Brain Tumor Detector

Sumaiya Khan (202480995), Mirza Abdullah Al Nafiz (202380541)
ENGI 9804 Industrial Machine Vision
Memorial University of Newfoundland
St. John's, NL

***Abstract-* This project presents a rule-based brain tumor detection system developed in MATLAB using classical image processing techniques. The system is designed as an accessible diagnostic tool for low-resource settings where deep learning solutions are impractical. The pipeline involves median filtering, Sobel edge detection, binary thresholding, connected component analysis, and morphological dilation. Results show successful tumor detection in most cases, but also expose failure in one test case, highlighting system limitations. This report outlines the motivation, implementation, results, and areas for future improvement.**

Index Terms— Brain Tumor Detection, Image Processing, MATLAB, Sobel Edge Detection, Morphological Operations

I. INTRODUCTION

Brain tumors are a serious health concern and among the leading causes of cancer-related deaths. Early and accurate detection is essential for improving treatment outcomes and survival rates. While MRI is the preferred imaging method for identifying tumors, manual interpretation by radiologists is time-consuming and prone to human error.

Automated tumor detection systems can help address these challenges by offering fast, consistent, and objective analysis. Although deep learning methods have shown strong performance, they require large datasets and significant computational resources—making them less practical in low-resource settings.

This project presents a lightweight, rule-based tumor detection system built in MATLAB using classical image processing techniques. The pipeline includes median filtering, Sobel edge detection, binary thresholding, region labeling, and morphological operations. These steps offer a deterministic, interpretable, and efficient alternative to machine learning, suitable for environments with limited resources.

The report outlines the problem, existing methods, the system design and implementation, results—including a failed case—and recommendations for future improvements.

II. PROBLEM STATEMENT

Accurate detection of brain tumors is essential for effective diagnosis and treatment planning. While MRI is the standard imaging tool, manual interpretation is time-consuming, subjective, and prone to variability among radiologists—especially in low-contrast or complex cases.

Although deep learning methods offer high accuracy, they require large annotated datasets and powerful hardware, which are often unavailable in low-resource or remote settings. Additionally, the "black-box" nature of these models raises concerns about transparency in medical decision-making.

There is a clear need for a reliable, interpretable, and resource-efficient method for brain tumor detection. This project addresses that gap by developing a rule-based image processing system in MATLAB that can automate detection using classical techniques, without relying on machine learning or high-end hardware.

III. LITERATURE REVIEW

In recent years, automated brain tumor detection systems have gained attention for their ability to assist radiologists in analyzing MRI scans more efficiently. These systems generally fall into two categories: deep learning-based approaches and classical image processing techniques.

Deep learning, particularly using Convolutional Neural Networks (CNNs), has shown high accuracy in segmenting and classifying brain tumors. Models like U-Net, trained on datasets such as BRATS, have demonstrated strong performance. However, they require extensive computational resources and large labeled datasets, making them impractical for many low-resource settings.

Classical image processing methods offer a simpler, more interpretable alternative. Techniques like median filtering, Sobel edge detection, thresholding, and morphological operations have been effectively used to isolate tumor regions without any training data. Studies by Khotanlou and Rajinikanth showed promising results using such rule-based approaches, especially in settings where transparency and efficiency are prioritized.

While deep learning dominates current research, classical methods remain valuable for environments where resources are limited and interpretability is essential. This project builds on such methods to develop a lightweight and accessible tumor detection system.

IV. PROPOSED SOLUTION

To overcome the limitations of data-heavy and resource-intensive methods, this project proposes a rule-based brain tumor detection system using classical image processing techniques in MATLAB. The system is lightweight, interpretable, and ideal for low-resource environments where deep learning is impractical.

The pipeline processes MRI scans through a series of deterministic steps designed to enhance and isolate tumor regions. It begins with *median filtering* to reduce noise while preserving edges, followed by *Sobel edge detection* to highlight high-intensity transitions typical of tumor boundaries. The image is then *binarized*, simplifying segmentation by converting it to black-and-white based on pixel intensity.

Next, *connected component labeling* identifies distinct regions, and features like area and solidity are calculated to filter out irrelevant shapes. Tumors typically appear as compact, high-solidity regions. These regions are then refined using *morphological dilation* to fill gaps and smooth edges. The largest high-solidity region is selected and *overlaid on the original MRI* for visualization.

This transparent and rule-based approach provides consistent, interpretable results without requiring training data. While it lacks the adaptability of machine learning, it offers a practical and efficient alternative for tumor detection in environments with limited computational resources.

V. METHODOLOGY

The methodology of this project is centered around a rule-based image processing pipeline aimed at detecting brain tumors in MRI scans. Each stage of the pipeline is carefully selected to contribute to the accurate and efficient isolation of tumor regions, without requiring training data or advanced computational resources. The approach is fully deterministic, meaning the same steps are applied consistently to each image, ensuring repeatability and transparency.

The core steps of the methodology are as follows:

Input Acquisition

MRI brain images are collected in standard image formats (e.g., JPG, PNG). These scans typically contain multiple tissues, and the goal is to isolate and identify regions that could represent tumors. The system is built to handle both grayscale and RGB images, converting RGB to grayscale if necessary.

