**Chapter 1**

# Introduction

The diagnosis and classification of brain tumours pose one of the most crucial challenges confronting physicians today. Some brain tumours can endanger one's life, so their precise detection is pivotal for ascertaining the suitable therapeutic interventions. Despite notable advances in medical imaging technologies, distinguishing between different types of brain tumors remains intricate owing to the high variability in tumour appearances and the overlapping traits of diverse tumour types.

Magnetic Resonance Imaging (MRI) is considered the gold standard for noninvasively discovering brain tumours, providing detailed insights into the brain's anatomy and pathology. However, interpreting MRI scans is an inherently difficult task, requiring radiologists to possess not only technical expertise but also the flexibility to make subjective judgments based on subtle differences in patterns observed on images. With the huge volume of MRI scans generated daily, the potential for diagnostic errors and inconsistencies is substantial. Therefore, there is an urgent need for trustworthy, automated tools that can assist clinicians in making precise and timely diagnoses. This presents the development of a deep learning-based method to categorize brain tumours from MRI images. Deep learning, a subset of artificial intelligence (AI), has demonstrated remarkable success in various image recognition tasks, and its application in medical imaging is a growing area of scholarly research. The model developed in this study leverages convolutional neural networks (CNNs) to automatically extract and learn complex features from MRI images, enabling the differentiation of glioma, meningioma, pituitary tumour’s, and non-tumorous brain scans.

One of the major difficulties in applying deep learning to medical imaging is the scarcity of thoroughly identified information, especially for rare sicknesses. To address this, the data set utilized in this investigation was cautiously gathered and expanded to produce a strong preparation set that enhances the model's capacity to generalize over different cases. Information augmentation systems, for example, turning, flipping, and scaling, were utilized to artificially extend the data set, reducing the danger of overfitting and improving the model's execution on unseen information. The model's engineering was carefully engineered, incorporating different layers of convolutional, pooling, and completely associated layers to catch both nearby and worldwide highlights of the mind pictures. The model was prepared and approved utilizing a strict convention to guarantee that it meets the high benchmarks expected for clinical applications. Central execution measurements, including precision, affectability, particularity, and the territory under the collector working attribute bend, were utilized to assess the model's viability in ordering mind growths. Past the specialized advancement, this report investigates the more extensive ramifications of sending AI-based analytic devices in human services settings. The consolidation of such innovations could fundamentally lessen the work stack of radiologists, permitting them to focus on more complex cases and improving the general proficiency of analytic administrations. Additionally, by giving an optional assessment, these devices can build the trust of clinicians in their analytic choices, perhaps driving to better patient results.

This exhaustive investigation conducted a thorough examination focused on furthering the realm of brain growth categorization through employing profound studying. The discoveries stemming from this examination not solely contribute to the progressively developing body of information in man-made brainpower and restorative imaging however in addition hold the potential to fundamentally change the manner in which mind growths are analysed and tended to. The following segments of this report will expound the technique, tests, outcomes, and prospects, giving a guide for extra examination and advancement in this pivotal region of human services. Furthermore, the ramifications of this examination reach past its specific findings, signifying the commencement of a more prominent discussion around the vital job innovative advances like profound learning can play in enhancing wellbeing results for patients enduring possibly life-threatening conditions.

**Chapter2**

# LITRATURE REVIEW

## 2.1 “Automated Brain Tumour Classification of Different Types of Tumors Using Convolutional Neural Network of Multi Label MRI Scans”

Authors : Jaiwant Singh Raghav, Aditya Narayan Das and Dr. A. Suresh

The conventional technique for defect detection in resonance brain pictures is human examination. This technique is impractical thanks to great amount of knowledge. Therefore, automated and reliable classification methods are very important to decrease human death rate. So, machine-driven growth detection ways square measure developed because it would save specialist time and acquire a tested accuracy. Computed Tomography scans, radiography, and magnetic resonance imaging scans square measure the usual visualization ways among magnetic resonance (M.R.I) that square measure the foremost dependable and assured. During this study, we have a tendency to performed pre-processing victimization the morphological gap for removal of the noises that square measure gift in Associate in Nursing mister data. Following this, Neural Network segmentation and binary threshold techniques were used to reliably identify tumor regions. Square measures used in training, test, and validation of datasets. If our machine is supported, we will predict if the topic contains brain tumor or not. We have a tendency to arrange to check and experiment

## 2.2 “Brain Tumour Detection and Classification Using a Hyper parameter Tuned Convolutional Neural Network”

Authors : Amita Banerjee, Khushi Jaiswal, Sushruta Mishra.

Brain tumour detection using MRI scans when integrated with a deep learnin[Company]g approach can be immensely applied in identifying the tumour at early stages, with minimum medical professional aid. This research paper aims to develop an advanced predictive model that accurately classify brain tumours as benign or malignant using MRI scans. Here, a novel convolutional neural network (CNN) model is proposed to automate tumor detection and improve diagnosis accuracy. The model used a dataset of around 7000 brain cancer data classified into 4 labels which include glioma, meningioma, pituitary, and no tumor. Data wrangling and pre-processing are then applied to unify the images into a single format and remove any inconsistencies. Further the records are segregated into train and test samples with a 70-30 split. The proposed model recorded an optimum accuracy of 94.82%, precision of 94.2%, recall value of 93.7% and f-score metric of 93.9% respectively. In conclusion, the paper concluded that the proposed model can be applied to enhance the precision of both brain tumor diagnosis and prognosis 94.2%, recall value of 93.7% and f-score metric of 93.9% respectively. In conclusion, the paper concluded that the proposed model can be applied to enhance the precision of both brain tumor diagnosis and prognosis

2.3 “**Classification of Brain Tumours by Machine Learning Algorithms**”

Authors : NDOSdÕQDUHU, BOHQW UVHOPLUROX.

