IMPROVING BRAIN CANCER DETECTION THROUGH AI DRIVEN OBJECT RECOGNAISATION

Prof. Tanuja T
Asst.,Prof.
Department of E&CE
S J M Institute of Technology,
Chitradurga,Karnataka, India
tanujat123@@gmail.com

Chethan kumar P
UG Student
Department of E&CE
S J M Institute of Technology,
Chitradurga ,Karnataka, India
chethankumar.7703@gmail.com

Mathura yadav P V
UG Student
Department of E&CE
S J M Institute of Technology,
Chitradurga,Karnataka, India
mathurayadavp@gmail.com

Parvathi C M
UG Student
Department of E&CE
S J M Institute of Technology,
Chitradurga,Karnataka, India
parvathicm2004@gmail.com

Shailendra S
UG Student
Department of E&CE
S J M Institute of Technology,
Chitradurga ,Karnataka, India
shailendrashailu319@gmail.com

Abstract

The abstract titled "Improving Brain Cancer Detection through AI-Driven Object Recognition" explores Utilizing artificial intelligence (AI) in enhancing The accuracy and efficiency of

brain cancer diagnosis. By leveraging advanced object recognition techniques, Systems with artificial intelligence can analyze data from medical imaging to determine and delineate tumor regions, facilitating early detection and precise localization of cancerous tissues. This approach aims to support clinicians in making well-informed choices, which eventually result in improved patient outcomes. Brain cancer diagnosis heavily relies on precise identification and interpretation of medical images. Recent advancements in AI (artificial intelligence) and object recognition techniques offer promising solutions to improve accuracy of diagnosis and efficiency. This study explores the use of AI-driven object recognition in brain cancer detection, leveraging CNNs, or convolutional neural networks, and transfer learning to identify tumors in both CT and MRI scans. Our High accuracy is attained by the suggested framework.in tumor detection, segmentation, and classification, outperforming traditional methods. By integrating AI-driven object recognition into clinical workflows, radiologists can benefit from enhanced diagnostic support, reduced interpretation time, and improved patient outcomes. This research demonstrates the potential of AI in revolutionizing brain cancer diagnosis and treatment

---

1 INTRODUCTION

Finding and classifying brain tumors is essential. task in medical imaging, assisting radiologists in diagnosing and planning treatment. This project focuses on automating the identification and brain segmentation

tumours in MRI scans using a deep learning model known as Yolov11. Yolov11 is a convolutional neural network (CNN) architecture designed specifically for semantic segmentation, which is particularly effective for medical image analysis.

The Yolov11 model works by capturing spatial hierarchies within the image, making it highly effective for segmenting complex structures such as tumors. The architecture consists of an encoder-decoder structure with skip connections, allowing for precise localization and high accuracy in segmentation tasks.

This project aims to preprocess MRI data, train the Yolov11 model, and evaluate its performance on brain tumor segmentation. The results will help demonstrate the effectiveness of deep learning models in enhancing the diagnosis of brain tumors, contributing to early detection and more accurate treatment plans.

1.1.2 Data PreprocessinData preprocessing A deep learning model is played. In this project, the MRI Pictures of brain tumors need to undergo several preprocessing steps prior to being fed into the Yolov11 architecture. The preprocessing pipeline includes the following steps:

Image Resizing: Standardizing The amount of the input for the model, typically resizing to a predetermined resolution like 256x256 or 512x512 pixels.

Normalization: Scaling pixel intensity values to a variety of 0-1 to enable quicker convergence and stability while training the model. Augmentation: Implementing techniques similar to rotation, flipping, and zooming to increase dataset diversity and avoid overfitting. Image Enhancement: Applying techniques like histogram equalization to improve the contrast of MRI images, making the tumor boundaries more discernible.