Preprocessing – Median Filtering

The grayscale image often contains noise, particularly salt-and-pepper noise that may affect segmentation accuracy. Median filtering is applied using a 3x3 or 5x5 window. This non-linear filtering method is effective in reducing noise while preserving the edges and structural integrity of the tumor region. It ensures that the edges remain sharp, which is critical for the next step.

Edge Detection – Sobel Operator

After denoising, edge detection is applied using the Sobel operator, which computes the gradient of image intensity. Tumors often exhibit distinct contrast from surrounding tissue, and the Sobel filter enhances these differences by identifying sharp transitions. The resulting edge map highlights the boundaries where pixel intensity changes rapidly likely tumor borders.

Binarization

To simplify the processing, the edge-detected image is converted into a binary image. A global threshold is applied to separate foreground (high-intensity regions) from background. This step prepares the image for connected component analysis by creating clear, separable regions of interest.

Connected Component Labeling

The binary image is scanned to identify and label all connected white regions (objects). Each object is then treated as a potential tumor candidate. MATLAB's bwlabel function is used here to assign unique labels to each connected region.

Feature Extraction – Area and Solidity

Each labeled region is analyzed for its geometric properties. Two key features are extracted:

1. *Area* – the number of pixels in the region.
2. *Solidity* – the ratio of the region's area to the area of its convex hull. High solidity indicates a compact, solid shape, which is a common characteristic of tumors.

Regions with very small areas or low solidity are filtered out, as they are likely to represent noise, anatomical artifacts, or non-tumor tissues.

Morphological Operations – Dilation

The remaining candidate regions are dilated using a morphological structuring element (typically a disk shape). This expands the region slightly, fills small gaps, and smooths the tumor boundary to create a more visually accurate mask. Morphological dilation enhances the interpretability of the output by approximating the true shape of the tumor.

Final Region Selection and Overlay

Among the filtered regions, the one with the largest area and acceptable solidity is selected as the tumor region. This region is then overlaid on the original MRI image, either by outlining it or masking it in a highlighted color for visual clarity. This final output can be used by medical personnel to quickly assess the presence and size of the suspected tumor.

This methodology ensures a balance between simplicity, accuracy, and computational efficiency. By leveraging the strengths of classical image processing techniques, it delivers a viable solution for brain tumor detection in contexts where more advanced AI tools may not be feasible.

VI. TOOLS & ENVIRONMENTS

The brain tumor detection system was developed using MATLAB R2023a with the Image Processing Toolbox. The implementation follows a script-based workflow and is designed to run efficiently on standard computing systems without the need for GPU acceleration.

Development Tools and Environment

- *Platform*: MATLAB R2023a
- *Image Processing Toolbox*: Used for filtering, edge detection, labeling, and morphology.
- *Hardware Requirements*: Standard PC or laptop; no GPU needed.
- *User Interface*: A basic script-driven workflow with image display after each stage.

This environment ensures the system remains lightweight, accessible, and easy to maintain; making it suitable for low-resource settings and educational use.

VII. IMPLEMENTATION

The workflow begins by loading the MRI image using *imread()* and converting it to grayscale using *rgb2gray()* if necessary. Noise is reduced using *medfilt2()*, which applies a median filter to preserve important structural details. Edge detection is performed using the Sobel operator via the *edge()* function to highlight intensity transitions, which are often indicative of tumor boundaries.

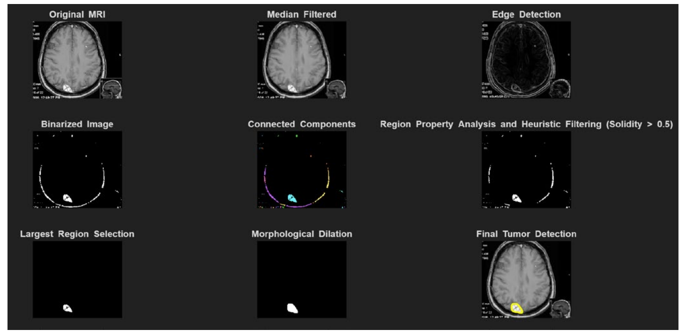
The edge map is then binarized with *imbinarize()* to create a black-and-white mask. Connected components are labeled using *bwlabel()*, and geometric features such as area and solidity are extracted with *regionprops()*. Regions with low solidity or very small area are filtered out to eliminate false positives. The most prominent region is selected and expanded using *imdilate()* to form a smooth and continuous tumor mask.

The final tumor region is overlaid on the original MRI scan for visualization, and intermediate outputs are saved to provide transparency and aid in evaluation.

VIII. TESTING AND RESULTS

The system was tested on a set of MRI brain scans with varying levels of clarity and contrast. In most cases, the detection pipeline performed well—successfully identifying and highlighting tumor regions. The intermediate outputs, including the edge maps and segmented masks, showed that each step contributed effectively to isolating the tumor.