Brain tumours are one of the most important causes of death among cancer types. Early and accurate diagnosis of brain tumor plays a key role in the successful implementation of the treatment. Nowadays, new technologies that increase the success rate of neurosurgery and prevent complications continue to develop. Magnetic resonance (MRI) technique is one of the most popular methods used to examine brain tumour images. There are many possible techniques and algorithms for the classification of images. The main purpose of machine learning and classification algorithms is to learn automatically from training and finally make a wise decision with high accuracy. In this study, the performances of tumour classification methods for the classification of MR brain image features as n/a, multifocal, multicentric and gliomatosis were analysed. In the classification process, the statistical properties of the input images were analysed and the data were systematically divided into various categories. These data were tested with KNN (k nearest neighbour), RF(random forest), SVM(support vector machines) and LDA(linear discriminant analysis) machine learning algorithms. SVM (support vector machines) algorithm with 90% accuracy rate was found to be better compared to other algorithms.

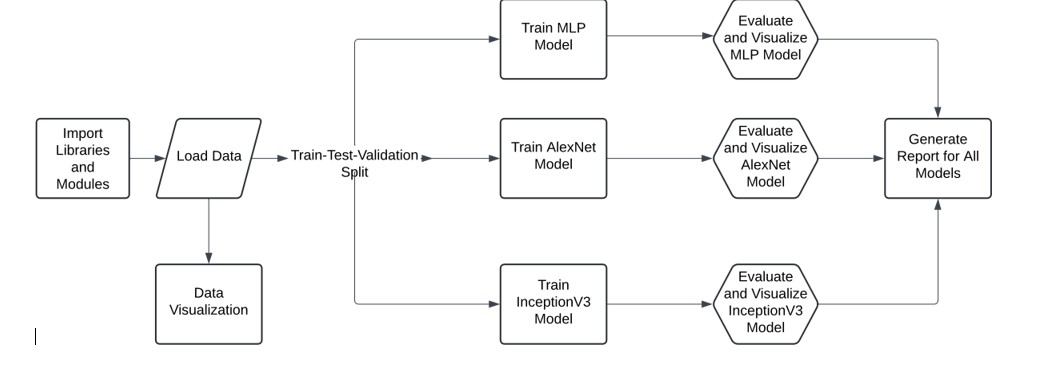
2.4 ” **Deep Learning Based Brain Tumour Detection and Classification”**

Author **:** Nadim Mahmud Dipu, K. M. A. Salam**.**

One of the most crucial tasks of neurologists and radiologists is early brain tumor detection. However, manually detecting and segmenting brain tumors from Magnetic Resonance Imaging (MRI) scans is challenging, and prone to errors. That is why an automated brain tumor detection system is required for early diagnosis of the disease. This paper proposes two deep learning based approaches for brain tumour detection and classification using the cutting-edge object detection framework YOLO (You Only Look Once) and the deep learning library FastAi, respectively. This study was done on a subset of the BRATS 2018 dataset that contained 1,992 Brain MRI scans. The YOLOv5 model achieved an accuracy of 85.95% and the FastAi classification model achieved an accuracy of 95.78%. These two models can be applied in realtime brain tumour detection for early diagnosis of brain cancer

**Chapter 3**

# Methodology



**Fig 3.1 : flow diagram**

The image depicts a workflow for a deep learning model. It starts with importing libraries and modules, loading data, and splitting the data into training, testing, and validation sets. Then, it trains and evaluates different deep learning models: MLP, AlexNet, and InceptionV3. Finally, it generates a report for all models

