JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

BRAIN TUMOUR DETECTION AND CLASSIFICATION USING IMAGE PROCESSING AND DEEP LEARNING ALGORITHMS

[1]Thota Siva Pavani, [2]KompellaVenkata Ramana

[1]M. Tech Scholar, [2]Professor
[1,2]Department of Computer Science & Systems Engineering,
[1,2]Andhra University College Of Engineering(A), Andhra University, Visakhapatnam, India

Abstract*:* Brain tumours are an aggressive disease, affecting thousands worldwide. In 2023, an estimated 94,390 people will receive a primary brain tumour diagnosis, and around 18,990 people are expected to die from malignant brain tumours (brain cancer). Given their severe physical, cognitive, and psychological impacts, early detection and accurate classification are critical for effective treatment and improved patient outcomes. Magnetic resonance imaging (MRI) is the most reliable technique for brain tumour detection. Leveraging advancements in deep learning, particularly convolutional neural networks (CNN), neural networks have demonstrated high accuracy in image classification and segmentation tasks. This project explores a comparative analysis of deep learning models to detect and classify brain tumours into three categories: gliomas, meningiomas, and pituitary tumours. Using a dataset containing 3,190 T1-weighted contrast-enhanced images, which were cleaned and augmented, the best-performing CNN model, consisting of three convolutional layers, achieved an accuracy of 90%. The proposed system integrates CNN and RCNN models for precise tumour segmentation, enabling faster and more reliable diagnoses and ultimately enhancing patient life expectancy and quality of care.

Keywords: brain tumour detection, classification, image processing, deep learning, CNN, RCNN, MRI scan, medical diagnostics, automated tumour prediction.

I. INTRODUCTION

A brain tumour, or intracranial tumour, refers to abnormal and uncontrolled brain tissue growth. The immune system often fails to recognise these tumours as foreign, as they carry the patient’s DNA, making it challenging for the body to mount a defensive response. The World Health Organisation (WHO) classifies brain tumours into four grades (I-IV), with glioma being a Grade I tumour, which is slow-growing and the least harmful. At the same time, meningioma is a Grade III tumour with higher chances of recurrence. Additionally, pituitary tumours, though benign, can invade the central nervous system (CNS) due to their proximity to the brain. Early and accurate detection of such tumours is critical, as they can metastasise and grow rapidly, posing significant risks to the patient's health. The complexity of brain tumour identification and classification poses challenges for physicians and radiologists, particularly in underdeveloped regions where access to expert healthcare is limited. These tasks involve intricate image analysis, localisation of tumour regions, and comparison with adjacent tissues. In such cases, expert radiologists apply filters to enhance image clarity, identify tumour types, and assess their growth stage. However, this manual process can be time-consuming and prone to error.

Recent technological advancements in artificial intelligence (AI), specifically in the areas of computer vision, image classification, and image segmentation, offer promising solutions for addressing these challenges. Deep learning, a subfield of AI, has demonstrated its potential in medical image analysis. Deep learning models, such as convolutional neural networks (CNNs), can automatically detect and classify tumours by learning from vast datasets of medical images. CNNs are particularly effective for image classification tasks, as they utilise layers of neurones to identify and extract key features from images.

In this project, I present a study of CNNs for brain tumour classification using a dataset that was augmented and expanded to a total of 66,960 images. Various CNN models were designed and trained to detect and classify brain tumours. The best-performing model, consisting of three convolutional layers and one dense layer with 128 nodes, achieved a test accuracy of 90%, a test loss of 0.43, and an F1 score of 91.

This project aims to automate the detection and classification of brain tumours from MRI scans, enabling faster and more accurate diagnosis. The system integrates state-of-the-art image processing techniques and deep learning algorithms, such as RCNN, to enhance tumour segmentation and detection. Furthermore, the project incorporates colour-coded visualisations to distinguish between actual tumours, tumour spread areas and predicted tumour growth, aiding medical professionals in their decision-making processes.

By combining AI-driven analysis with medical imaging, this project offers a scalable and efficient solution for brain tumour detection, ultimately contributing to improved patient care and medical research.

II. LITERATURE SURVEY

In the work ‘Brain Tumour Classification Using Convolutional Neural Networks’ proposed by J. Seetha and S. Selvakumar Raja [1]: The automatic brain tumour detection is performed by using fuzzy C means (FCM)-based segmentation, texture and shape feature extraction, and SVM and DNN-based classification. They used the ‘image net’ database for classification. So the training was performed for only the final layer. They concluded with a training accuracy of 97.5%. Ali Ari and Davut Hanbay [2] proposed a ‘Deep learning-based brain tumour classification and detection system’. The system is implemented to distinguish the tumours into two classes: benign and malignant. The proposed system has three stages, which are preprocessing the extreme learning machine local receptive fields (ELM-LRF)-based tumour classification and image processing-based tumour region extraction. The system is restricted to only cranial MR images, which have mass. The experimental system hasachieved an accuracy of 97.18%.