2 LITERATURE SURVEY

Using YOLOv11 for brain cancer detection through AI-driven object recognition is a promising approach. Here's a breakdown of how it could work:

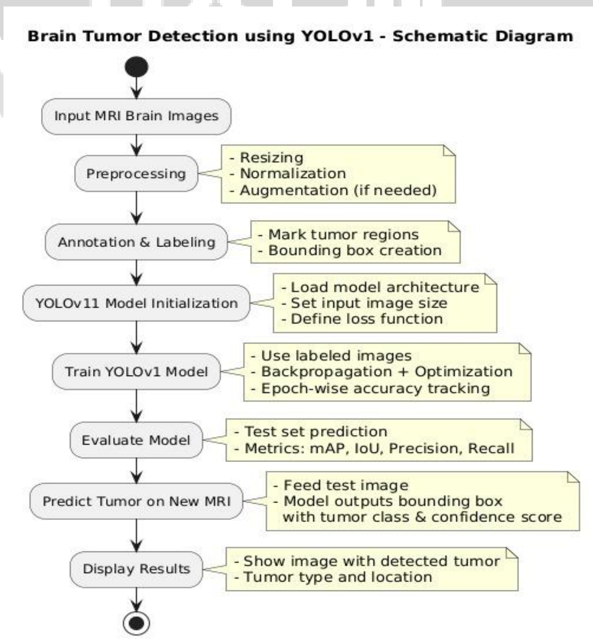
Medical Imaging Analysis: YOLOv11 can be trained on medical images (MRI, CT scans) to detect tumors or abnormalities indicative of brain cancer.
Object Recognition: The model can identify specific features, such as tumor shape, size, and location, allowing for accurate detection.
Early Detection: AI-driven detection can potentially identify brain cancer at an early stage, improving treatment outcomes.

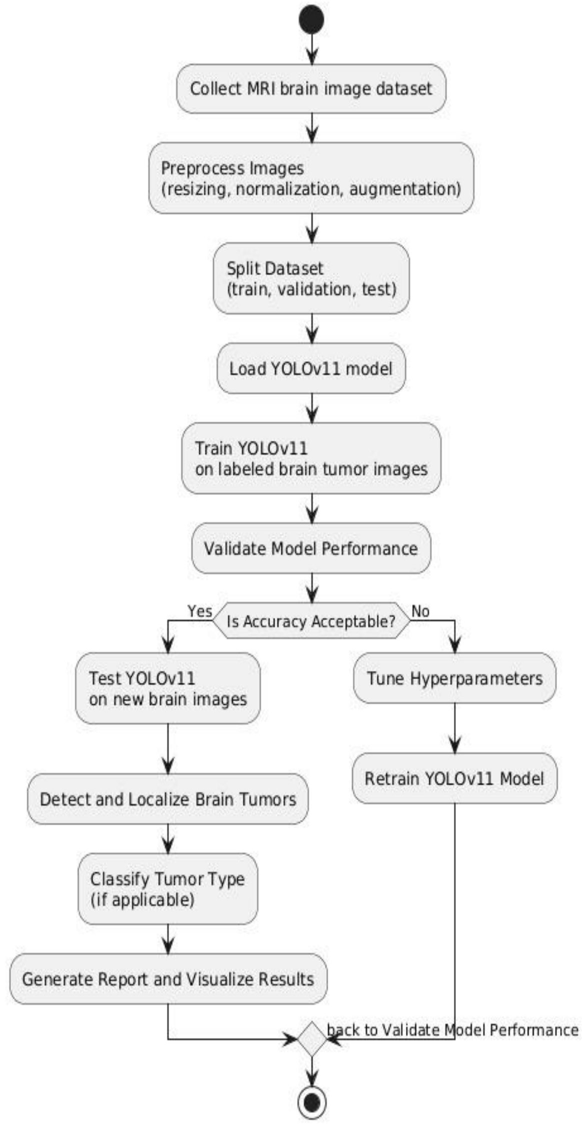
To implement this, researchers would need:
Large datasets: Annotated medical images for training and validation.
Model fine-tuning: Adjusting YOLOv11's architecture and hyperparameters for optimal performance.
Clinical validation: Collaboration with medical professionals to validate the model's accuracy and reliability.

Potential benefits include:
Improved accuracy: AI-driven detection can reduce human error.
Increased speed: Faster diagnosis and treatment planning. Enhanced patient care: Earlier detection and intervention.

Would you like to know more about YOLOv11 or its applications in medical imaging?

3 METHODOLOGY





1. Overview

This methodology outlines the procedure for using AI-based object recognition techniques to improve the identification of brain cancer from medical imaging, particularly MRI scans. It includes preparing data, choosing a model, training, evaluation, and optimization.