In a clear example, the tumor appeared as a distinct, dense region that passed the area and solidity thresholds. Morphological dilation further refined the output by closing small gaps, resulting in a clear and accurate tumor boundary.



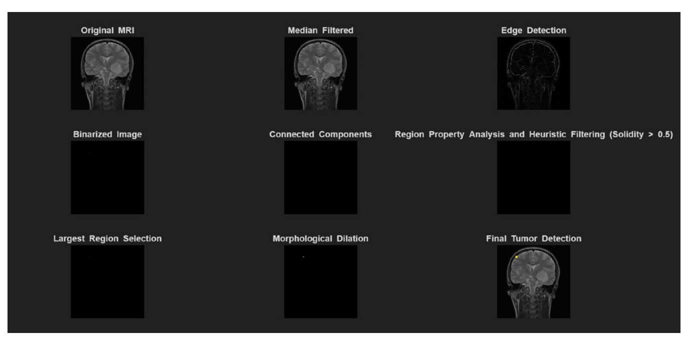
However, in one case, the system failed to detect a tumor due to poor contrast and weak edges. The region was either missed during edge detection or filtered out due to not meeting the solidity criteria. This revealed a limitation of the fixed-rule approach when dealing with low-quality or ambiguous images.

Despite this, the system demonstrated reliable performance in most scenarios and produced interpretable, step-by-step visual outputs to support its results.

IX. FAILED CASE AND ERROR ANALYSIS

In one test case, the system failed to detect a tumor due to poor contrast and unclear boundaries in the MRI scan. The Sobel edge detector could not highlight meaningful edges, and the binarization step produced a fragmented mask. As a result, no region passed the area and solidity filters, and the tumor was missed entirely.

This case highlights a key limitation of rule-based methods—they depend heavily on clear contrast and shape regularity. When those conditions are absent, detection fails. The use of global thresholding also contributed to the issue, suggesting that adaptive thresholding or contrast enhancement could improve robustness in such challenging scenarios.



Limitations by Processing Stage

Original MRI: Tumor region may not have strong contrast compared to surrounding tissue.

Median Filtering: Reduces noise but does not enhance low-contrast tumor regions.

Edge Detection (Sobel): Fails when tumor boundaries have weak or gradual intensity transitions.

Binarized Image: Global thresholding misses tumor regions with subtle intensity differences.

Connected Components: No significant regions remain if thresholding eliminates useful structures.

Region Property Analysis and Filtering: Low-solidity or irregularly shaped tumors may be excluded as false positives.

Largest Region Selection: No region is selected if earlier steps fail to isolate any valid component.

Morphological Dilation: Ineffective if no candidate region is available from prior filtering.

Final Tumor Detection: Fails silently with no visual output if all upstream detections fail.

X. Conclusion

This project successfully developed a rule-based image processing system for detecting brain tumors in MRI scans using MATLAB. The approach, based on classical techniques like filtering, edge detection, and morphological operations, produced accurate results in most cases while remaining lightweight and interpretable.

The system's strength lies in its simplicity and transparency, making it suitable for low-resource settings. However, a failed test case revealed limitations when handling low-contrast or irregular images, showing that fixed thresholds may not always be reliable.

Overall, the project met its objective of creating an efficient and accessible tumor detection tool, offering a foundation for future improvements.

XI. Future Work

While the current system shows promising results, several improvements can enhance its accuracy and robustness. One key area for future development is the introduction of *adaptive thresholding* techniques, such as Otsu's method or local thresholding, which can better handle variations in image brightness and contrast.

Incorporating *contrast enhancement* or histogram equalization during preprocessing could also improve the visibility of tumor regions in low-quality scans. Additionally, instead of relying solely on geometric features like area and solidity, future versions could explore *texture-based analysis* or shape descriptors to better distinguish tumors from similar-looking structures.

Another promising direction is to integrate *machine learning or hybrid approaches*, where classical techniques handle initial segmentation, and a lightweight classifier refines the results. This would improve accuracy without requiring large datasets or complex training pipelines.

Lastly, expanding the dataset for broader testing and integrating a simple *graphical user interface (GUI)* would enhance usability and support deployment in clinical or educational settings.

XII. References

[1] R. Gonzalez and R. Woods, 'Digital Image Processing', 4th ed., Pearson, 2018.

[2] M. Islam, et al., 'Brain Tumor Detection Using Traditional Image Processing Techniques', IJACSA, 2022.

[3] S. Bauer, R. Wiest, L.-P. Nolte, and M. Reyes, "A survey of MRI-based medical image analysis for brain tumor studies," *Physics in Medicine & Biology*, vol. 58, no. 13, pp. 97–129, Jul. 2013.

[4] M. Havaei et al., "Brain tumor segmentation with deep neural networks," *Medical Image Analysis*, vol. 35, pp. 18–31, Jan. 2017.

[5] P. Rajinikanth, G. D. D. J. Kumar, and R. Subramanian, "Automatic brain tumor segmentation using region labeling and morphological operations," *Procedia Computer Science*, vol. 45, pp. 262–271, 2015.