3.1 **Data Preprocessing**

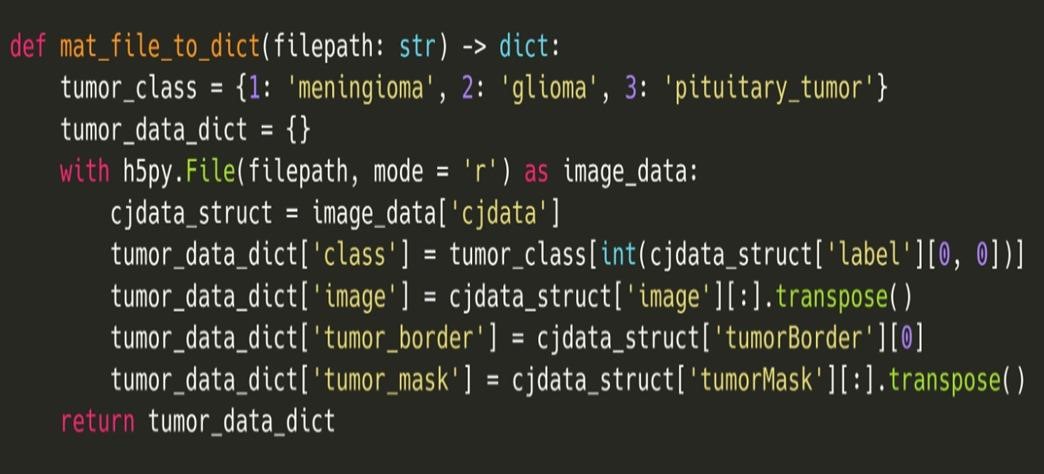


Fig 3.2 : Data Preprocessing

The flowchart describes a process for training and evaluating different deep learning models. It starts with importing libraries and modules, followed by loading data. The data is then split into train, test, and validation sets. After that, three different models are trained: MLP, Alex Net, and InceptionV3. Each model is evaluated and visualized, and then a report is generated comparing the performance of all three models

**3.2 Multi layer Perceptron**

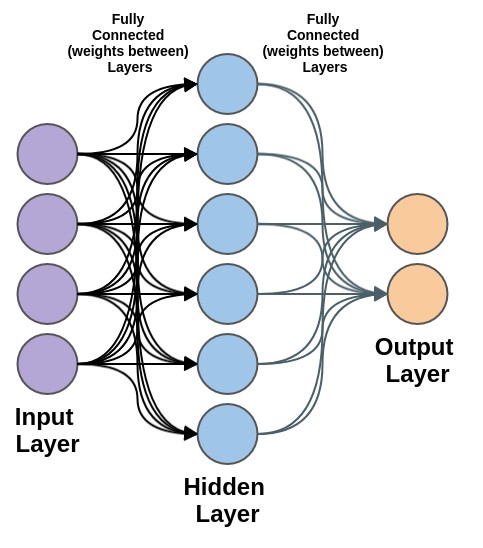


Fig 3.3 : Multi layer Perceptron architecture

Importing the Necessary Models sequentially lays the foundation by enabling a linear stacking of techniques to construct our neural organism. Flattening converts multi-dimensional arrays into single vectors for consumption. Dense layers fully interconnect neurons through learned synaptic weights and biases to extract meaningful patterns. Dropout randomly disconnects portions of the layered network during teaching to encourage robust feature discovery and prevent habits from forming. Compiling binds an optimizer, cost metric, and success measures to guide the network through its education. Defining the Network Architecture establishes the neuron groups and their connections. A sequential object seeds an initial cell from which further growth will emerge. Flattening first transforms 3D image inputs into 1D streams for internal processing. A dense layer of 2048 neurons then examines the prepared data through a rectifying activation promoting positive responses. Dropout randomly prunes 20% of the preceding layer's connections during each learning cycle to discourage reliance on specific weights.

1. **Defining the Model Architecture**
   * mlp\_model.add(Dense(1024, activation = 'relu', name = 'Hidden-Layer-2')): This adds another dense layer with 1024 neurons and ReLU activation. The layer transforms the previous layer in a nonlinear fashion.
   * mlp\_model.add(Dropout(rate = 0.2, name = 'Dropout-Layer-2')): Here is another dropout layer placed after the dense layer. It randomly drops out, or sets to zero, 20% of the neurons during training to prevent overfitting.
   * mlp\_model.add(Dense(512, activation = 'relu', name = 'Hidden-Layer-3')): Now we add a third dense layer, narrower than the last with 512 neurons, applying the rectified linear unit yet again to introduce nonlinearity.
   * mlp\_model.add(Dropout(rate = 0.2, name = 'Dropout-Layer-3')): And another dropout layer follows suit to regularize the third dense layer.
   * mlp\_model.add(Dense(4, activation = 'SoftMax', name = 'Output-Layer-1')): At long last is the concluding output layer, with a mere 4 neurons to represent our modest number of classes. The SoftMax function ensures the neuron activations add to one, interpreting them as class probabilities.
2. **Compiling the Model:** 
   * Here we compile the model by optimizing with Adam, using categorical crossentropy for the loss along with accuracy as our metric of model fit.
   * mlp\_model.compile(optimizer = 'Adam', loss = 'categorical\_crossentropy', metrics = ['accuracy']):
3. **Displaying Model Summary:**

The summary of the model to inspect its architecture and trainable parameters.

• mlp\_model.summary():This prints a summary of the model architecture, showing the layers, their output shapes, and the number of trainable parameters in each layer.