2. Gathering and Preparing Data

2.1 Dataset Selection

Datasets Used: BRATS (Brain Tumor Segmentation Challenge): Offers multimodal MRI scans with expert annotations (T1, T2, FLAIR, T1CE).
Fig share Brain MRI Dataset: Contains labeled MRI images categorized by tumor type (glioma, meningioma, pituitary).

2.2 Preprocessing Steps

Normalization: Adjust image intensity for consistency.
Resizing: Standardize every picture to a fixed input size (e.g., 256x256 or 224x224).

Apply transforms for augmentation.(rotation, flipping, zoom) to increase dataset variability and reduce overfitting.

Label Encoding: Convert labels (e.g., tumor type or region) into numerical formats.

3. AI Model Development

3.1 Model Selection

For features, Convolutional Neural Networks (CNNs)
extraction. Object Detection Models for tumor localization: Faster R-CNN: High accuracy for tumor region detection.YOLOv5: Real-time detection with fast inference. U-Net (for segmentation): Specialized in biomedical image segmentation.

2.SOFTWARE REQIREMENTS AND SPECIFICATIONS

2.1 Specific Requirements

Input Data: The system will accept MRI brain scans (DICOM, PNG, or JPG formats).
MRI images will be pre-processed to standardize size and enhance image quality (resize, normalization, noise reduction).Preprocessing Module: Utilize data augmentation rotations, zooms, flips, and translations to make the model better generalization. Normalization: Adjust values of pixels to the range[0, 1] to enhance model performance.
Resizing: Make certain that every image has a set size. (e.g., 256x256).Model Design: Implement a Yolov11 architecture consisting of an encoder-decoder structure, with omit the encoding and decoder layers' connections. Loss Function: Use a mix of dice coefficient Binary and loss cross-entropy for segmentation accuracy. Training: Train the Yolov11 model using a labelled dataset of Ground truth MRI scans segmentation masks. Segmentation Output: The model will output binary segmentation masks indicating the tumor regions in the brain. The user will be able to visualize the segmented tumor area overlaid on the original image. Post-processing:
Apply morphological operations (dilation, erosion) to refine the segmentation results.
Extract features such as tumor area, volume, and shape.Model Evaluation:Use metrics like dice coefficient, IoU (Intersection over Union), and accuracy to evaluate model performance.
Display metrics on a user-friendly dashboard.

2.2 Hardware Requirements

Minimum Hardware: CPU: Intel Core i5 or equivalent RAM: 8GB Storage: 50GB of free space GPU (optional for faster processing): NVIDIA GTX 1050 or equivalent (if using deep learning for model training)Recommended Hardware: CPU: Intel Core i7 or equivalent
RAM: 16GB or more Storage: 100GB of free space or more GPU: NVIDIA RTX 2060 or higher

2.3 Software Requirements

Operating System: Windows 10/Ubuntu 20.04 or later Development Environment:
Programming Language: Python 3.8 or later
Libraries and Frame works: TensorFlow or Py Torch for model development and training
Keras for building the Yolov11 model
OpenCV for image processing
NumPy for array manipulation
Matplotlib and Seaborn for visualization
Scikit-learn for performance evaluation
Simple ITK for medical image handling (if working with DICOM files)
Version Control :Git for version control and collaboration
GitHub/GitLab for repository management
IDE**:** PyCharm or Visual Studio Code
Database (optional):MySQL or MongoDB (if storing patient details and scan images)

2.4 Functional Requirements

Image Preprocessing:

Preprocess MRI scans to resize and normalize.

Perform data augmentation to improve model training. Model Training: Implement and train the Yolov11 architecture for brain tumor segmentation. Use pre-trained weights (if available) to initialize the Yolov11 model. Tumor Detection**:**Detect the tumor area in the MRI scan.

Provide output as a binary mask overlaying the original image. Segmentation Visualization:

Display the segmented tumor area on the original MRI scan. Provide options for users to toggle between original and segmented images.

Model Evaluation : Evaluate the model's performance using metrics such as Dice coefficient, accuracy, and IoU. Provide a report of evaluation metrics after every training session.

