

ROI-Guided Brain MRI Segmentation and Tumor-Region Visualization Using MATLAB

Nafisa Ibnat Tasfee

nafisatasfee@gmail.com

The University of Texas at Arlington

Research Article

Keywords: Brain MRI, tumor-region segmentation, region of interest, binomial smoothing, thresholding, MATLAB, graphical user interface.

Posted Date: October 29th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11532921/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

ROI-Guided Brain MRI Segmentation and Tumor-Region Visualization Using MATLAB

Nafisa Ibbat Tasfee

Department of Mechanical and Aerospace Engineering, The University of Texas at Arlington, Arlington, TX, USA

Abstract—A MATLAB workflow and graphical user interface (GUI) are presented for region-of-interest (ROI)-guided segmentation and visualization of a suspected tumor region in brain magnetic resonance imaging. The method is applied to one axial FLAIR image from the REMBRANDT collection. A normalized 5×5 binomial kernel smooths the image. Central-difference filters provide a separate edge-magnitude view, while three fixed ROIs guide intensity-interval thresholding. A brightness cutoff and coordinate-based exclusions refine pixel selection. Binary hole filling is then applied to the mask used for the boundary overlay. The mathematical formulation defines each processing stage and its role in constructing the segmentation mask. The GUI displays the input, smoothed image, edge magnitude, isolated region, and boundary overlay in one window. These outputs show the selected region both in isolation and within the surrounding anatomy. The implementation brings image processing, visualization, and user interaction into a single MATLAB workflow.

Keywords—Brain MRI, tumor-region segmentation, region of interest, binomial smoothing, thresholding, MATLAB, graphical user interface.

I. Introduction

Image segmentation separates an image into regions using information such as intensity, local structure, and spatial location. In medical imaging, these operations help isolate areas of interest and display them in relation to surrounding anatomy [1, 2]. Brain MRI provides a useful setting for studying this process because intensity patterns and tissue boundaries can be examined through filtering, thresholding, and mask-based visualization.

Thresholding gives a direct rule for selecting pixels. A global method applies a threshold across the image, while a region-of-interest (ROI) method restricts selection to a defined area. Otsu's method is a well-known example of choosing a global threshold from an intensity histogram [3]. The present method uses intensity intervals calculated from ROI statistics within fixed spatial windows. Each processing stage follows an explicit pixel-selection rule.

Filtering and mask processing serve different roles. Low-pass filtering reduces rapid local intensity changes before thresholding. Derivative filters highlight intensity transitions and provide a structural view of the image [1]. The segmentation mask identifies the selected pixels. A boundary overlay then places the region within the original image.

Two MATLAB implementations were developed: a script that creates five separate figures and a GUI that displays the same outputs in one window. The formulation specifies the smoothing kernel, derivative filters, three ROIs, threshold intervals, coordinate rules, and mask construction. These components connect the processing steps to their visual outputs and provide a consistent basis for image inspection.

The analysis examines the processing outputs for one axial FLAIR image. Broader brain-tumor segmentation studies compare algorithms across subjects using expert-labeled images, as illustrated by the BraTS benchmark [4]. Evaluation on annotated images is the next stage of this work.

II. Image Material and Preparation

The input image, stored locally as `Brain_4.dcm`, is an axial fluid-attenuated inversion recovery (FLAIR) image from the REMBRANDT collection in The Cancer Imaging Archive (TCIA), subject HF0953 [5]. Gusev *et al.* describe the associated REMBRANDT study [6].

The DICOM metadata specify an image size of 256×256 pixels, in-plane spacing of 0.938×0.938 mm, and slice thickness of 3.0 mm. The repetition, echo, and inversion times are 10004, 155, and 2200 ms, respectively. The series description is `AXIAL FLAIR`.