From the Department of Biomedical Engineering, Helwan University, Cairo, Egypt, the authors Hossam H. Sultan, Nancy M. Salem, and Walid Al-Atabany [3] proposed ‘Multi Classification of Brain Tumour Images’. In the work done, the authors used CNN to classify three types of brain tumours-meningioma, glioma, and pituitary tumour-and identify their types and grades. Their first dataset included MRIs with three different views: axial, coronal, and sagittal views. The second dataset had images on T1-weighted contrast-enhanced that included different grades of glioma (Grade II, Grade III, and Grade IV). Their proposed network had 16 layers starting from the input layer, which holds the preprocessed images passing through the convolution layers and their activation functions (3 convolution, 3 ReLU, normalisation, and 3 Max pooling layers). They also had two dropout layers, a softmax layer, and finally a classification layer that produces the predicted class. This proposed architecture had achieved an accuracy of 98.7%.

III. EXISTING SYSTEM

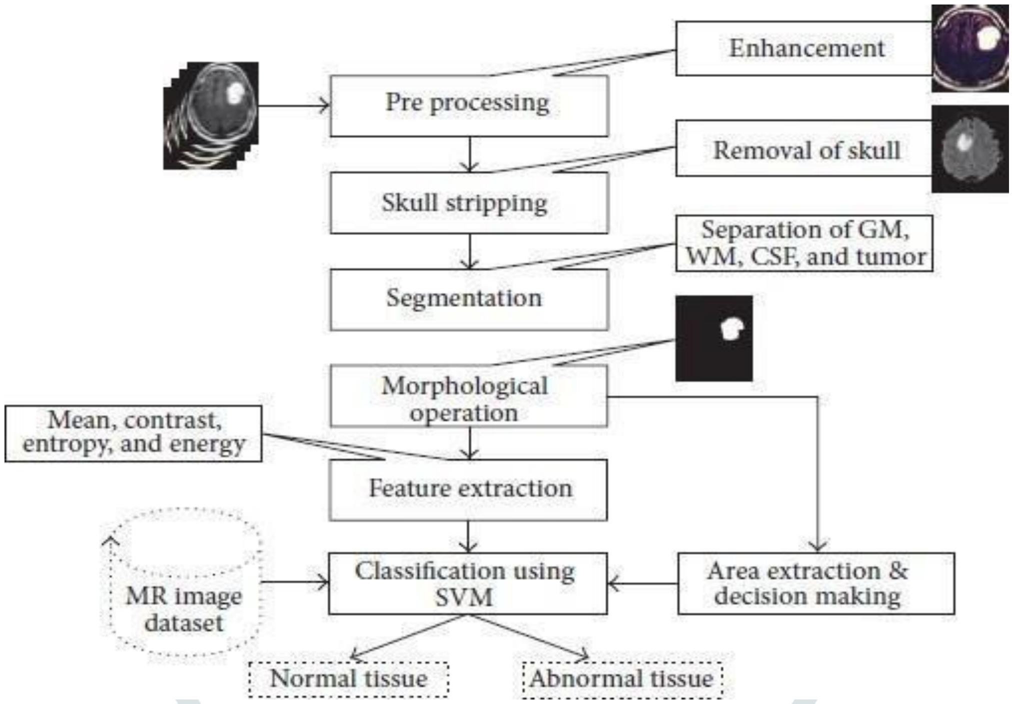


Figure 1: Workflow of Existing System.

In the first stage, there is a computer-based procedure to detect tumour blocks and classify the type of tumour using a convolutional neural network algorithm for MRI images of different patients. The second stage involves the use of different image processing techniques such as histogram equalisation, image segmentation, image enhancement, morphological operations, and feature extraction that are used for brain tumour detection in the MRI images for the tumour-affected patients. This work introduced one automatic brain tumour detection method to increase the accuracy and decrease the diagnosis time.

IV. PROPOSED SYSTEM

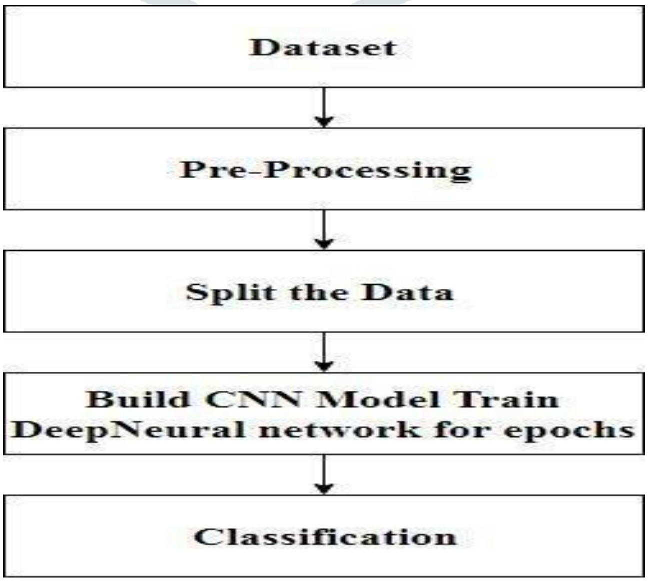


Figure 2: Workflow of Proposed System.