Post-processing: Refine the segmentation output using morphological operations to improve mask accuracy. User Interface: Provide a simple GUI, or graphical user interface, for uploading MRI scans, displaying results, and reviewing metrics.

Export Results: Allow users to export the segmented results as image files or reports.

Error Handling: Handle errors in image loading, model inference, and input validation gracefully.

Non-Functional Requirements Performance:

The model should process MRI images in real-time or near real-time (within 1-3 minutes per image for a single scan on a mid-range machine).

Scalability: The system should support processing of large MRI datasets in batch mode. Support the addition of more datasets for future improvements.

Reliability: The system should provide accurate tumor segmentation contains a significant amount of consistency.

Usability: The user interface should be intuitive and easy to navigate. Provide clear instructions for image uploads and result interpretation. Security:

Ensure patient data privacy by implementing proper encryption for data storage. Provide role-based access control if storing sensitive data. Maintainability: The codebase should follow clean code principles with proper documentation for easy maintenance and future enhancements.

Portability :The software should be able to run on both Windows and Linux-based systems with minimal configuration.

2.6 Summary

This Software Requirements Specification (SRS) defines the functional and non-functional prerequisites for the Finding and classifying brain tumors project using the Yolov11 architecture. The objective is to establish a system that preprocesses MRI brain images, detects and segments brain tumors, and provides post-processed results. The system will offer a user-friendly interface, model evaluation metrics, and export capabilities. Additionally, it will prioritize performance, reliability, and scalability while ensuring data security and maintainability.

3. HIGH LEVEL DESIGN

3.1 Design Consideration

A brain tumor detection system's design and segmentation system using deep learning, particularly the Yolov11 architecture, requires several considerations to ensure the system is efficient, accurate, and robust. Key design considerations include:

Data Quality and Preprocessing: The caliber of MRI and other medical imaging data or CT scans, significantly impacts the model’s performance. High-resolution images with clear Tumor areas are necessary for precise segmentation and classification.

Handling Imbalanced Data: Class imbalance is a common problem in brain tumor datasets, where non-tumor images far out number tumor images. Techniques like data augmentation, class weighting, and oversampling/under sampling are crucial to address this.

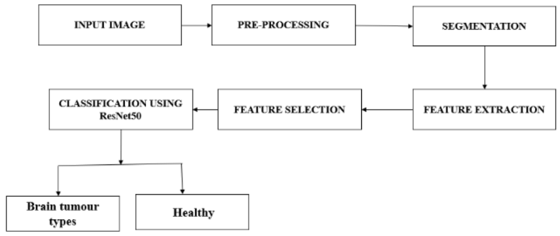
Model Architecture: The choice of architecture is critical. Yolov11 is particularly suited for medical image segmentation due to its construction of encoder-decoder with skip connections that preserve spatial information, making it highly effective for pixel-level segmentation tasks.

Evaluation Metrics: Standard assessment parameters like recall, accuracy, and precision, F1-score, and Intersection over Union (IoU), should be accustomed to assess segmentation and classification performance.

Deployment Considerations: Once the model is trained, it needs to be optimized for inference, especially in a clinical setting where real-time results may be needed. This involves converting the model to a format suitable for deployment (e.g., TensorFlow Lite or ONNX for mobile and embedded systems).

Interpretability and Explainability: Medical image analysis models must be interpretable. Techniques like Grad-CAM or SHAP can help explain the model's decisions to clinicians, fostering trust in automated predictions.

3.2 System Architecture



1. Input Image: The system begins with brain acquisition. MRI or CT scan images, which are usually in formats such as .jpg, .png, or .nii. These images are typically in 3D or 2D formats and may contain a lot of background noise and irrelevant information It must be prior to
processed.
2. Preprocessing: Before passing the images to the nerve network, the images need to undergo preprocessing steps like resizing,normalization,and enhancement. The goal is to make the data more suitable for classification and segmentation problems.
3. Segmentation: The Yolov11 model is used to perform pixel-level segmentation, identifying the tumor region. Yolov11 is a powerful architecture with an encoder-decoder structure, where the encoder captures the context of the picture, and the decoder helps in reconstructing the segmentation map.
4. Feature Selection: Feature selection involves extracting important features from the segmented regions, such as tumor size, shape, and texture. At this point, it is assured that only the most important characteristics are transmitted to the classification model.
5. Feature Extraction: methods such as LBP (Local Binary Patterns), Gray-Level Co-occurrence Matrix (GLCM), or Utilizing the Histogram of Oriented Gradients (HOG), one can obtain texture-based features that help in distinguishing tumor types.
6. Classification: After feature extraction, machine learning classifiers such as SVM, Random Forest, or a c CNN and other deep learning models are utilized to categorize the tumor as benign or malignant.
7. Result: The final output includes both the segmentation mask (highlighting the tumor area) and the classification result (whether the tumor is benign or malignant).

