UNIVERSITY PARTNER

Artificial Intelligence and Machine Learning (6CS012)

Deep Learning for Image Classification: From Scratch to Transfer Learning

Student Id	: 2331498
Student Name	: Mohammed Samrose
Group	: L6CG15
Module Leader	: Mr. Siman Giri
T/W	: Mr. Aatiz Ghimire
Submitted on	: 5/19/2025

Table of Contents

Abstract

The objective of this project is to apply CNNs to identify and separate brain MRI images between glioma, meningioma, pituitary and no tumor cases. For the study, two CNN models are developed, one basic and one deeper with regularization, along with implementing MobileNetV2 through transfer learning. All the images in this dataset which come from Kaggle, have been sorted into training and testing groups. Images were first resized to 224x224 pixels, normalized and then data was augmented. The top result of 95% validation accuracy was obtained by CNN, with MobileNetV2 (frozen) getting 91.5%. The performance of Fine-tuned MobileNetV2 decreased to 86.5% because it fitted the models too much to the small available data. Even though CNN surpassed transfer learning in how accurate it was, it took longer to train. When using frozen weights, transfer learning gave good performance and was a smart choice for situations where resources were limited. Workers are currently investigating how larger data sets and better regularization methods influence the outcome of fine-tuning.

1. Introduction

Classifying medical images into categories helps automate the process and improves the efficiency of healthcare providers. Image classification is mainly achieved with CNNs due to their ability to find patterns in layered datasets. The project includes sorting brain MRI images to recognize the type of tumor, something difficult in medicine as the look of tumors changes and detecting them accurately is necessary.

Its purpose is to test and compare solitary training for CNNs from scratch and utilizing a pre-existing MobileNetV2 for transfer learning. The task relies on deep learning, covering areas such as preprocessed data, designing a neural network and using evaluation metrics from Weeks 1 to 6 in the course. It has been proved with literature that CNNs are useful in medical imaging and that transfer learning works in datasets where data is limited. It includes building a basic CNN, improving it by adding regularization and modifying MobileNetV2 so that it is best suited for the required task.

2 Dataset

In this project, we used the Brain Tumor MRI Dataset provided by Kaggle (https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset). 4 classes are included in the dataset: glioma, meningioma, pituitary and no tumor. There are 3,264 images used for training and 394 images for testing. The images consist of grayscale or RGB scans that are often about 512x512 pixels in size.

2.1 Preprocessing

The images were resized to 224x224 pixels to match the MobileNetV2 input size and also to encourage computational thriftiness. Normalization was employed to transform the pixel values to the range [0, 1] for better convergence of the model. Data augmentation techniques like random rotations (up to 20 degrees), flipping about the horizontal axis, and zooming by up to 20% were employed to encourage variability within the dataset and discourage overfitting. The Keras ImageDataGenerator was employed to carry out these procedures.

2.2 Data Split

We divided the dataset into 80% training (2,611 images) and 20% validation (653 images) sets, which left the test set (394 images) for final testing. This provides a good balance between model training and validation with enough data for testing performance.

2.3 Challenges

Class balance was challenging to achieve since the no-tumor class contained a few more images than the rest. Augmentation assisted to a certain extent by creating diversified samples. Resizing high-resolution MRI images can also lead to loss of fine details, and that was avoided by careful choice of interpolation techniques (bilinear).

3 Methodology

This section outlines the model architecture, training strategies, and evaluation metrics used in the project.

3.1 Baseline CNN

The vanilla CNN had three convolutional layers (32, 64, and 128 filters, 3x3 kernel, ReLU activation), each followed by max-pooling (2x2). Three fully connected layers (512, 256, and 128 units, ReLU) were followed by a softmax output layer for classification into four classes. Categorical cross-entropy loss and SGD optimizer (learning rate 0.01) were used for training the model.

3.2 Deeper CNN

The deeper CNN included six more convolutional layers (32, 64, 128, 256, 512, 512 filters) with batch normalization after each convolutional layer and dropout (0.5) after dense layers. Dense layers were L2 regularized (0.01) to avert overfitting. Adam optimizer (learning rate 0.001) was also utilized, as was categorical cross-entropy loss. Three dense layers (1024, 512, 256 units) came before the softmax output.