The file is read using MATLAB's `dicomread` function [7]. The image is stored in single precision and converted to double precision for convolution. Processing uses the stored intensity values. Let $I(i, j)$ denote the intensity at row i and column j , with

$$\Omega = \{1, \dots, R\} \times \{1, \dots, C\}. \quad (1)$$

Coordinates follow MATLAB's one-based row-column indexing. ROI limits and exclusion rules are specified in pixels. References to the left or right side describe the displayed image, rather than patient-coordinate directions.

The filtered image, isolated intensity map, and boundary overlay are used to examine the segmented region.

III. Mathematical Formulation

A. Processing Structure

Figure 1 shows the workflow. The input supplies two branches. The first applies central-difference filters to produce an edge-magnitude image. The second smooths the input and selects pixels using ROI-specific intensity intervals and exclusion rules. The edge magnitude is a display output, while the smoothed image supplies the segmentation thresholds. Original intensities are retained in the isolated region and used as the background for the boundary overlay.

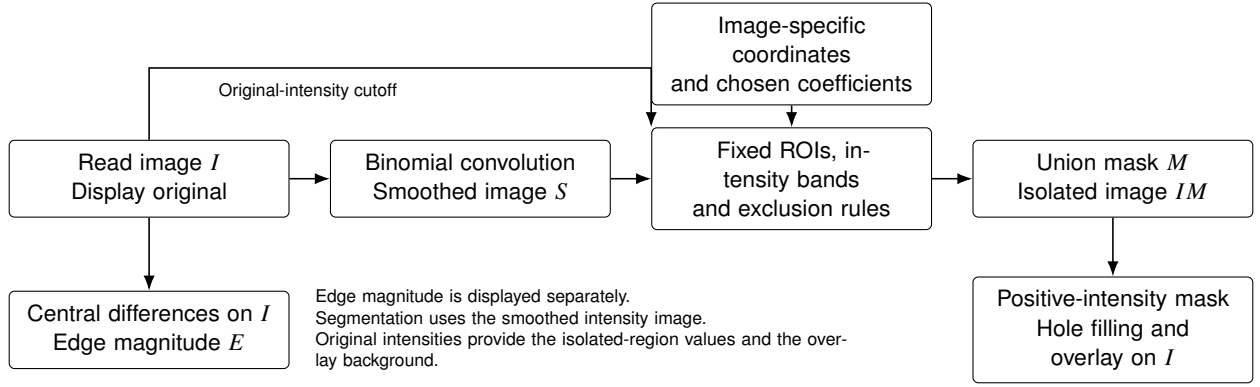


Figure 1. Processing workflow for ROI-guided brain MRI segmentation and visualization. The original image supplies separate edge-display and segmentation branches. The segmentation mask is formed from the smoothed image, ROI settings, intensity intervals, and exclusion rules.

B. Normalized Binomial Smoothing

The low-pass filter uses the kernel

$$K = \frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}. \quad (2)$$

This separable binomial kernel approximates Gaussian smoothing and implements a weighted local average [1]. With the entries of b indexed from -2 to 2 ,

$$b = \frac{1}{16} [1 \ 4 \ 6 \ 4 \ 1], \quad K = b^T b. \quad (3)$$

All weights are nonnegative and sum to one. Away from the padded image boundary, the filter therefore preserves constant intensity. The smoothed image is

$$S(i, j) = \sum_{p=-2}^2 \sum_{q=-2}^2 K(p, q) I(i - p, j - q). \quad (4)$$

Here, $K(p, q)$ denotes the centered indexing of the matrix in (2). Convolution is implemented using MATLAB's `conv2` function [8]:

$$S = \text{conv2}(\text{double}(\text{img}), K, \text{'same'});$$

The 'same' option returns an array with the input dimensions. Values outside the image are treated as zero during convolution.

The kernel's frequency response explains its smoothing action. For row and column angular spatial frequencies ω_r and ω_c , measured in radians per pixel, the centered response is

$$H(\omega_r, \omega_c) = \cos^4\left(\frac{\omega_r}{2}\right) \cos^4\left(\frac{\omega_c}{2}\right). \quad (5)$$

This follows from the one-dimensional transform, $(6 + 8 \cos \omega + 2 \cos 2\omega)/16 = \cos^4(\omega/2)$, and the separable form in (3). The response equals one at zero frequency and falls toward zero at the highest row or column spatial frequencies. The filter therefore reduces rapid intensity changes while retaining slowly varying image structure. This averaging also softens sharp boundaries and small features.