The proposed system has mainly five modules. Dataset, preprocessing, splitting the data, building a CNN model, training a deep neural network for epochs, and classification. In the dataset, I can take multiple MRI images and take one as the input image. In preprocessing the image, encoded the label and resized the image. In splitting the data, I set the image as 75% training data and 25% testing data. Then a built CNN model trains deep neural networks for epochs. Then the image is yes or no; if the tumour is positive, then it returns yes, and if the tumour is negative, then it returns no.

V. OBJECTIVE

The system leverages cutting-edge deep learning techniques, powered by TensorFlow and a T4 GPU, to discern subtle images. With CNN as its backbone, the system delves deep into image understanding, enhancing its ability to accurately distinguish between the two. Through rigorous training and precise binary classification, the system achieves spot-on identification, classification, and detection of tumour images. To provide doctors with good software to identify tumours and their causes. Save patients’s time. Provide a solution appropriately in the early stages. Get timely consultation.

VI. METHODOLOGY

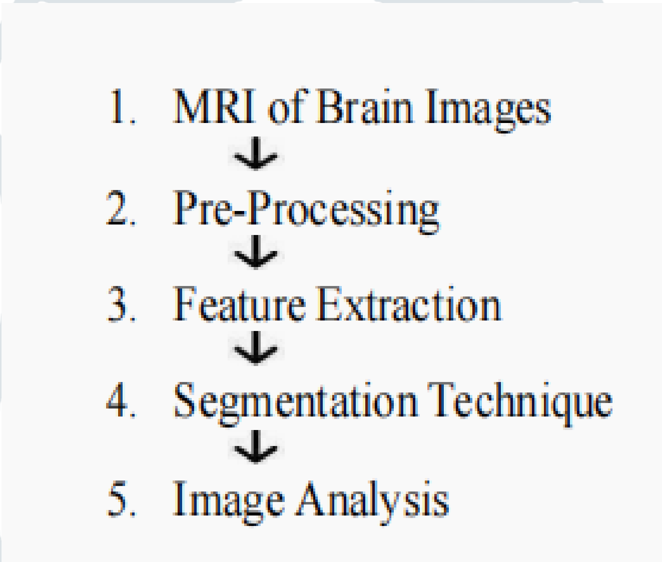


Figure 3: Brain tumour detection steps

Data Set: Magnetic resonance imaging (MRI) is a medical imaging technique used in radiology to produce high-quality two-dimensional or three-dimensional images of the brain and brainstem. The database includes two-dimensional T1-weighted contrast-enhanced images. The dataset includes three different views: axial, coronal, and sagittal views. These images were classified into 4 classes. The number of images obtained for different tumour types is:

Type	Training Sample	Testing Sample	Total
Glioma	703	100	803
Meningioma	790	115	905
No Tumour	678	95	668
Pituitary	720	100	814

Table 1: Initial Datasets

Data Preprocessing: In the pre-processing stage, I try to create uniformity in data before feeding it to neural networks. The images had high resolution, and I downsized them to 224x224x3. This helps preserve all relevant information for networks in fewer file sizes. The MRIs contained a black background around the central image of the brain. This black background provides no useful information for classification and would be wasted if fed to neural networks. Hence the images were cropped around the main contour. Here the biggest contour is selected and marked. Next, we find the extreme points of the contour and crop the image on those endpoints. Thus, I removed most of the unwanted background and some noise present in the original image. This process is done for each image in the dataset. However, sometimes the contour mapping and cropping algorithm was not able to correctly recognise the correct contour and wrongly cropped the image. Such images resulted in distorted images and were removed by manual inspection after the 'augmentation' phase.

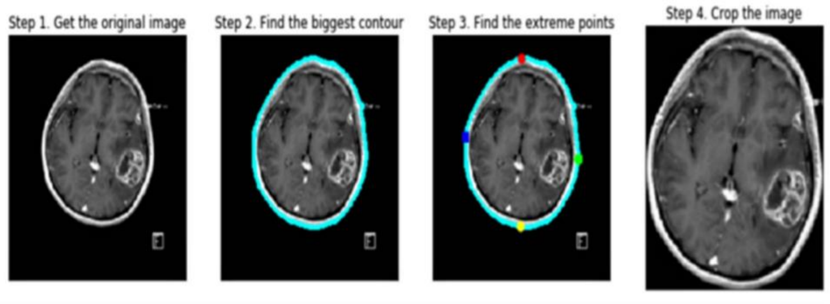


Figure 4: Contour Cropping

Here the biggest contour is selected and marked. Next, I find the extreme points of the contour and crop the image on those endpoints. Thus, I have removed most of the unwanted background and some noise present in the original image. This process is done for each image in the dataset. However, note that sometimes the function may not be able to correctly recognise the correct contours and makes a mistake and wrongly crops the image. Such images should be removed by manual inspection before entering the phase. The amount of data gathered was very low and could cause the models to underfit. Hence, I could use a brilliant technique of data augmentation to increase the amount of data. This technique relies on rotations, flips, changes in exposure, etc. to create similar images.