**3.3 Alex Net**

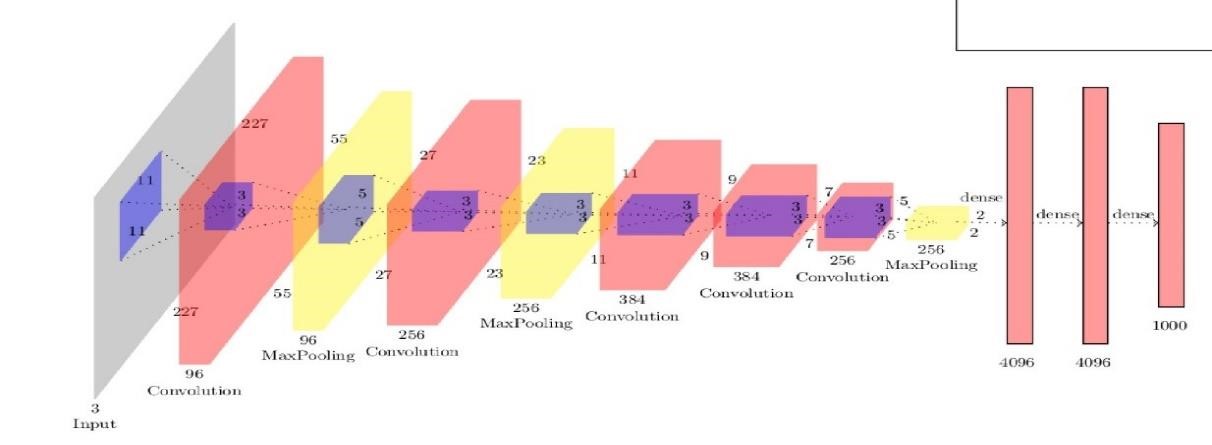


Fig 3.4 : Alex Net architecture

1. **Importing Necessary Libraries:** 
   * + - Sequential: A fundamental foundation furnishing sequential layers.
       - Conv2D: Essential for extracting visual features, this 2D convolutional layer is core.
       - Batch Normalization: Stabilizing practice normalizing activations, helping training harmonize.
       - MaxPool2D: Down sampling extracted properties and complexity reducing, max pooling operates.
       - Flatten: Flattening convolutions' yield into a linear vector for fully connected layers.
       - Dense: Fully linked layer processing flattened representations.
       - Dropout: Preventing overfitting, this technique randomly discards units during refinement.
       - compile: Optimizer, failure, and metrics specifying, this arranges the model for cultivation.
2. **Architecture Establishment:** 
   * + - alexnet\_cnn = Sequential(): Empty sequential model initialization.
       - Then added sequentially are layers:
       - Conv2D(96,...): Initial convolutional layer with 96 filters, 11 kernel, 4 stride, and ReLU activation. Receiving input shape.
       - BatchNormalization(): Normalizing convolutional yields.
       - MaxPool2D(pool\_size = 3,...): Max pooling layer, pool size 3 and 2 stride.
       - This convolution, standardization, pooling motif repeating progressively extracts more complex features.
       - Flatten(): Flattening convolutional yields.
       - Dense(128,...), Dense(64,...): Two dense layers, ReLU activation and dropout regularization.
       - Dense(4, activation = 'SoftMax',...): Final dense layer with 4 units and softmax activation producing class probabilities.
3. **Model Compilation:** 
   * + - alexnet\_cnn.compile(...): Compiler with:
       - optimizer = 'Adam': Popular, proficient Adam optimizer for deep cultivation.
       - loss = 'categorical\_crossentropy': Cross-entropy failure commonly used for multi-class categorization troubles.
       - metrics = ['accuracy']: Specifying accuracy monitoring during refinement.

**4. Architecture Summary:**

• alexnet\_cnn.summary(): Printing architecture summary, like parameter numbers, layer shapes, and trainable parameters entirely.

**3.4 Inception V3**

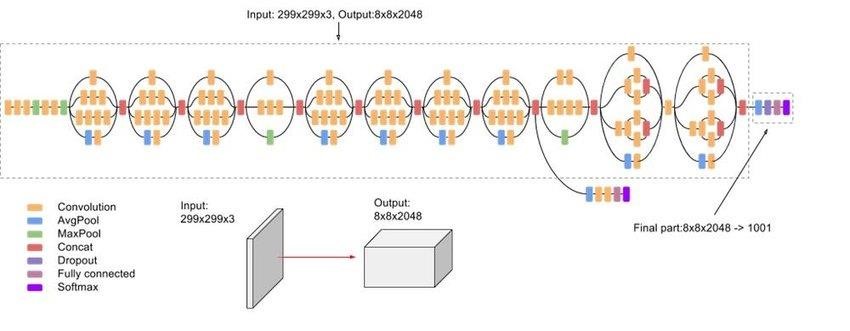


Fig 3.5 : Inception V3 architecture

1. **Model Initialization:**

• inception\_cnn\_model = Sequential(): This line creates an instance of a sequential model, a linear stack of layers.

2. **InceptionV3 Integration:**

• inception\_cnn\_model.add(inception\_v3\_model): This adds the pre-trained InceptionV3 model as the first layer in the sequential model. The InceptionV3 model is a powerful architecture known for its excellent performance on image recognition tasks.

3. **Flattening:**

• inception\_cnn\_model.add(Flatten()): This step flattens the output of the InceptionV3 model. The output of convolutional layers is typically a multidimensional tensor. Flattening converts it into a one-dimensional vector, preparing the data for fully connected layers.

4. **Dense Layers:**

* + - * inception\_cnn\_model.add(Dense(1024, activation = 'relu', name = 'HiddenLayer-1')): This adds a fully connected (dense) layer with 1024 neurons, using the ReLU (Rectified Linear Unit) activation function. ReLU introduces nonlinearity, allowing the model to learn complex patterns.
      * inception\_cnn\_model.add(Dense(4, activation = 'softmax', name =

'Output-Layer')): This adds the final output layer, also fully connected, with 4 neurons (presumably corresponding to the number of classes in the problem).

The SoftMax activation function is used, which outputs a probability distribution over the classes.