3.3 Transfer Learning with MobileNetV2

MobileNetV2 pre-trained on ImageNet was modified by removing its top fully connected layers and introducing new dense layers (512, 256 units, ReLU) and a softmax layer for four classes. Two strategies were used:

- Feature Extraction: The convolutional base was frozen, and training was only done on the new dense layers (learning rate 0.001, Adam).
- Fine-Tuning: Top 20 convolutional base layers were unfrozen, and the entire model was trained with a lower learning rate (0.0001, Adam) to prevent 3 catastrophic forgetting.

Images were resized to 224x224 pixels and preprocessed with the preprocessing function of MobileNetV2.

3.4 Hyperparameters

Each model was trained for 50 epochs with a batch size of 32. Early stopping (with patience of 10 epochs) was employed to check validation loss and prevent overfitting. Training was on Google Colab with GPU acceleration.

4 Experiments and Results

This section presents the experiments conducted and their outcomes, comparing the baseline CNN, deeper CNN, and MobileNetV2 models.

4.1 Baseline vs. Deeper Architecture

The baseline CNN attained around 82% validation accuracy and macro F1-score of 0.82. The no-tumor class achieved the best F1-score (0.95), while glioma was worst (0.70). The deeper CNN greatly increased performance to 95% validation accuracy and macro F1-score of 0.95. The no-tumor class was still best (0.98), and meningioma worst (0.90). Loss plots indicated the deeper CNN to converge more quickly and to lower validation loss (0.25 compared to 0.42).

4.2 Computational Efficiency

The first CNN took around 20 minutes to train, whereas the deeper CNN took 35 minutes to train because of its higher complexity. MobileNetV2 (frozen) was the quickest and took 15 minutes to train since only the dense layers were trained. Fine-tuned MobileNetV2 took 25 minutes to train because of partial unfreezing. All of the models used Google Colab's GPU acceleration.

4.3 Training with Different Optimizers

The deeper CNN was trained using both SGD and Adam as well. Adam converged quicker (in 20 epochs) and with better accuracy (95%) than SGD (90% at 30 epochs). The adaptive learning rate of Adam was more appropriate for the complicated architecture

4.4 Challenges in Training

The baseline CNN had overfitting issues. This was shown in diverging training and validation loss curves. Deeper CNN addressed this through regularization (batch normalization, dropout, L2). Fine-tuned MobileNetV2 had some overfitting issues, possibly due to the fact that the dataset size was relatively small, making unfreezing convolutional layers less effective.

5 Fine-Tuning or Transfer Learning

MobileNetV2 was used for transfer learning because of its efficiency and performance on ImageNet. When used for feature extraction, the frozen convolutional base achieved a validation accuracy of 91.5% and macro F1-score of 0.91 (no-tumor: 0.98, meningioma: 0.83). Fine-tuning with top 20 layers unfrozen resulted in a lower validation accuracy of 86.5% and macro F1-score of 0.86 (no-tumor: 0.94, meningioma: 0.76). Surprisingly, the frozen MobileNetV2 performed better than the fine-tuned MobileNetV2, probably because the small training set (3,264 images) was not sufficient to retrain convolutional layers without overfitting. MobileNetV2 (frozen) was less accurate than the deeper CNN (91.5% vs. 95%) but faster to train and with fewer tunings than the custom CNNs. The two MobileNetV2 variants performed better than the baseline CNN.

6 Conclusion and Future Work

This project demonstrated the effectiveness of CNNs and transfer learning for the classification of brain MRI images. Scratch-trained deep CNN with Adam achieved the highest 5 validation accuracy (95%) and macro F1-score (0.95), demonstrating the potential of well-designed architectures with appropriate regularization. Frozen MobileNetV2 was also a close competitor by achieving 91.5% accuracy with significantly less training time and is thus most apt for resource-limited setups. Fine-tuned MobileNetV2 did worse due to overfitting, showing the importance of dataset size in transfer learning.

- Larger and more diverse datasets to improve fine-tuning outcomes.
- Advanced regularization techniques, such as CutMix or MixUp, to enhance generalization.
- Ensemble methods combining custom CNNs and pre-trained models for improved accuracy.
- Higher-resolution inputs with architectures like EfficientNet to preserve fine details.

These improvements could further enhance model performance and applicability in realworld medical diagnostics.