Data Augmentation: I have a few images to train and test a neural network. For a network to learn to a greater extent, I can increase the amount of data that is fed to it. This was done by using the technique of 'Data Augmentation'. This technique relies on rotations, flips, changes in exposure, zoom, etc. to create similar images. The process of augmentation was applied to every image of the dataset; this caused the number of images to increase by a factor of 21. Thus, with so much huge data, the chances of underfitting the network can be reduced.

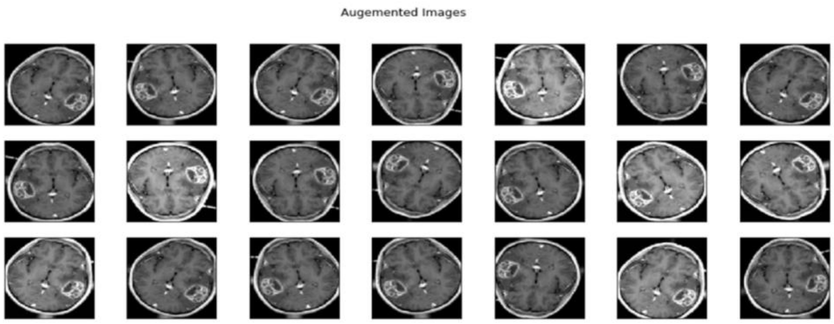


Figure 5: Augmentation

Model Selection: After careful consideration of the task requirements and computational resources, I opted to use the CNN architecture as the backbone of our deep learning model. CNN is a widely used pre-trained model known for its excellent performance in image classification tasks.

Model Training: I divided the preprocessed dataset into a ratio of 3:1 where the training and validation sets, with 75% of the data used for training and 25% for validation. I then trained the CNN model on the training set, fine-tuning its parameters using the Adam optimiser and binary cross-entropy loss function.

[1] Training Data

No	Class	Initial Count	Augmented Count	Discarded Count	Final Count
1	Glioma	703	14763	750	14013
2	Meningioma	790	16590	1000	15590
3	No tumour	678	14238	150	14088
4	Pituitary	720	15120	80	15040

Table 2: Training Data

[2] Testing Data

No	Class	Initial Count	Augmented Count	Discarded Count	Final Count
1	Glioma	100	2100	85	2015
2	Meningioma	115	2414	126	2288
3	No tumour	95	1995	63	1932
4	Pituitary	100	2100	98	2002

Table 3: Testing Data

Few images were distorted during the pre-processing and augmentation phase. These images were distorted due to improper cropping of the contour. Such images were noisy and unclear and thus were not collected in the final data set. I concluded with a total of 58731 training and 8237 testing/validation images.

Model Evaluation: Once training was complete, I evaluated the trained model's performance on a held-out test set that was not seen during training. I computed various evaluation metrics, including accuracy, precision, recall, and F1 score, to assess the model's effectiveness in classifying the images.

Performance Analysis: Finally, I analysed the model's evaluation metrics and visualisations, such as confusion matrices and ROC curves, to gain insights into its performance. This analysis helped us identify areas for improvement and fine-tune the model's hyperparameters to achieve better detection accuracy and robustness.

