BRAIN TUMOUR CLASSIFICATION USING CNNS

Marco Mastrangelo
Student# 1009133518
marco.mastrangelo@mail.utoronto.ca

Hussein Ismail
Student# 1008747910
h.ismail@mail.utoronto.ca

Devraj Solanki
Student# 1009065707
devraj.solanki@mail.utoronto.ca

Damilola Aina
Student# 1008932292
dami.aina@mail.utoronto.ca

Total Pages: 8

1 INTRODUCTION

Brain tumours are a potentially life-threatening medical condition. The 5-year relative survival rates for certain malignant brain tumours can be as low as 6% (Ostrom et al., 2019). Therefore, the accurate and early detection of brain tumours is crucial for improving patient outcomes. Currently, medical professionals use magnetic resonance imaging (MRI) scans to visually identify brain tumours. However, the manual identification of brain tumours can be challenging due to the large number of images produced from one MRI scan (Ullah et al., 2023). Additionally, the large variety of brain tumour sizes, shapes, and locations makes it even so more challenging to detect and classify different types of brain tumours (Ullah et al., 2023). Applying deep learning models, specifically convolutional neural networks (CNNs) (which exhibit high proficiency in image classification tasks), can result in faster processing of MRI images and more accurate detection of brain tumours by eliminating human error. This project aims to produce a convolutional neural network model that can accurately detect and classify 3 different types of brain tumours: glioma, meningioma, and pituitary.

2 ILLUSTRATION OF THE MODEL

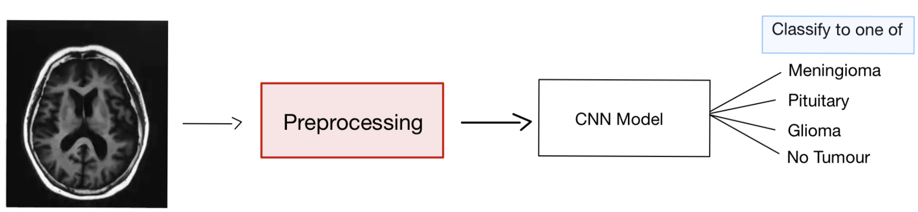

Figure 1: Basic Illustration of Deep Learning Model. Image: Hussein Ismail

The goal of this project is to input an image of an MRI brain scan, preprocess it to improve accuracy and robustness, and then classify what type of tumour is displayed in the MRI, if one exists.

3 Background and Related Work

Using deep learning techniques to identify abnormalities in MRI scan images is an ongoing area of research in academia. Classification and/or segmentation of brain tumours in MRI images appears to be a well-documented topic within this area. A widely known CNN architecture that is used for medical imaging is the *U-Net* architecture, proposed in 2015 by researchers at the University of Freiburg (Ronneberger et al., 2015). This architecture is extremely influential as it is designed to perform well on a small amount of data. U-Net learns the important feature(s) of an image and creates a segmentation mask, highlighting key area(s) of the image where the feature(s) are present. This architecture is widely used in medical imaging segmentation research, with others creating different versions of U-Net. For example, Futrega et al. (2021) proposes an optimized U-Net specifically for brain tumour segmentation tasks.