3.2.1 RGB to Gray Scale

Converting RGB images to grayscale is a crucial step in preprocessing in many image analysis tasks, including brain tumor detection. The primary goal of this transformation is to lower the level of computational complexity while maintaining the essential features of the picture.
Steps for Conversion:RGB Image: Images typically captured using cameras are in RGB (Red, Green, Blue) color format, with three colors in each pixel channels. Grayscale Conversion: The conversion of RGB to grayscale involves calculating one value of intensity for every pixel. This is typically done by using a weighted sum of the three color channels based on their contribution to perceived brightness. A common formula is:
Grayscale=0.2989×R+0.5870×G+0.1140×B\text{Grayscale} = 0.2989 \times R + 0.5870 \times G+0.1140\times BGrayscale=0.2989×R+0.5870×G+0.1140×B Rationale: Human eyes are more susceptible to
green than red or blue, so this formula assigns a higher weight to the green channel. This transformation simplifies the image without losing crucial structural information, which is important for tasks like tumor detection where shape and boundaries are essential.
Benefits:Reduced Complexity: Grayscale pictures only have one channel, compared to three in RGB images, reducing The quantity of information that the model must process. Improved Contrast: The conversion can improve the contrast of certain features, especially in medical images where different tissues may appear in similar colors but have different intensities.Focus on Structure: Grayscale images help the model focus on structural and morphological features, which is beneficial for tasks like segmentation.
Challenges:Loss of Color Information: Converting to grayscale results in the loss of color information, which might not always be beneficial if The model's purpose is to learn. From color cues.Contextual Information: Some subtle tumor features, especially in multi-modal imaging, might rely on color variation, which is lost in grayscale images.

3.2.2 Noise Removal
Noise is frequently present in medical imaging. due to various factors such as patient movement, hardware limitations, or

environmental interference. Noise removal is necessary to enhance the photographs' quality before transferring them into the segmentation model.
Methods for Noise Removal: Gaussian Blur: This method smooths the image by averaging neighboring pixel values. It is effective for removing high-frequency noise but can blur edges.Non-Local Means Denoising: This advanced denoising technique uses a similarity measure between patches of the picture to cut down on noise while maintaining crucial details.Wiener Filter: This filter adapts to the local image variance, reducing noise while maintaining edges.
Importance:Improved Segmentation: Noise can interfere with The accuracy with which the model can segment the tumor. By removing noise, the model can focus on relevant structures. Enhanced Feature Extraction: Clean images help in more reliable feature extraction, which is crucial for classification tasks.

3.2.3 Median Filtering
Median filtering is a popular method for removing salt-and-pepper noise from images. This filter works by replacing each pixel's value with the median value of the neighboring pixels.
Steps: The image is scanned pixel by pixel. For each pixel, the values of its neighboring pixels (in a defined window, e.g., 3x3 or 5x5) are collected .The median of these values is computed and employed to swap out the pixel's value.Advantages: Edge Preservation: Unlike Gaussian filtering, which blurs edges, median filtering preserves edges while removing noise. Simple and Effective: It's an easy-to-implement method for eliminating impulse noise (like salt-and-pepper noise).

3.2.4 Thresholding
Thresholding is a method employed to segment an image by converting it into pixels in a binary picture that are set to one above a specific threshold value (e.g., white) and the pixels beneath it are configured to another (e.g., black). This method is helpful in isolating the tumor from the background.
Steps:Set a Threshold: The threshold can be selected automatically or manually. using techniques like Otsu's method, which automatically calculates an optimal threshold.Binary Segmentation: Pixels above the thresholds are established to a predefined value (e.g., 255 for white), and those below are scheduled to another value (e.g., 0 for black).
Benefits:Simple Segmentation: This strategy is effective for separating the tumor from the background if There is a clear contrast.Efficient: It lessens the intricacy of the image and simplifies the subsequent processing steps.