C. Central-Difference Edge Visualization

Spatial derivatives highlight local intensity changes [1]. The horizontal and vertical kernels are

$$K_x = \frac{1}{2} \begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad K_y = \frac{1}{2} \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \quad (6)$$

Both kernels operate on the original image:

$$E_x = I * K_x, \quad E_y = I * K_y, \quad E = \sqrt{E_x^2 + E_y^2}. \quad (7)$$

The powers and square root are evaluated pixelwise. Because `conv2` performs convolution [8], the interior responses are

$$E_x(i, j) = \frac{I(i, j-1) - I(i, j+1)}{2}, \quad (8)$$

$$E_y(i, j) = \frac{I(i-1, j) - I(i+1, j)}{2}.$$

Reversing either directional sign leaves E unchanged because the responses are squared. The derivative responses are expressed per pixel.

The resulting gradient magnitude emphasizes the outer contour and internal transitions. The directional convolutions are linear, while their magnitude combination is nonlinear. This edge-display branch complements the segmentation view by showing local structure independently of the ROI mask.

D. ROI Statistics and Intensity Intervals

Three rectangular ROIs define the image areas considered for segmentation. Their coordinates and coefficients are listed in Table 1. The main ROI supplies the mean and maximum smoothed intensities used by all three branches:

$$\mu = \frac{1}{|\mathcal{R}_1|} \sum_{(i,j) \in \mathcal{R}_1} S(i, j), \quad m = \max_{(i,j) \in \mathcal{R}_1} S(i, j). \quad (9)$$

Let $\Delta = m - \mu$, with $\Delta \geq 0$. Each ROI uses a lower and upper threshold:

$$L_k = \mu - a_k \Delta, \quad U_k = \mu + b_k \Delta, \quad k \in \{1, 2, 3\}. \quad (10)$$

The coefficients a_k and b_k set the interval width below and above the ROI mean. These image-specific settings were

Table 1. ROI coordinates and intensity-interval coefficients. Index limits are inclusive, and every interval uses the statistics of $\mathcal{R}_1$.

ROI	Rows	Columns	a_k	b_k
$\mathcal{R}_1$	60:150	72:132	0.05	0.78
$\mathcal{R}_2$	86:112	132:152	0.10	0.72
$\mathcal{R}_3$	76:92	80:94	0.08	0.76

selected for the displayed target region. Because $b_k < 1$, the upper threshold lies below the ROI maximum when $\Delta > 0$. The rule therefore selects an intensity band rather than retaining only the brightest pixels.

The main window covers the central target area. The second window extends selection toward the image-right side, and the third covers an upper-left subregion. ROI restriction supplies the location information, while the threshold intervals determine which intensities are retained within those windows.

E. Brightness and Coordinate Exclusions

A brightness cutoff is applied to the original image:

$$s = 0.90 \max_{(i,j) \in \Omega} I(i,j), \quad Q(i,j) = \mathbf{1}\{I(i,j) < s\}. \quad (11)$$

This condition excludes pixels at or above 90% of the global maximum. It complements thresholding of the smoothed image by limiting selection of very bright original pixels.

Additional coordinate rules refine selection within each ROI. Let $P_k(i,j)$ be a binary acceptance indicator. For the main ROI,

$$\begin{aligned} P_1(i,j) = & \mathbf{1}\{-(i < 85 \wedge j < 90)\} \\ & \cdot \mathbf{1}\{-(i > 128 \wedge j > 118)\} \\ & \cdot \mathbf{1}\{-(62 \leq i \leq 74 \wedge 86 \leq j \leq 94)\}. \end{aligned} \quad (12)$$

For the second and third ROIs,

$$P_2(i,j) = \mathbf{1}\{-(i < 92 \wedge j > 145)\}, \quad (13)$$

$$P_3(i,j) = \mathbf{1}\{-(62 \leq i \leq 74 \wedge 86 \leq j \leq 94)\}. \quad (14)$$

Here, $\mathbf{1}\{\cdot\}$ is one when a condition is true and zero otherwise; $\wedge$ denotes logical AND and $\neg$ denotes NOT.