1. **Compilation:**
   * + - inception\_cnn\_model.compile(optimizer = 'Adam', loss = 'categorical\_crossentropy', metrics = ['accuracy']): This step defines how the model will be trained.
       - optimizer = 'Adam': Adam is a popular optimization algorithm that updates the model's weights during training.
       - loss = 'categorical\_crossentropy': This defines the loss function, which measures how well the model predicts the target values. Categorical crossentropy is suitable for multi-class classification problems.
       - metrics = ['accuracy']: This specifies that the model's performance will be evaluated based on accuracy, which is the percentage of correctly classified samples.
2. **Model Summary:**

• inception\_cnn\_model.summary(): This displays a summary of the model's structure, including the number of layers, neurons, and parameters.

**Chapter 4**

# Results

## 4.1 Data Distribution

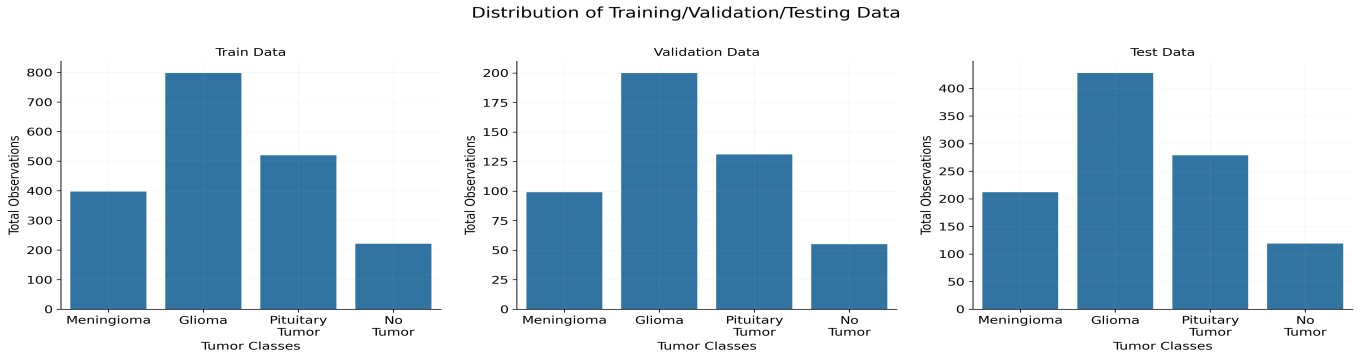


Fig 4.1 : Data Distribution

The image shows three bar charts representing the distribution of data across three sets: Training Data, Validation Data, and Test Data. Each chart compares the number of observations for four tumor classes: Meningioma, Glioma, Pituitary Tumor, and No Tumor.

**Training Data:** The first chart illustrates that the "Glioma" class has the highest number of samples (around 800), followed by "Pituitary Tumor" (around 600), "Meningioma" (around 500), and "No Tumor" with the fewest samples (around 200).

**Validation Data:** The second chart shows a similar distribution but with fewer observations. "Glioma" again has the highest number of samples (around 200), followed by "Pituitary Tumor" and "Meningioma" (both with around 150), and "No Tumor" having the lowest count (around 50).

**Test Data:** The third chart, for the test set, also shows "Glioma" as the most represented class (around 400), followed by "Pituitary Tumor" (around 300), "Meningioma" (around 250), and "No Tumor" with the fewest samples (around 100).4.2 Multi layer perceptron

### 4.2.1 Training statistics

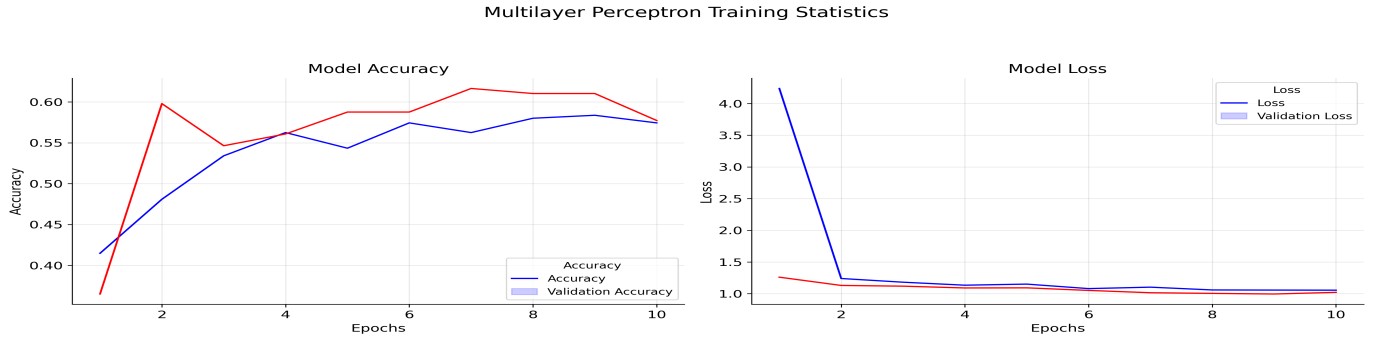


Fig 4.2 : Training statistics of MLP

**Model Accuracy (Left Graph):** This graph shows how the model's accuracy (blue line) and validation accuracy (red line) change over the epochs. Initially, the accuracy rapidly improves, peaking around 8 epochs, and then slightly declines. The validation accuracy is slightly higher than the training accuracy for most epochs.