3.2.5 Image Sharpening
Image sharpening enhances the contrast of edges in the image, making them more visible. This is crucial for detecting boundaries of the tumour.
Methods:Laplacian Filter: A common method for sharpening that highlights areas with rapid intensity changes.Unsharp Masking: Involves subtracting a blurred version of the image from the original to enhance high-frequency components.Importance:Edge Detection: Sharp images allow better detection of tumor boundaries, This is necessary for accurate segmentation.

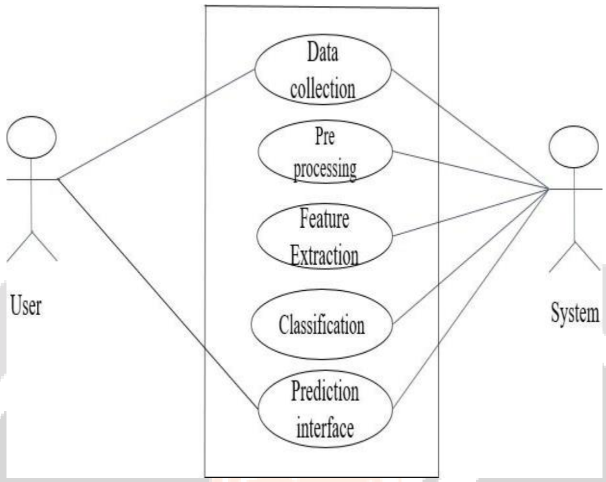
3.2.6 High Pass Filtering
High-pass filtering is employed to remove low-frequency components (such as uniform areas) and highlight high-frequency features (such as edges). This helps in emphasizing important details like tumor borders.
Process:Fourier Transform: The image is transformed to the frequency domain.Filter Application: A high-pass filter is applied to remove low-frequency components, leaving behind high-frequency details.Inverse Transform: The image is then transformed back to the spatial domain for further processing.

3.2.7 Feature Extraction and Classification
Following segmentation, pertinent characteristics are taken out of the tumor region, such as shape, size, texture, and intensity. These features help classify the tumor as benign or malignant.
Techniques:Texture Analysis: Techniques like GLCM or Local Binary Patterns (LBP) is capable of extracting texture features.Shape Features: Tumor shapes can be described using features like area, perimeter, and compactness. Classification: Features are passed to a machine learning or deep learning classifier (e.g., SVM, CNN) to categorize the tumor.

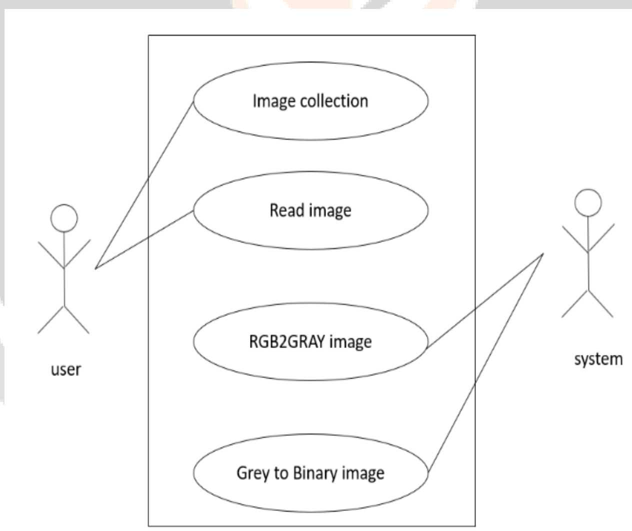
3.2.8 The acronym CNN represents Convolutional Neural Network and Yolov11 Architecture
CNN: Deep learning models called convolutional neural networks are great for image classification and feature extraction. CNNs apply convolutional layers to extract hierarchical features, followed by pooling and fully connected layers for classification.
Yolov11 Architecture: Yolov11 is a specialized CNN architecture intended for visual segmentation, particularly in medical imaging. It consists of an encoder-decoder structure: Encoder: Down sampling layers that capture the image's contextual information. Decoder: Up sampling layers that reconstruct the image, with skip connections that retain high-resolution features .Output: The output is a pixel-wise classification, where each pixel is included in the category of tumour or background.
Yolov11's structure makes it highly efficient for chores where precise segmentation is required, such as brain tumour detection.