VII. DESIGN

7.1 ARCHITECTURE DIAGRAM:

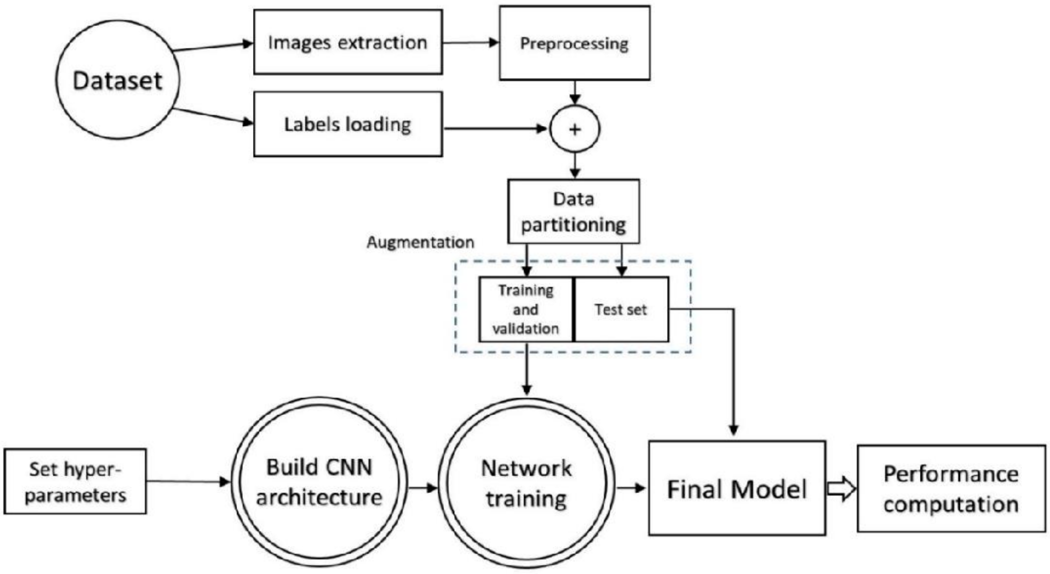


Figure 6: Architecture of the system

VIII. ALGORITHM USED

A. *Convolution Neural Network:*

Convolutional Neural Networks (CNN) is one of the variants of neural networks that have shown unprecedented accuracies in the field of image classification. The hidden layers of a CNN typically consist of convolutional layers, pooling layers, fully connected layers (dense layers), and normalisation layers. A convolution layer reads the input image and applies multiple filters to the image, and a feature map is created. All these filters are initialised randomly and become the parameters, which will be learnt by the network subsequently during backpropagation. These filters help in extracting the right and relevant features from the input data. CNN captures the spatial features from an image. Spatial features refer to the arrangement of pixels and the relationship between them in an image. They help in identifying the object accurately, the location of an object, as well as its relation with other objects in an image. After this, it applies a ReLU function on the feature map, which helps increase non-linearity. Next, it applies a pooling layer to each feature map, which progressively reduces the spatial size of the representation to reduce the number of parameters and computations in the network. Then it flattens the pooled images into one long vector, which is input for a fully connected artificial neural network. Further, the network works exactly as an ANN, where output is generated in the final layer and error is backpropagated into the network. This error then helps the network adjust the weights to minimise the error in the next consecutive epochs. For this paper, we designed and trained 27 different CNN models on our augmented data set. Figure 3 shows our proposed CNN model, which is named 3-conv-128-nodes-1-dense and has a total of 16 layers. The model begins with an input layer, which accepts images in a batch size of 32 with a size of 224x224x3. Then we have 3 convolution layers along with their activation functions and max-pooling layers. To prevent overfitting, a dropout layer is used, followed by a fully connected layer of 128 nodes, a softmax layer to predict the output, and finally, a classification layer that produces output in one of the four predicted classes.

The detailed explanation of the network is as follows: The input layer reads an image in a resolution of 224x224x3 and sends it down to one of the three convolution layers. The convolution layers are a bundle of convolution, activation, and max-pooling layers. Firstly, the convolution layer applies sliding K convolutional filters (kernels) of size M × N over the input images by moving the filters along the input and computing the dot product of the weights and the input. These kernels are used as feature identifiers, such that kernels in the early layers detect only low-level features like outlines, boundaries, and spots, while advanced ones are used to detect more and more complex features. Each of our three convolution layers is succeeded by an activation layer; the activation function of a node defines the output of that node given an input or set of inputs. For our model, I have used the Rectified Linear Unit (ReLu). Next in the network, I have a pooling layer. The addition of a pooling layer is used for ordering layers within a neural network that may be repeated one or more times in a given model. The pooling layer operates upon each feature map separately and creates a new set of the same number of pooled feature maps, but the pooling layer will always reduce the size of each feature map by the pooling factor. In Max Pooling, each of the regions is represented by the filter, and we take the max of that region and create a new output matrix where each element in the new matrix is the max of a region filter from the original input. After our final max-pooling of the third convolution layer set, the output is of the size 17x17x128. In a neural network, a fully connected layer occupies most of the parameters, and hence, neurones develop codependency amongst each other during training, which dampens the individual power of each neurone, leading to overfitting of training data. A dropout is an approach of regularisation in neural networks that helps reduce interdependent learning amongst the neurones in the fully connected layers.

Hence, after max-pooling, I have a dropout layer. When I have such a layer during training, some number of layer outputs are randomly ignored or “dropped out.” This has the effect of making the network architecture look like and be treated-like layers with a different number of nodes and connectivity to the prior layer. In effect, each update to a layer during training is performed with a different “view” of the configured layer. Finally, I conclude the network with a fully connected layer of 128 nodes followed by the activation layer softmax, which gets us output in one of the four classes. The accuracy of our best CNN model was 90%, and the value of the loss was 0.43.