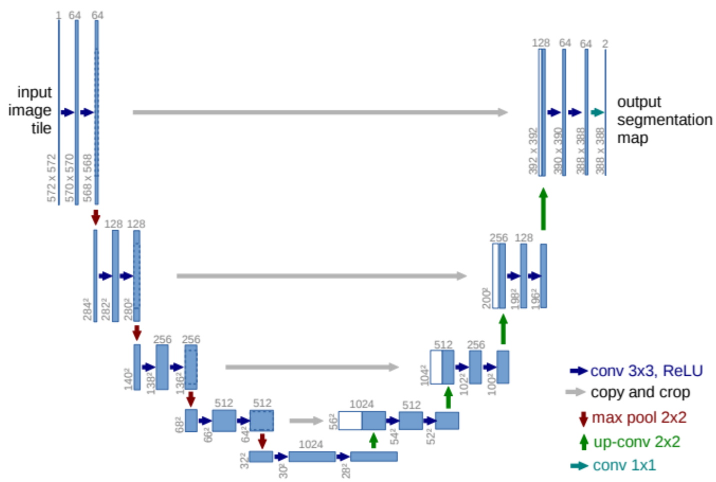

Figure 2: General U-Net Architecture. Image: Olaf Ronneberger

Researchers today continue to make strides to improve the accuracy of these CNN models, with some architectures boasting test accuracies as high as 99.27% (Ullah et al., 2023). This specific network – called *tumourDetNet*, uses 48 convolutional layers with ReLU and Leaky ReLU activation functions. Recently, researchers have been making use of the concept of transfer learning in their CNN models, as shown in Kumar et al. (2023). This CNN model leverages this idea by using a pre-trained model called *Res-Net 50* to classify tumours as either benign or malignant, and achieves a test accuracy rate of 99.3% and 98.4% respectively for these two types. Raza et al. (2022) proposes a hybrid model, using all but the last 5 layers of *GoogLeNet* and adding on an extra 15 layers, to achieve a test accuracy of 99.67%. Even with these impressive advancements in the past years, these CNN models still face challenges in attempts to become applied within the healthcare sector for several reasons (Xie et al., 2022). For example, there exists a limited amount of publicly available MRI scans to train these networks (due to patient privacy and security concerns). Furthermore, it is challenging for clinicians and radiologists to explain why a CNN outputs the results it does. For example, A radiologist cannot simply say that a patient has a brain tumour because a CNN says they do. These models can still make errors from time to time and still require human supervision. Xie et al. (2022) explains the current state of using CNNs to detect brain tumours and the challenges concerning putting these models into practice.

4 Data Processing

The group conducted extensive research on datasets containing MRI images of brains with tumours, and chose to use the following two datasets from Kaggle – Nickparvar (2021), and Sartaj et al. (2020). Both datasets have brain tumour MRI images, organized into four folders – meningioma, glioma, pituitary, and no tumour. These datasets give the group a solid foundation, providing a healthy number of images to use for each class.

4.1 Dataset Description

As mentioned in Section 4, both datasets contain MRI images that follow four class groups: meningioma, glioma, pituitary, and no tumour. After merging these two datasets, the group ends up with 2582 total photos containing meningioma tumours, 2547 images containing glioma tumours, 2658 photos containing pituitary tumours, and 2500 images containing no tumours.

4.2 Proposed train, validation and test split

The team's proposed dataset contains 10287 images in total from these four image classes. The distribution across the four classes is the following: 24.3% of the images have no tumours, 25.8% have pituitary tumours, 25.1% have meningioma tumours and 24.8% have glioma tumours. The team plans to split the data into a 60-20-20 distribution, with about 60% of the images going towards training the model (approximately 1500 images for each class), 20% going towards the validation data (approximately 500 images for each class), and the final 20% going towards the testing data. With this current class distribution, the model can be trained and tested without having to account for class imbalances.

4.3 Cleaning Data

With the images successfully organized, the next step would be to ensure that the format of all input images are uniform. Black pixels will be added to images with non-symmetrical dimensions to make them symmetrical. The `torch.transforms.Resize(...)` function will be used to resize all images to 400x400 pixels (Pytorch Official Documentation). Additionally, images will be visually inspected to verify their usability; meaning blurry or unidentifiable images will not be used when training or testing the team's model.

4.4 Augmentation of Data

When training the model, the team will enhance images through augmentation to introduce variation, improving the robustness of the team's model (Amazon, 2024). These augmentations would be applied to images to varying extents; some images may remain unaffected, while others might incorporate one, a few, or even all augmentations. Specifically, variations will be introduced to images by modifying the following features:

Color Intensity: Adding a small constant value to each RGB value of each pixel of the image, a faded image could be mimicked.