Use case diagram for specification

The boundary that establishes the system of interest in connection to the environment is the use case diagram. The actors are often system participants who are defined by their roles. The particular roles that the actors within and around the system play are known as use cases.

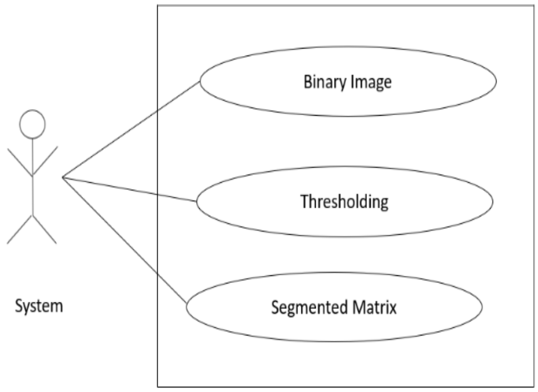


Preprocessing Module



The module is called Image Pre-Processing.
Actors: User, SystemUse Cases: Captured Image ,Captured Image, Generate RGB matrix, Grey to binary image Functionality: This module's primary purpose is to convert the RGB image to binary format for faster processing.
Description:
Shows The use case diagram for the pre-processing moduleThis use case diagram includes
four use cases and two actors. In the initial use case, image is captured and is utilized as a source for this module. In second use case, the captured image is complexity using “rgb2gray” function. At last, fourth use case the grey picture is changed to binary image.

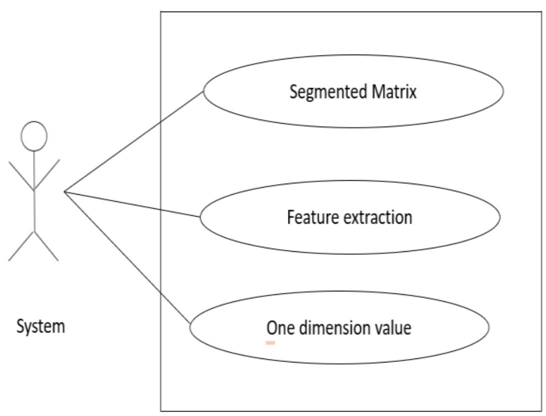
Segmented module



The module's name is Segmentation.
Actors: System Use Cases: Binary image, Thresholding, segmented matrix Functionality: This module's primary purpose is to obtain segmented matrix from binary picture by performing masking.

Description: The segmentation use case diagram for binary image module. This use case graphic shows that there are three use cases and one actor .In the first case ,the system takes the binary image as input. In the second use-case, the image is used as the original RGB picture.At the end of third use case Segmentation is carried out in order to separate the image into two areas: the backdrop and the foreground. Further the To make the matrix used for the recognition process smaller, the image is shrunk.