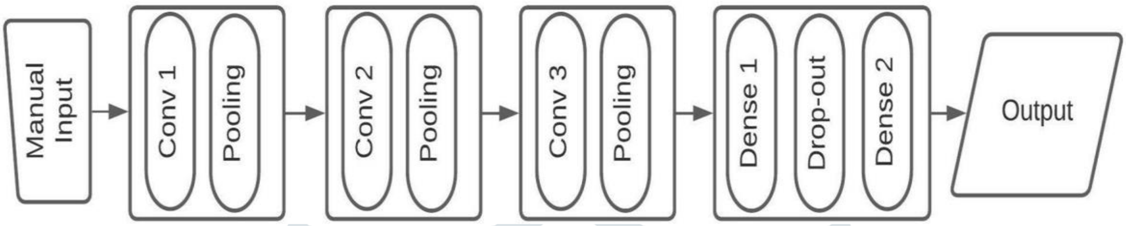


Figure 7: CNN Model Architecture

CNN Models	Test Accuracy	Test Loss	F1 - Score
1-conv-32-nodes-0-dense	86	1.77	86
2-conv-32-nodes-0-dense	85	0.76	85
3-conv-32-nodes-0-dense	88	0.48	88
1-conv-64-nodes-0-dense	85	2.39	85
2-conv-64-nodes-0-dense	89	0.55	89
3-conv-64-nodes-0-dense	89	0.55	89
1-conv-128-nodes-0-dense	84	1.36	84
2-conv-128-nodes-0-dense	82	1.01	81
3-conv-128-nodes-0-dense	86	0.95	86
1-conv-32-nodes-1-dense	24	1.38	10
2-conv-32-nodes-1-dense	88	0.59	89
3-conv-32-nodes-1-dense	86	0.58	86
1-conv-64-nodes-1-dense	26	1.36	13
2-conv-64-nodes-1-dense	88	0.48	89
2-conv-64-nodes-1-dense	88	0.54	89
1-conv-128-nodes-1-dense	85	0.73	85
2-conv-128-nodes-1-dense	85	0.77	83
3-conv-128-nodes-1-dense	90	0.43	91

1-conv-32-nodes-2-dense	24	1.38	10
2-conv-32-nodes-2-dense	24	1.38	10
3-conv-32-nodes-2-dense	63	0.84	57
1-conv-64-nodes-2-dense	24	1.38	10
2-conv-64-nodes-2-dense	83	0.70	83
3-conv-64-nodes-2-dense	85	0.55	86
1-conv-128-nodes-2-dense	24	0.00	10
2-conv-128-nodes-2-dense	85	0.78	87
3-conv-128-nodes-2-dense	85	0.64	85

Table 4: CNN Model

B. Region-based Convolutional Neural Network (RCNN):

In this project, RCNN (Region-based Convolutional Neural Network) is employed to enhance the accuracy and efficiency of brain tumour detection and classification from MRI scans. The initial step involves meticulous data preparation, where MRI images are annotated with precise bounding boxes around tumour regions and subjected to augmentation techniques such as rotation, scaling, and flipping to enrich the dataset and improve model robustness. Region proposals, which are potential areas where tumours might be located, are generated using advanced algorithms like selective search or EdgeBoxes.

Each proposed region is then processed using a convolutional neural network (CNN) architecture, such as VGG16 or ResNet, which extracts detailed features from these regions. These features are essential for distinguishing between different types of tumours and for accurate localisation. To classify the tumours, a Support Vector Machine (SVM) or similar classifier is employed, which interprets the extracted features and assigns tumour types and grades.

Bounding box regression is applied to fine-tune the coordinates of the detected tumours, improving the precision of tumour localisation. The RCNN model undergoes rigorous training on the annotated and augmented dataset, with careful evaluation using metrics such as precision, recall, accuracy, and intersection over union (IoU) to ensure its effectiveness. Performance validation is carried out on a separate validation set to mitigate overfitting and to confirm the model's generalisability across different cases.

Furthermore, the integration of advanced post-processing techniques, such as non-maximum suppression, helps in eliminating duplicate detections and refining the final tumour boundaries. The finalised RCNN model is then deployed within a clinical environment, offering healthcare professionals a robust tool for automated tumour detection and classification. The system also includes visualisation features that enhance the interpretability of results by applying colour-coded overlays to differentiate between actual tumours, spread areas, and predicted growth regions. This comprehensive approach aims to streamline the diagnostic process, providing timely and accurate support for medical decision-making and improving patient outcomes.