The exclusions act within their respective branches. Thus, a pixel excluded by one branch can still be accepted by an overlapping branch. Within $\mathcal{R}_3$, the excluded row range in (14) lies outside the ROI, so $P_3 = 1$ throughout that window.

F. Mask Construction and Region Isolation

The selected-pixel mask for branch k is

$$\begin{aligned} M_k(i,j) = & \mathbf{1}\{(i,j) \in \mathcal{R}_k\} Q(i,j) P_k(i,j) \\ & \cdot \mathbf{1}\{L_k \leq S(i,j) \leq U_k\}. \end{aligned} \quad (15)$$

The three masks are combined by logical union:

$$M = M_1 \vee M_2 \vee M_3. \quad (16)$$

The MATLAB loops implement this union by assigning the positive label 64 whenever a branch accepts a pixel. The

selected locations are then used to retain the original image intensities:

$$I_{\text{iso}}(i,j) = I(i,j)M(i,j). \quad (17)$$

Pixels outside M are set to zero. This keeps the intensity information from the input image while using the smoothed image to make the interval decisions.

G. Hole Filling and Boundary Display

A positive-intensity mask is formed from the isolated image, followed by binary hole filling:

$$B_0(i,j) = \mathbf{1}\{I_{\text{iso}}(i,j) > 0\}, \quad B = \text{FillHoles}(B_0). \quad (18)$$

Hole filling is a morphological operation that adds enclosed background regions to a binary mask [9]. The implementation uses `imfill(B0, 'holes')` [10].

The three masks have distinct roles. M records threshold selection, B_0 retains selected pixels with positive original intensity, and B fills enclosed gaps for boundary display. The isolated intensity image uses M , while the overlay uses B . Filling can therefore enclose pixels that were absent from the intensity-selected region. The final contour is drawn over the original image to show the location and extent of the displayed region.

IV. MATLAB Implementation and Interface

The workflow was implemented in a processing script and a GUI. The script, `brain_mri_segmentation.m`, reads `Brain_4.dcm` using `dicomread` [7] and produces five individual figure windows. Code listings and implementation materials are shared through the GitHub repository listed under Data and Code Availability.

The GUI, implemented in `brain_tumor_detection.m`, uses a MATLAB figure, five axes, and pushbutton controls created with `uicontrol` [11]. Selecting *Upload Brain Image* opens a file-selection dialog. DICOM images are read with `dicomread`; JPEG, PNG, and TIFF inputs are read with `imread`. Color images in the non-DICOM branch are converted to grayscale with `rgb2gray`.

Image loading starts the processing callback. The first panel displays the input, and the remaining panels show the smoothed image, edge magnitude, isolated region, and boundary overlay. The *Clear* callback resets the axes and stored image.

The ROI coordinates and coefficients are fixed in the processing function. The GUI caps the upper ROI indices at the image dimensions. Input images must be two-dimensional, contain nonempty ROIs within the specified coordinates, and match the intended region location. Image-specific ROI adjustment is required when the target location or image geometry changes.

V. Results

A. Input and Low-Pass Output

Figure 2 compares the input MRI image with the low-pass result. The image contains a comparatively bright, irregular target region on the image-left side. Smoothing reduces fine-scale intensity variation and softens the transition between

neighboring pixels. The larger regional pattern remains visible after filtering.

Each panel uses `imagesc` with its own intensity limits. The colorbars identify these ranges, allowing the smoothing effect to be interpreted from both the image structure and displayed values.

B. Edge Magnitude and Isolated Region

Figure 3(a) shows the central-difference edge magnitude. The outer head contour and internal transitions are emphasized. These features provide a separate view of image structure, while segmentation follows the ROI and threshold rules.