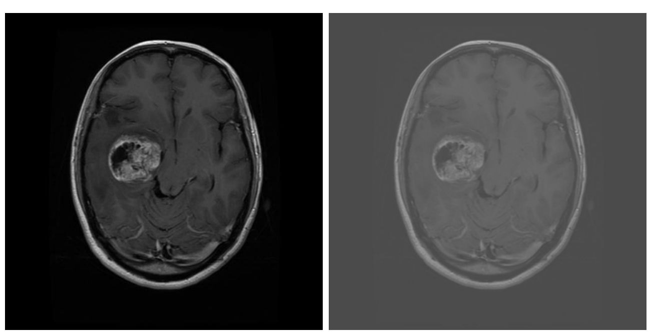

Figure 3: Before and after of MRI when adding a constant RGB value to cause a 'faded' effect. Image: (Original) Nickparvar (2021)

Image Flipping: Using the `torch.transforms.functional.hflip(...)` and the `torch.transforms.functional.vflip(...)` functions within the PyTorch library (Pytorch Official Documentation), the image can be horizontally and/or vertically flipped to alter the orientation of the scan.

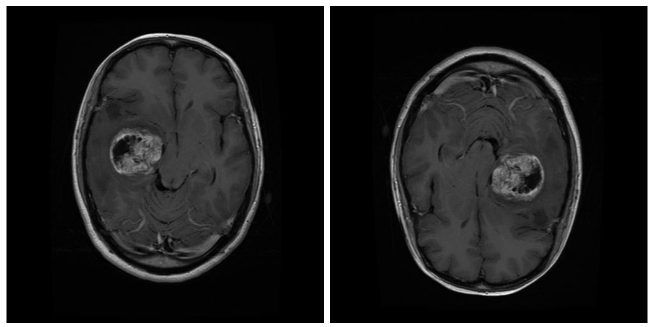

Figure 4: Before and after of MRI after being vertically and horizontally flipped.
Image: (Original) Nickparvar (2021)

By altering some of the data in such a way, the team creates a dataset with more variety that would prepare the model for a wider range of real-world scenarios, thereby improving model robustness.

5 Architecture

For the specified problem, using a convolutional neural network (CNN) architecture is most suitable. CNNs reduce the number of parameters without compromising the quality of the model. This is beneficial for classifying images as the input size is so large that each pixel is considered a tunable parameter. CNNs accomplish this by applying a kernel in a series of strides to the original input to group features together, this is done repeatedly until the model is left with a small number of grouped abstract parameters to evaluate (Mishra, 2019). The model is then trained to detect and classify the input based on these grouped parameters, ultimately reducing computation intensity. Below is a diagram of the general architecture of a CNN.

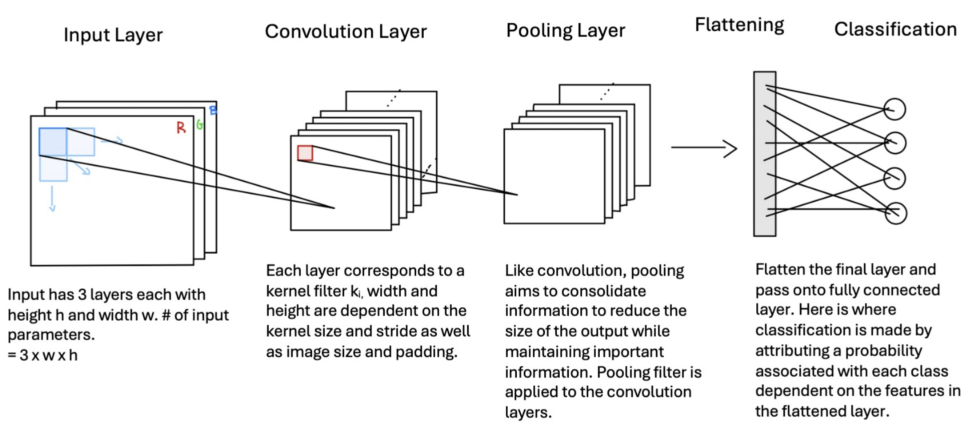

Figure 5: General CNN Architecture. Image: Hussein Ismail

There can be any number of convolution and pooling layers, this will be an architectural choice that the group will make after experimentation and further research on similar models. However, the general idea is that the model consolidates the input image into abstract features, eliminating unnecessary information to reduce computational complexity once it reaches the fully connected layer, leading to easier classification.