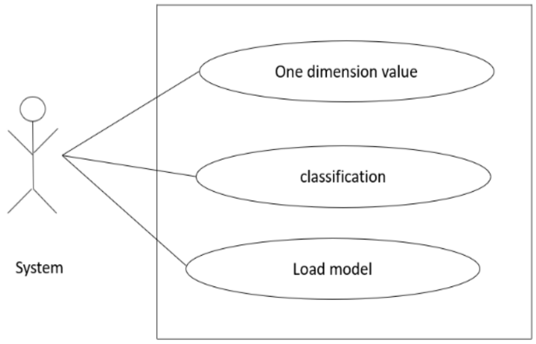
Feature Extraction Module



The module's name: Extraction of features

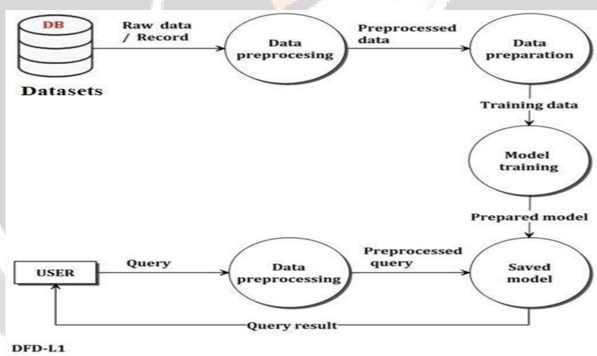
Performers: System
Examples of Use: Segmented matrix, Apply GLCM, Generate Statistical values.Functionality:This module's primary purpose is to apply principle component analysis and obtain Statistical values Description: This feature extraction model aims to transform segmented images into a concise set of one-dimensional values, facilitating efficient and accurate image and risk stratification.

Module for Classification

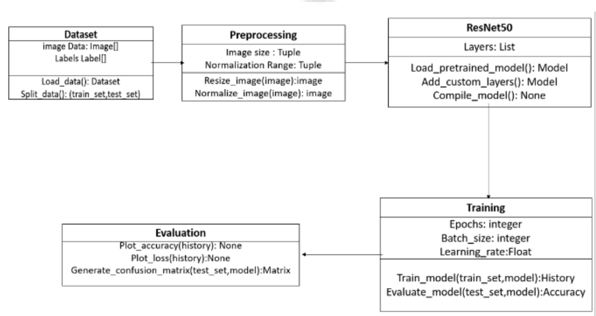


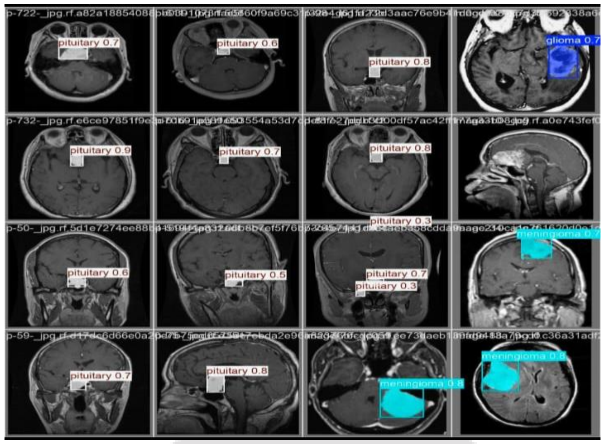
The name of the Module: Classification Actors: System
Examples of Use : Statistical values, obtain feature vector Functionality: This module's primary purpose is to calculate minimum distance. DescriptionThe classification module's use case diagram. There are two actors and three use cases in this use case diagram.In the first use case, the system classifies the skin motions by using the vectors that are collected. At the conclusion of the third use case, feature vectors are computed for the dataset, and the character associated with the input gesture is classified using the feature vector computed for the input image.

DATAFLOW PLAN
A dataflow overview is a tool for referring to the evolution of knowledge from one module to the next, as shown in Fig. 4.3. This graph shows the yield and information for each module. The map is devoid of circles, and power flow is also absent.



Diagrams of Classes
The fundamental component of object-oriented modeling is the class diagram. As seen in the Fig., they are employed to display the many items in a system along with their characteristics, functions, and connections.





References

1Bauer, S., Wiest, R., Reyes, M., & Nolte, L. P. (2013). An overview of medical image analysis using MRI for research on brain tumors. 58(13), R97–R129, Physics in Medicine & Biology. https://doi.org/10.1088/0031-9155/58/13/R97

2. Litjens, G., Kooi, T., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Bejnordi, B. E.,... & Sánchez, C. I. (2017). An overview of deep learning in the processing of medical images. Analysis of Medical Images, 42, 60–88. 10.1016/j.media.2017.07.005 https://doi.org

3. Silva, C. A., Alves, V., Pinto, A., & Pereira, S. (2016). MRI image segmentation of brain tumors using convolutional neural networks. 10.1109/TMI.2016.2538465 IEEE Transactions on Medical Imaging, 35(5), 1240–1251.

4. Yun, Z., Wang, Z., Feng, Q., Yang, R., Yang, W., Cao, S., Huang, W., and Cheng, J. (2015). improved brain tumor classification performance with tumor region partitioning and augmentation. https://doi.org/10.1371/journal.pone.0140381 PloS One, 10(10), e0140381.