Figure 3(b) shows the selected region against a zero-valued background. Retained pixels carry their original intensities. The outline and internal gaps reflect the combined effects of the three ROI windows, bounded intensity intervals, brightness cutoff, and coordinate exclusions. Pixels that fail these conditions remain outside the isolated intensity map.

C. Boundary Overlay and GUI

Figure 4(a) displays the boundary after the positive-intensity mask is hole-filled. The red contour places the selected region within the original MRI image. This view complements the isolated image by retaining the surrounding anatomical context.

The GUI in Fig. 4(b) brings the five outputs together. Image loading initiates processing, and the interface presents the original, smoothed, edge-magnitude, isolated, and marked views in a consistent arrangement. The upload and clear controls support repeated image inspection within the same window.

VI. Discussion

A. Connection Between Theory and Output

The formulation explains the behavior of each stage. The normalized kernel produces weighted averaging, and its frequency response explains the reduction of rapid spatial changes. Central differences emphasize intensity transitions. ROI statistics set the threshold bands, while coordinate rules restrict the accepted locations. The logical union combines these decisions into the isolated region. Hole filling then produces the mask used for the overlay.

The isolated image retains the pixels accepted by the selection rules. The overlay traces the hole-filled mask, which can enclose gaps within the selected region. Displaying both outputs shows how post-processing affects the final boundary.

B. Parameter Dependence and Study Scope

Intensity statistics and spatial settings jointly determine the selected region. The ROI mean and maximum set the threshold values; coordinates and interval coefficients are selected for the target image. Different slice positions, orientations, resolutions, or target locations require corresponding ROI adjustments. The global maximum controls the brightness cutoff, so high-intensity pixels also influence selection.

The study reports a single-image segmentation and visualization result. Quantitative assessment of detection accuracy

and generalization requires independent images with expert reference masks. The workflow is intended for image analysis and method development, rather than clinical diagnosis. Extension to three-dimensional analysis requires a complete image volume and consistent region definitions across slices.

VII. Further Development and Evaluation

Further development will focus on annotated images, a consistent ROI-selection procedure, and comparisons with established segmentation methods. BraTS provides an example of a benchmark framework for expert-labeled brain-tumor segmentation [4]. Development and test images should be separated by subject, with parameter selection completed before independent evaluation.

For a predicted region A and expert reference region G , the Dice coefficient [12] and intersection over union are standard overlap measures [13]:

$$\text{Dice}(A, G) = \frac{2|A \cap G|}{|A| + |G|}, \quad (19)$$

$$\text{IoU}(A, G) = \frac{|A \cap G|}{|A \cup G|}. \quad (20)$$

These measures define the planned quantitative evaluation. A convention for empty masks must be specified, and the intensity-selected mask M and hole-filled mask B should be evaluated separately.

A controlled comparison can keep the ROI procedure fixed while examining smoothing, interval thresholding, the brightness cutoff, and coordinate exclusions. Otsu thresholding [3] provides a useful comparison under the same ROI restriction. Sensitivity to small ROI shifts and coefficient changes will help characterize dependence on manual settings. Runtime measurement and input-handling tests will support further GUI development.

VIII. Conclusion

A MATLAB workflow and GUI were developed for ROI-guided segmentation and visualization of a suspected tumor region in an axial FLAIR MRI image. The implementation combines normalized binomial smoothing, central-difference edge visualization, three ROI-based intensity intervals, brightness and coordinate exclusions, and binary mask processing. The mathematical formulation describes the complete pixel-selection rule and the boundary-display operation.

The image outputs show the selected region both in isolation and within its original anatomical context. The GUI integrates all five views and provides upload and clear controls. The formulation and shared implementation materials support further evaluation on annotated MRI data.

Author Contributions

Nafisa Ibnat Tasfee developed the image-processing workflow, wrote the MATLAB code, designed the GUI, analyzed the image outputs, and prepared the manuscript.