6 Baseline Model

For this project, the team will compare their neural network against a simple, existing convolutional neural network used to detect pneumonia from x-ray imagery(Foster, 2019). The network consists of 2 convolutional layers, 2 max-pooling layers, 1 flatten layer, 2 fully connected layers and 1 dropout layer. The original output layer consists of two neurons that classifies images as pneumonia or not pneumonia. To be able to compare the baseline model with the team's model; the output layer of the baseline model will be augmented to contain 2 additional neurons so that it can predict between the 4 classes used in the project (3 different types of brain tumor and no brain tumor). The team's neural network and the baseline neural network will be compared by evaluating each model's accuracy in classifying brain scan images.

7 Ethical Considerations

The use of deep learning models in the medical industry has been growing at an unprecedented rate. These models are built to mimic human capability, but these advancements are making humans more susceptible to being indolent (AI for Social Good, 2024). The use of this technology, especially a brain tumour classification model, should be used in tandem with human knowledge and influence, as opposed to the model working independently. This is due to the fact that are various ethical considerations that prevents the sole use of deep learning applications for medical purposes.

7.1 Diagnosis Accuracy

With deep learning models, acquiring an accuracy of 100% is generally seen as not feasible. This gives birth to the rise of false negatives and positives. False negatives are terms coined in the event that something is true but labelled as false, while false positives are when something is false but labelled as true (NGSS Engineering Practices).

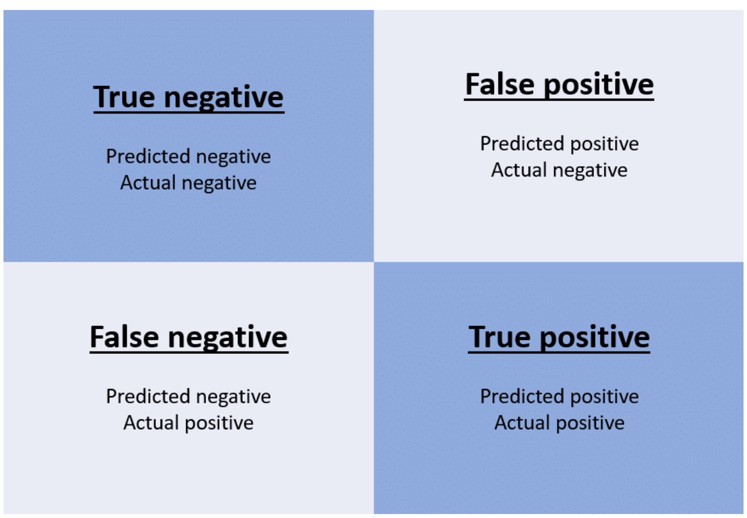

Figure 6: Confusion matrix. Image: NillsF blog

A false negative diagnosis (meaning that a brain tumour was not detected) would prevent or delay treatment, worsening patient outcomes. On the other hand, a false positive diagnosis (meaning that a brain tumour was falsely detected) would cause unnecessary mental and emotional distress to the patient. This would also mislead patients into undergoing unnecessary medical treatments that pose significant side effects and risks.

7.2 Limitations

With every model, having a wide variety of data to test is crucial, especially in the medical field, where every individual may vary based on a number of different factors. Due to this issue, more brain scans would be required to make a more accurate and applicable model. However, accessing more brain scans may be an issue, as most of these scans are not publicly accessible. Acquiring consent for the usage of this data could be seen as a tedious process, while using this data without proper consent could result in confidentiality issues within the organization and patients in question.

8 Potential Project Risks

This project can involve several potential risks that relate to either team dynamics and workload distribution, or the mechanics of the project itself. Below is a register containing these risks and the strategies that can be employed to mitigate these risks.

Table 1: Potential project risks, occurrence likelihoods, and mitigation strategies.