IX. LIBRARIES AND PACKAGES

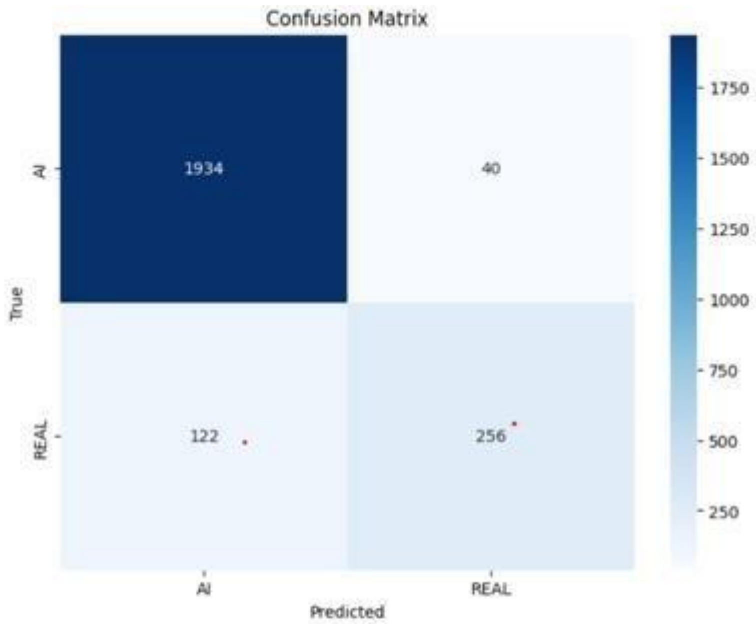
Import all necessary Libraries: The libraries imported in this project are TensorFlow, TensorFlow Keras layers, TensorFlow Keras models, TensorFlow Keras Adam optimiser, TensorFlow Keras LearningRateScheduler callback, TensorFlow Keras, EarlyStopping callback, Matplotlib pyplot, TensorFlow Keras application, Scikit-learn confusion_matrix function, Scikit-learn classification_report function, Seaborn, NumPy, and OpenCV.

GPU Availability and Memory Growth Configuration: As we move forward, the code ensures optimal GPU utilisation by checking for the availability of GPUs and configuring memory growth accordingly. Initially, it utilises TensorFlow's 'list_physical_devices' function to retrieve a list of physical GPU devices. Subsequently, it iterates through each GPU in the list, enabling memory growth using'set_memory_growth'. This configuration allows TensorFlow to dynamically allocate memory as needed, ensuring efficient utilisation of GPU resources during model training.

X. RESULT ANALYSIS

In the result analysis, I evaluate the model's performance through multiple metrics and visualisations:

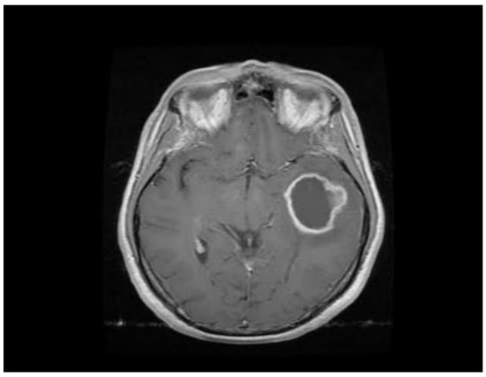
Confusion Matrix: The confusion matrix provides a tabular representation of the model's predictions versus the actual labels (tumour or NO tumour). It helps us understand how well the model is distinguishing between the two classes, showing the



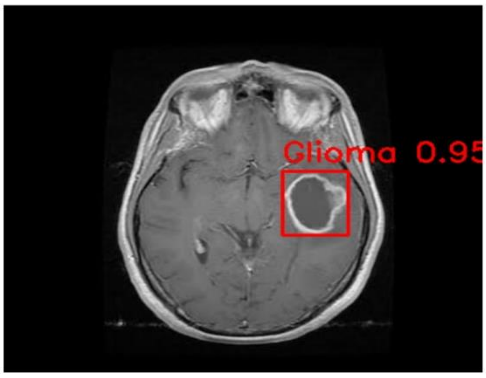
Counts of true positives, false positives, true negatives, and false negatives.

Image Prediction Accuracy: The accuracy of image predictions gives an overall indication of how well the model is classifying images correctly. It is a crucial metric to assess the model's overall performance on the test dataset.

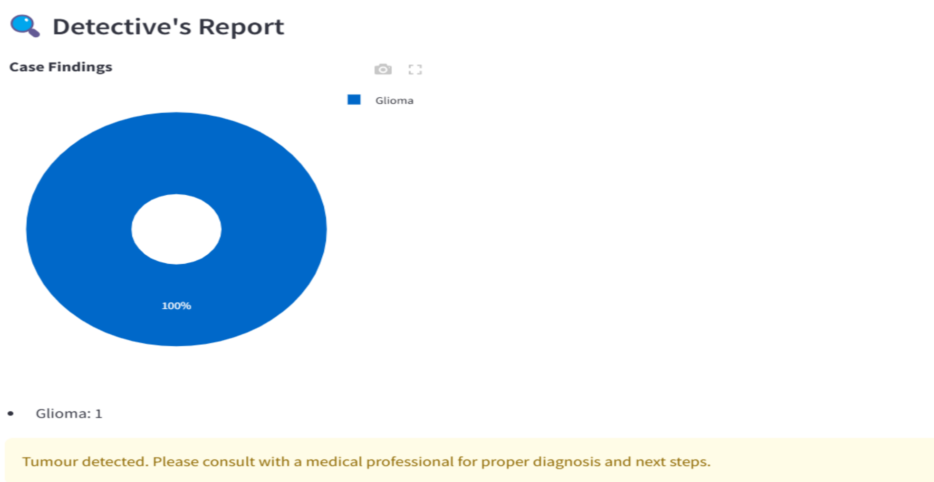
Output:



Uploaded MRI Scan



Detective AI Findings



Classification Report: The classification report summarises various classification metrics such as precision, recall, F1-score, and support for each class (AI and REAL). It provides insights into the model's performance for each class, including its ability to correctly classify tumour images.

```
y_pred = model.predict(X_test, batch_size=64, verbose=1)
y_pred_bool = np.argmax(y_pred, axis=1)
print(classification_report(Y_test, y_pred_bool))

7/7 ━━━━━━━━━━━━━━━━━━━━ 1s 217ms/step
              precision    recall  f1-score   support

           0       0.95      0.21      0.34       100
           1       0.71      1.00      0.83       115
           2       0.70      1.00      0.82       105
           3       0.95      0.76      0.84        74

    accuracy                           0.75       394
   macro avg       0.83      0.74      0.71       394
weighted avg       0.81      0.75      0.71       394
```

Figure 8: Classification Report

These analyses collectively offer insights into the model's performance, highlighting areas where it excels and areas where it may need improvement. They guide further iterations of model development and fine-tuning to enhance its effectiveness for real-world applications.

XI. CONCLUSION

In this project, I have automated the diagnosis procedure for brain tumour detection by the use of image processing. Apart from several existing brain tumour segmentation and detection methodologies present for MRI of brain images, the project has proved to provide an average accuracy of up to 97%. All the steps for detecting brain tumours that have been discussed, starting from MRI image acquisition, and preprocessing steps to successfully classification the tumour using the two segmentation techniques, have been done. Preprocessing involves operations like wavelet-based methods and has been discussed. Quality enhancement and filtering are important because edge sharpening, enhancement, noise removal, and undesirable background removal improve the image quality as well as the detection procedure. Among the different filtering techniques, classification-based segmentation segments tumours accurately and manufactures sensible results for a big information set; however, undesirable behaviours can occur in cases where a category is unrepresented in training data. Despite several types of problems, these classification methods can first detect whether there is a tumour or not, and if it is there, then they can determine whether the tumour is benign or malignant.

XII. FUTURE WORK

The future scope of the project involves exploring advanced neural network architectures, leveraging transfer learning, and fine-tuning models for MRI image detection. Real-time deployment optimisation, ensemble methods, continuous evaluation, and ethical considerations are paramount for responsible application, innovation, and positive societal impact.

XIII. REFERENCES

[1] J. Seetha and S. Selvakumar Raja," Brain Tumour Classification Using Convolutional Neural Networks," in Biomedical Pharmacology Journal, September 2018, Vol. 11(3), p. 1457-1461. http://biomedpharmajournal.org/vol11no3/brain-tumour-classificationusing-convolutional-neural-networks/

[2] Ali Ari and Davyut Hanbay, "Deep learning-based brain tumour classification and detection system," in Turkish Journal of Electrical Engineering Computer Sciences, 2018, 26: 2275–2286. https://journals.tubitak.gov.tr/elektrik/abstract.htm?id=23203

[3] Hossam H. Sultan, Nancy M. Salem, and Walid Al-Albany, "MultiClassification of Brain Tumour Images Using Deep Neural Network," IEEE, 2019.https://ieeexplore.ieee.org/abstract/document/8723045

[4] Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun, "Deep Residual Learning for Image Recognition," https://arxiv.org/pdf/1512.03385.pdf

[5] https://braintumour.org/brain-tumour-information/brain-tumour-facts/#quick-facts

[6] Sambasivarao K. "Region of Interest Pooling," https://towardsdatascience.com/region-of-interest-pooling-f7c637f409af

[7] Karan Chauhan1, Shrwan Ram2, "Image Classification with Deep Learning and Comparison between Different Convolutional Neural Network Structures using Tensorflow and Keras," International Journal of Advance Engineering and Research Development. (Volume 5, Issue 02, February 2018)

[8] Himanshu Rawlani, "Visual Interpretability for Convolutional Neural Networks" https://towardsdatascience.com/visual-interpretability-for-convolutional- neural-networks-2453856210ce

[9] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition" https://arxiv.org/pdf/1409.1556.pdf

[10] Sentdex: ”Machine Learning with Python” https://www.youtube.com/user/sentdex
[11] Kaggle - Data set Brain tumour classification (MRI) | Kaggle
[12] Karl Weiss, Taghi M. Khoshgoftaar, DingDing Wang, ”A survey of transfer learning” A survey of transfer learning | Journal of Big Data | Full Text (springeropen.com)