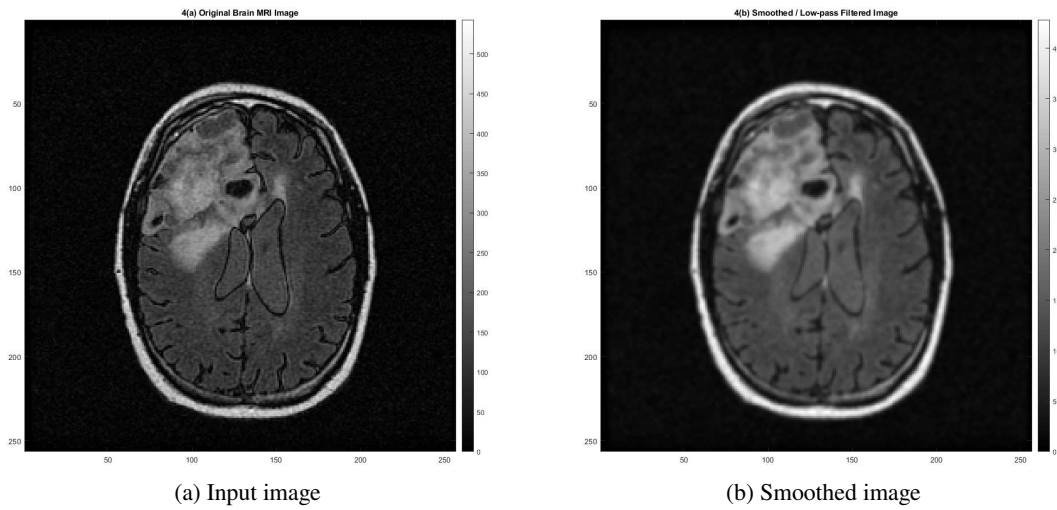


Figure 2. Axial FLAIR image from the REMBRANDT collection in TCIA, subject HF0953 [5, 6]: (a) input image and (b) output after 5×5 binomial smoothing. Each panel uses its own displayed intensity scale.

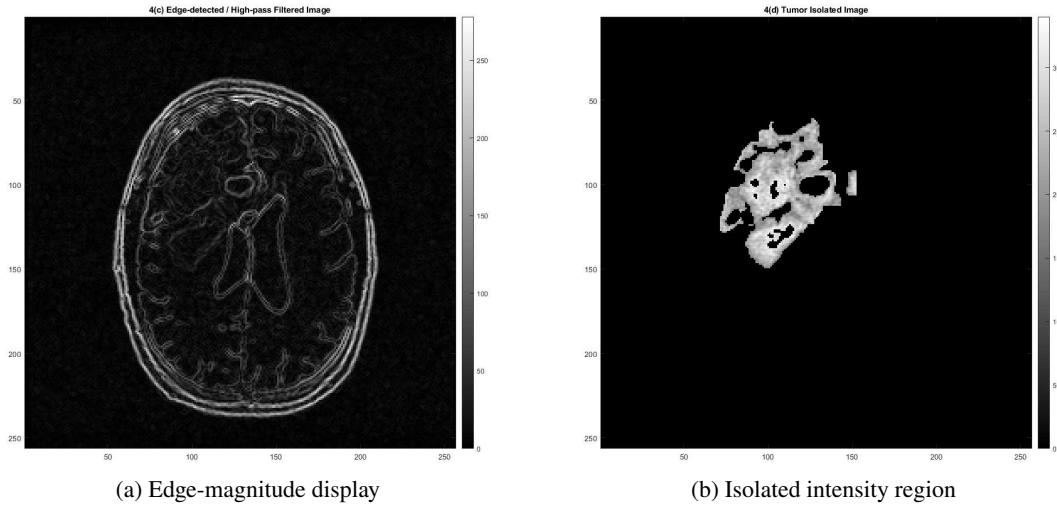


Figure 3. Processing outputs for the REMBRANDT axial FLAIR image, subject HF0953 [5]: (a) central-difference edge magnitude and (b) the ROI-segmented region with original pixel intensities.

