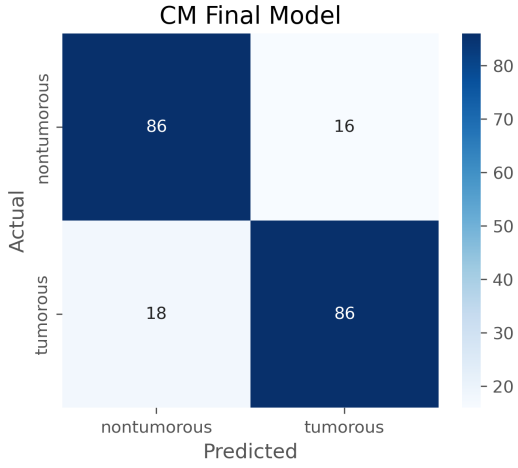
(b) Fine-Tuning Stage 1



(c) Fine-Tuning Stage 2

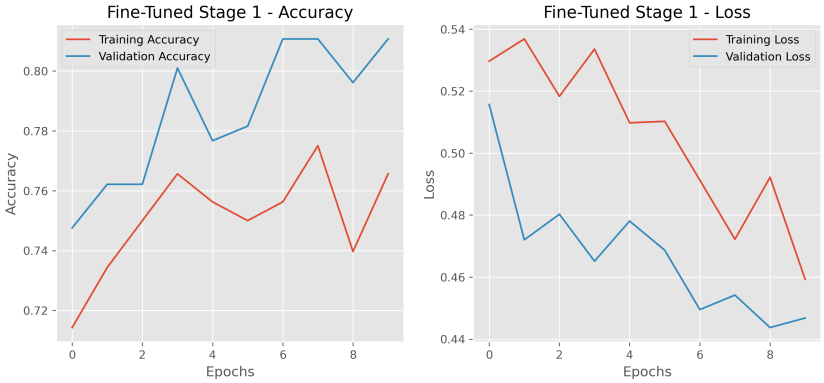


(d) Fine-Tuning Stage 3

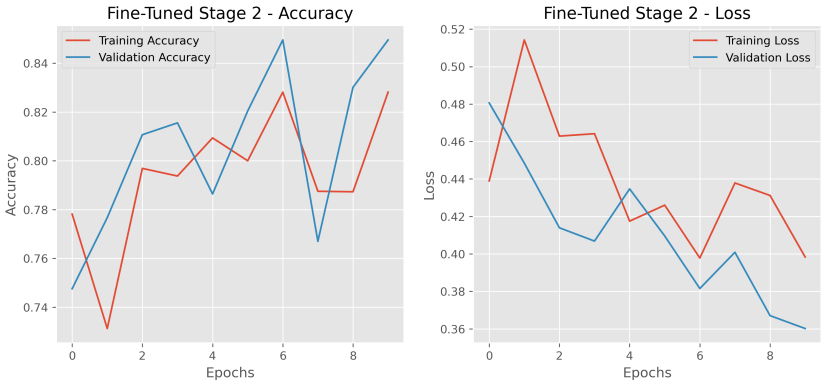


(e) Final Fine-Tuned Model

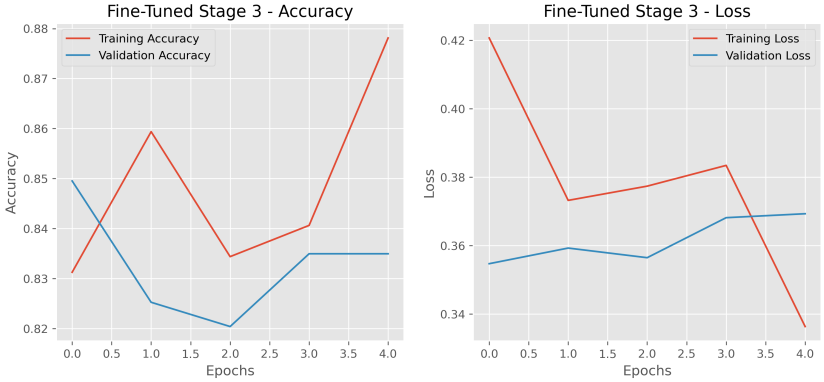
Figure 1: Confusion Matrices for Each Fine-Tuning Stage