RISK	**LIKELIHOOD**	**MITIGATION STRATEGIES**
Group member drops the course	Low	All group members decide to split the missing member's workload into three even parts.
Group member(s) busy with other non-project related work	Medium	Schedule weekly meeting times and work times well in advance.
Model over fits to training data	High	Look at potentially removing layers to reduce network capacity, applying regularization, or using dropout layers (Carremans, 2018).
Model training takes longer than expected	Medium	Optimize code to ensure the use of the most efficient data structures, test different optimizers, or attempt to make use of transfer learning. (Parashar, 2022).

The following subsections below explain each of these risks in more detail, and specifically in the context of this project.

8.1 Potential Risk #1: a group member drops the course

Likelihood: *Low*

Each team member has expressed their willingness and enthusiasm to create a quality project. Furthermore, none of the team members are taking more than two courses this summer, so the chances of a member dropping this course due to a high workload are relatively low. However, unforeseen circumstances can occur, and if a team member drops the course, the remaining members do not have a problem dividing the missing team member's workload into three even parts.

8.2 Potential Risk #2: group member(s) busy with non-project related work

Likelihood: *Medium*

During this summer term, group members are also involved in various extracurricular activities such as part-time jobs, research positions, and design teams. These activities will likely reduce the amount of free time that the group has to work on the project. Therefore, the group will set meeting times and work periods a few days to a week in advance to ensure time is set aside to prioritize the project.

8.3 Potential Risk #3: model overfits training data

Likelihood: *High*

This project is based on image classification, which makes the group susceptible to overfitting the training data. In the context of MRI scan images, the dataset contains MRI photos taken from different angles and with variations in lighting. Furthermore, the appearance of brain structures differs from person to person. This can pose a problem, as a deep CNN model can simply ’memorize’ these images and not generalize well to the test data. To mitigate this problem, techniques such as data augmentation, reduction of layers, regularization, and dropout layers can be applied to help lessen the effects of overfitting. See Carremans (2018) for more solutions to this issue.

8.4 Potential Risk #4: model training takes longer than expected

Likelihood: *Medium*

The group understands that the training time for these models should not exceed the max session time on Google Colab. In the context of this project, training a deep CNN with several thousand images can make this issue likely to occur. To mitigate this risk, several interventions can be implemented. The code can be reviewed and tweaked to ensure optimality and efficiency, and different optimizers can be tested. See Parashar (2022) for more information on these techniques. Furthermore, the group can choose to employ a technique called transfer learning, which essentially takes a separate pre-trained model that can identify the general features of the MRI images, and then uses the final layers of the group’s model to locate the more specialized features of the MRI images (Brownlee, 2019). This can significantly reduce train time, as the starting layers of our model have been pre-trained.

9 Project Plans

The team used a Gantt chart to break down tasks and deadlines. The project timeline is split into 2-week segments to create checkpoints that ensure the team is on track to complete the project on time. Names are color-coded to signify which team members are assigned to specific tasks. However, collaboration is expected and encouraged across all tasks.