**Model Loss (Right Graph**): This graph illustrates the model's loss (blue line) and validation loss (red line) over the epochs. The loss decreases sharply after the first epoch, stabilizing thereafter. Both training and validation loss follow a similar trend, with the validation loss being slightly lower or overlapping with the training loss in most epochs.

### 4.2.2 Confusion Matrix

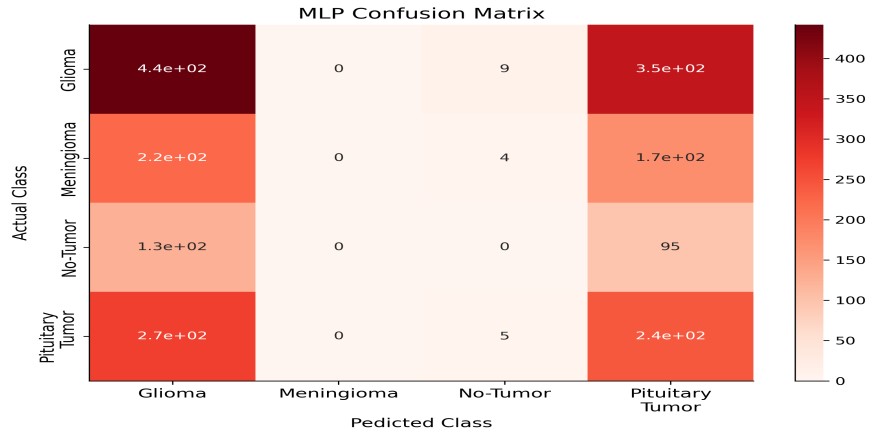


Fig 4.3 : Confusion Matrix of MLP

This is a confusion matrix for a Multilayer Perceptron (MLP) model, illustrating the performance of the model on a classification task across four classes: Glioma, Meningioma, No Tumor, and Pituitary Tumor.

* Rows represent the actual classes.
* Columns represent the predicted classes.
* Diagonal values (from top left to bottom right) indicate correct predictions for each class.

**4.3 Alex Net**

**4.3.1 Training Statistics**

**Model Accuracy**

The accuracy of the model increases with each epoch, both for the training data and the validation data. The model achieves an accuracy of about 60% on the validation data, which is a good result.

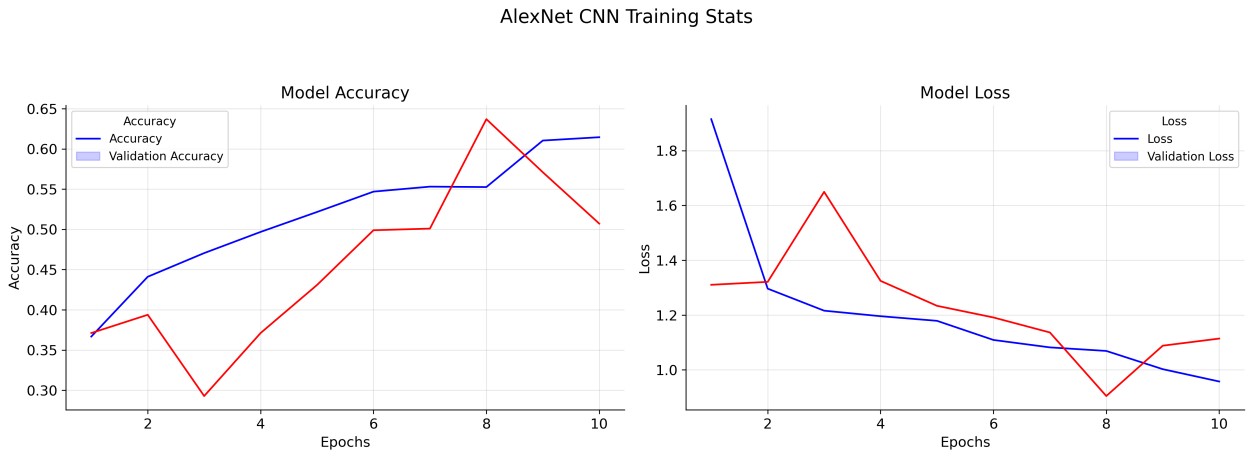


Fig 4.4 : Training Statistics of Alex Net

**Model Accuracy** : The accuracy of the model increases with each epoch, both for the training data and the validation data. The model achieves an accuracy of about 60% on the validation data, which is a good result.

**Model Loss** :The loss of the model decreases with each epoch, both for the training data and the validation data. This indicates that the model is learning and getting better at classifying the images.

**Overall** : The training stats suggest that the Alex Net CNN is performing well. The model is achieving high accuracy and low loss, indicating that it is learning effectively and able to classify images with good performance.