(a) Training vs Validation Loss and Accuracy per epoch for Fine-Tuning Stage 1



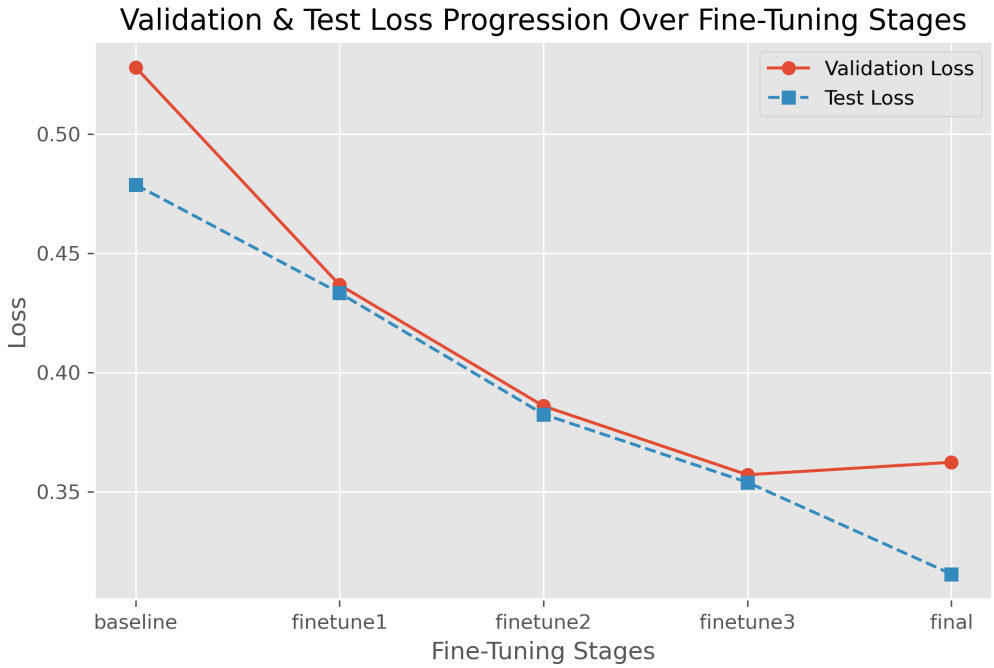
(b) Training vs Validation Loss and Accuracy per epoch for Fine-Tuning Stage 2



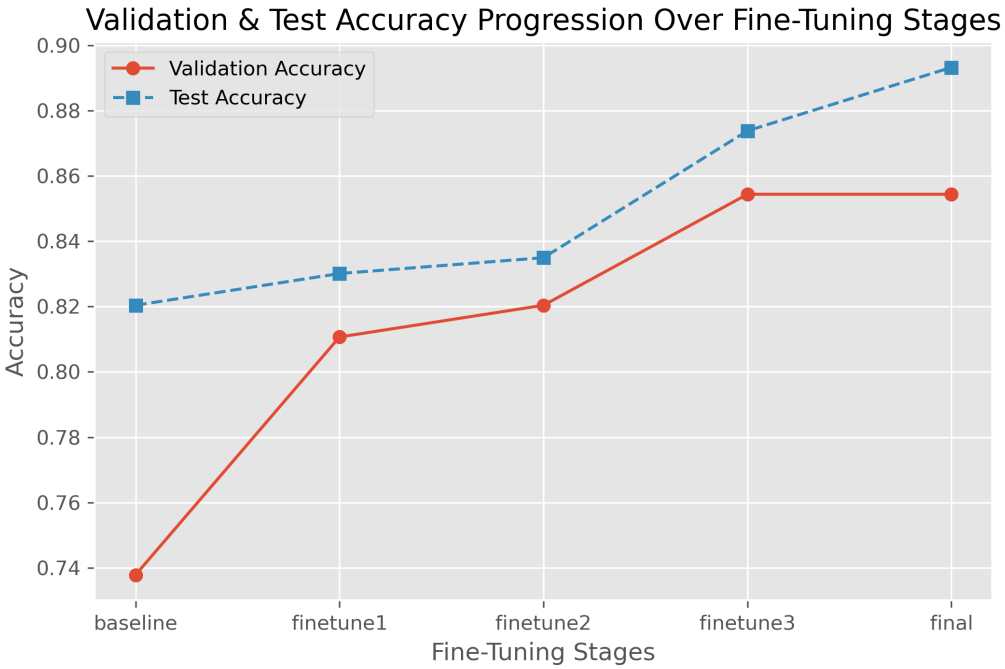
(c) Training vs Validation Loss and Accuracy per epoch for Fine-Tuning Stage 3

(d) Training vs Validation Loss and Accuracy per epoch for Final Fine-Tuning Stage

Figure 2: Training vs Validation Loss and Accuracy for each Fine-Tuning Stage



(a) Validation vs Test Loss Across Fine-Tuning Stages



(b) Validation vs Test Accuracy Across Fine-Tuning Stages

Figure 3: Comparison of Validation and Test Performance Across Fine-Tuning Stages

4 Conclusions

This project explored the use of a VGG19-based deep learning model to classify brain tumors from MRI scans, progressively fine-tuning it to increase performance. The results demonstrate the incremental fine-tuning by unfreezing certain layers at a time significantly improve classification accuracy and ability to generalise to unseen data.

As shown in Figure 3.b, validation and test accuracy improved well across fine-tuning stages, with the most significant performance boost occurring between the baseline model and fine-tuning stage 1. This suggests that unfreezing some layers and retraining deeper layers within Block 5 was greatly beneficial in refining feature extraction for tumor detection. Figure 3.a highlights that validation loss decreased substantially from the baseline to the fine-tuning stages but plateaued after stage 3, indicating lessening returns from further fine-tuning. Additionally, the slight increase in validation loss in the final fine-tuning stage indicates that unfreezing the entire VGG19 model introduced a risk of overfitting, due to the model adapting to the training data.

Overall, fine-tuning VGG19 progressively improved tumor classification accuracy, helping the model reach a test accuracy (89.3%) after unfreezing the whole model. However, validation performance plateaued at the last fine-tuning stage, suggesting that further improvements could be explored.

References

[1] S. K. Zhou, H. Greenspan, C. Davatzikos, J. S. Duncan, B. Van Ginneken, A. Madabhushi, J. L. Prince, D. Rueckert, and R. M. Summers, "A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises," *Proceedings of the IEEE*, vol. 109, no. 5, pp. 820–838, 2021.

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.