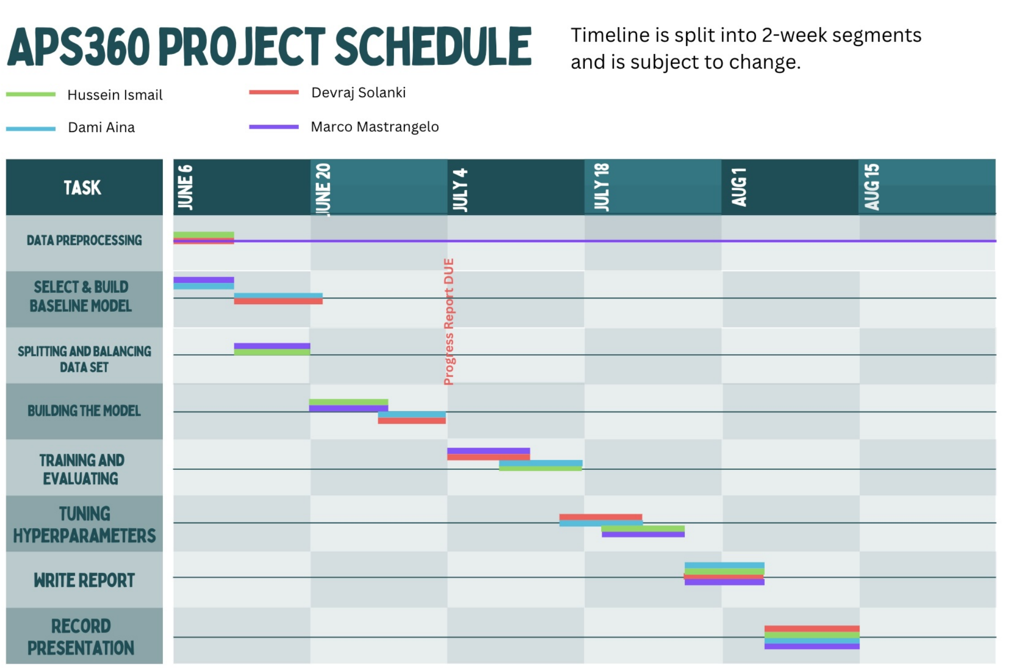

Figure 7: Project timeline and plan. Image: Hussein Ismail

10 Link to GitHub Repository

All project-related documents and code can be accessed at Team 20's GitHub repository using the link: `https://github.com/damilola-aina/Brain-Tumor-Detection`

References

Team AI for Social Good. Artificial intelligence – the double-edged sword that can make humans vulnerable and lazy. *AI Blog*, 2024.

Amazon. What is data augmentation. *Amazon Web Services*, 2024.

Jason Brownlee. A gentle introduction to transfer learning for deep learning. *Deep Learning for Computer Vision*, 2019.

Bert Carremans. Handling overfitting in deep learning models. *Towards Data Science*, 2018.

Lyron Foster. Building a medical image classifier with deep learning and python. *Medium*, 2019.

Michal. Futrega, Alexandre. Milesi, Michal. Marcinkiewicz, and Pablo Ribalta. Optimized u-net for brain tumor segmentation. *NVIDIA*, 2021.

Sandeep. Kumar, Shilpa. Choudhary, Arpit. Jain, Karan. Singh, Ali. Ahmadian, and Mohd Bajuri. Brain tumor classification using deep neural network and transfer learning. *Brain Topography*, 2023.

Prafful Mishra. Why are convolutional neural networks good for image classification. *Medium*, 2019.

Team NGSS Engineering Practices. Practices of science: False positives and false negatives.

Msoud Nickparvar. Brain tumor mri dataset, 2021. URL `https://www.kaggle.com/dsv/2645886`.

Quinn. Ostrom, Giono. Cioffi, Haley. Gittleman, Nirav. Patil, Kristin. Waite, Carol. Kruchko, and Jill. Barnholtz-Sloan. Cbtrus statistical report: Primary brain and other central nervous system tumors diagnosed in the united states in 2012–2016. *Neuro-Oncology*, 1 -100, 2019.

Abhay Parashar. How to train your deep learning models faster. *Comet Blog*, 2022.

Pytorch Official Documentation. *Resize*.

Asaf. Raza, Javed. Khan, Ijaz. Ahmad, Salama. Ahmed, Yousef. Daradkeh, Danish. Javeed, Ateeq. Rehman, and Habib Hamam. A hybrid deep learning-based approach for brain tumor classification. *MDPI Electronics*, 2022.

Olaf. Ronneberger, Philipp. Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. 2015.

Bhuvaji. Sartaj, Kadam. Ankita, Bhumkar. Prajakta, Dedge. Sameer, and Kanchan Swati. Brain tumor classification (mri), 2020. URL `https://www.kaggle.com/dsv/1183165`.

Naeem. Ullah, Ali. Javed, Ali. Alhazmi, Syed. Hasnain, Ali. Tahir, and Rehan. Ashraf. A unified deep learning model for brain tumor detection and classification. *PLOS ONE.*, 2023.

Yuting. Xie, Fulvio. Zaccagna, Leonardo. Rundo, Claudia. Testa, Raffaele. Agati, Raffaele. Lodi, David. Manners, and Caterina Tonon. Convolutional neural network techniques for brain tumor classification (from 2015 to 2022): Review, challenges, and future perspectives. *Diagnostics (Basel)*, 2022.