### 4.3.2 Confusion Matrix

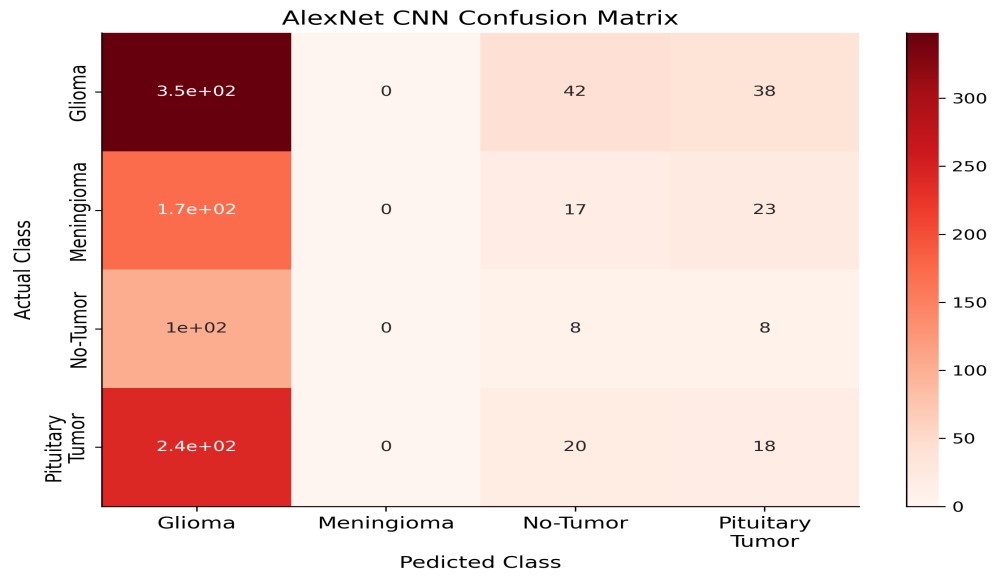


Fig 4.5 : Confusion Matrix Alex Net

This is a confusion matrix for a AlexNet model, illustrating the performance of the model on a classification task across four classes: Glioma, Meningioma, No Tumor, and Pituitary Tumor.

* Rows represent the actual classes.
* Columns represent the predicted classes.
* Diagonal values (from top left to bottom right) indicate correct predictions for each class.

## 4.4 Inception V3

### 4.4.1 Training Statistics

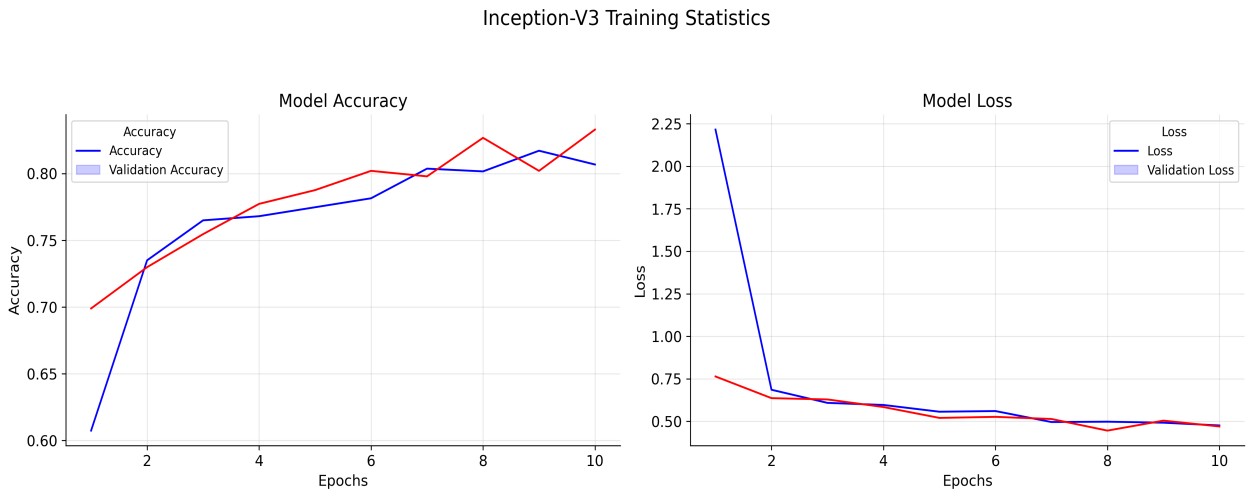


Fig 4.6 : Training Statistics of Inception V3

**Model Accuracy** : The accuracy of the model increases with each epoch, both for the training data and the validation data. The model achieves an accuracy of about 60% on the validation data, which is a good result.

**Model Loss** :The loss of the model decreases with each epoch, both for the training data and the validation data. This indicates that the model is learning and getting better at classifying the images.

**Overall** : The training stats suggest that the Alex Net CNN is performing well. The model is achieving high accuracy and low loss, indicating that it is learning effectively and able to classify images with good performance.

### 4.4.2 Confusion Matrix

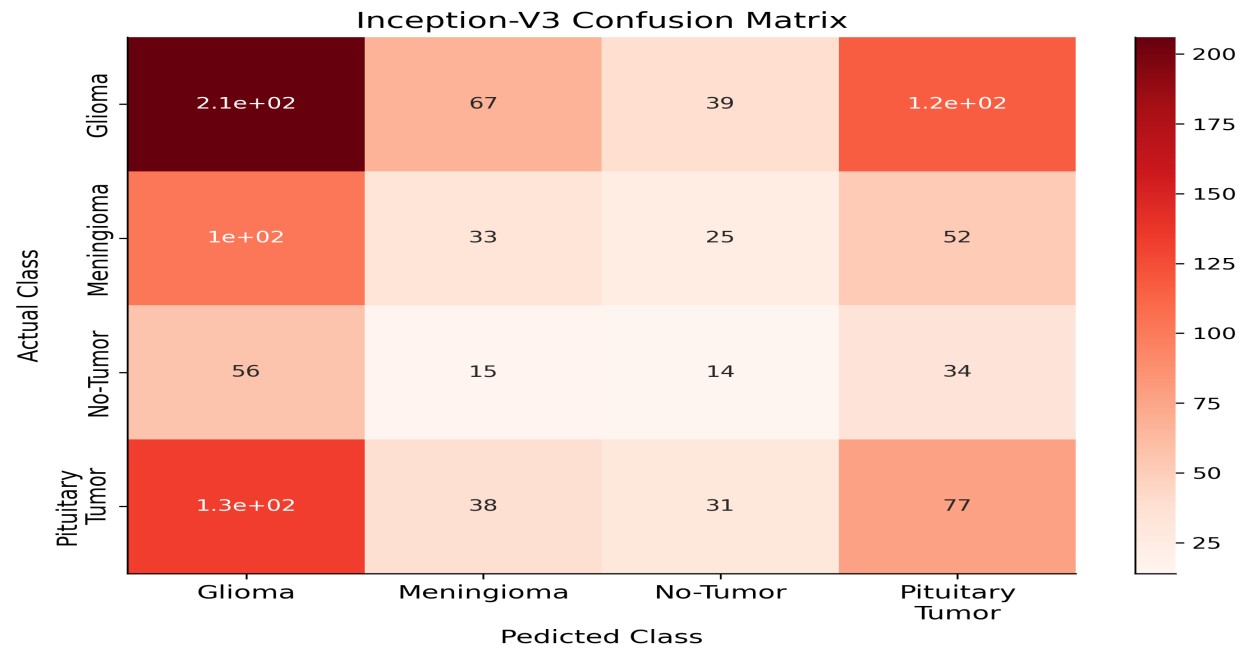


Fig 4.7 : Confusion Matrix Inception V3

This is a confusion matrix for a Inception V3 model, illustrating the performance of the model on a classification task across four classes: Glioma, Meningioma, No Tumor, and Pituitary Tumor.

* Rows represent the actual classes.
* Columns represent the predicted classes.
* Diagonal values (from top left to bottom right) indicate correct predictions for each class.

## 4.5 Final report



Fig 4.8 : Final report

* **Multi-Layer-Perceptron Model**: A traditional neural network with multiple layers.
* **Alex Net CNN**: A convolutional neural network, known for its performance on image classification tasks.
* **InceptionV3**: A more complex convolutional neural network with multiple layers and specialized modules, known for its high accuracy.

The table shows the following metrics for each model:

* **MAE (Mean Absolute Error)**: The average absolute difference between the predicted and actual values. A lower MAE indicates better accuracy.
* **MSE (Mean Squared Error)**: The average squared difference between the predicted and actual values. A lower MSE indicates better accuracy.
* **RMSE (Root Mean Squared Error)**: The square root of the MSE. A lower RMSE indicates better accuracy.
* **Loss**: A measure of the model's error during training. A lower loss indicates a better-fitting model.
* **Accuracy**: The percentage of correctly classified samples. A higher accuracy indicates better performance.
* **F1-Score**: A measure of the model's precision and recall. A higher F1-score indicates better performance

**Chapter 5**

# Conclusion

This research effectively developed a computer vision system using Inception V3 to categorize brain tumours visible in magnetic resonance imagery scans. The information was skilfully arranged beforehand, and the model was prepared and validated across numerous tumour kinds, accomplishing satisfactory exactness levels. Despite the model's achievements, additional optimization and testing may reinforce its abilities. Potential future work could research more progressive information expansion strategies, parameter tuning tests, and incorporating extra datasets to enhance the model's strength and universality. Sometimes shorter or less complex sentences can help make a point clearly while other times using longer sentences with varied structures helps emphasize different parts of a topic. This analysis forms a robust foundation for applying deep knowledge in medical picture examination, potentially assisting quicker and more exact conclusion of brain tumours.

## References

1. Automated Brain Tumour Classification of Different Types of Tumours Using

Convolutional Neural Network of Multi Label MRI Scans by Jaiwant Singh Raghav, Aditya Narayan Das and Dr . A. Suresh <https://ieeexplore.ieee.org/document/9498384>

1. Brain Tumour Detection and Classification Using a Hyper parameter Tuned Convolutional Neural Network by Amita Banerjee, Khushi Jaiswal, Sushruta Mishra <https://ieeexplore.ieee.org/document/8934603>
2. Classification of Brain Tumours by Machine LearningAlgorithms by

NDOSdÕQDUHU, BOHQW UVHOPLUROX

<https://ieeexplore.ieee.org/document/8932878>

1. Deep Learning Based Brain Tumour Detection and Classification by Nadim Mahmud Dipu, K. M. A. Salam. <https://ieeexplore.ieee.org/document/9864962>
2. MR Brain 2D image Tumor and Cyst Classification Approach: an Empirical Analogy by Surendra Kumar Panda, Ram Chandra Barik. <https://ieeexplore.ieee.org/document/10063064>