Data and Code Availability

MATLAB code listings and implementation materials for the image-processing workflow and GUI are available in the author's GitHub repository [14]:

<https://github.com/ntasfee21/Tumor-Detection-from-MRI-image>

The input MRI is an axial FLAIR image from the REMBRANDT collection in TCIA, subject HF0953 [5, 6]. The source image is obtained from the cited collection.

AI-Assistance Disclosure

AI tools assisted with language editing, manuscript organization, code formatting, and presentation of the mathematical formulation.

References

- [1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Pearson, 2018.
- [2] D. L. Pham, C. Xu, and J. L. Prince, "Current methods in medical image segmentation," *Annual Review of Biomedical Engineering*, vol. 2, pp. 315–337, 2000. doi: 10.1146/annurev.bioeng.2.1.315.
- [3] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979. doi: 10.1109/TSMC.1979.4310076.
- [4] B. H. Menze *et al.*, "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Transactions on Medical Imaging*, vol. 34, no. 10, pp. 1993–2024, 2015. doi: 10.1109/TMI.2014.2377694.
- [5] L. Scarpace, A. E. Flanders, R. Jain, T. Mikkelsen, and D. W. Andrews, "Data from REMBRANDT," [data set], The Cancer Imaging Archive, 2015. doi: 10.7937/K9/TCIA.2015.588OZUZB.
- [6] Y. Gusev *et al.*, "The REMBRANDT study, a large collection of genomic data from brain cancer patients," *Scientific Data*, vol. 5, art. 180158, 2018. doi: 10.1038/sdata.2018.158.

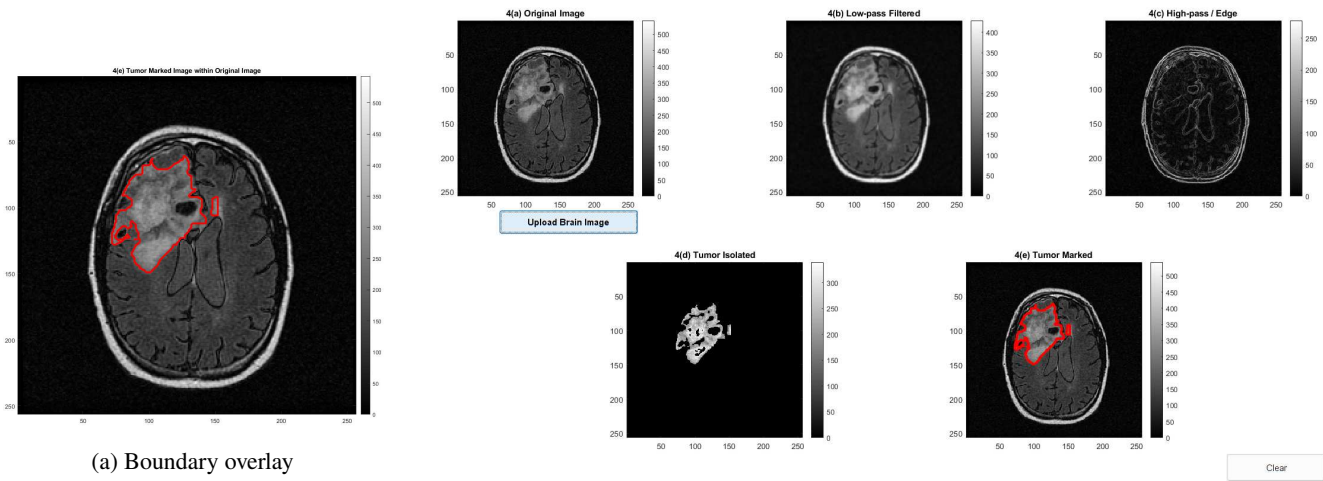


Figure 4. Boundary visualization and MATLAB GUI for the REMBRANDT image, subject HF0953 [5]. (a) The red contour outlines the hole-filled mask on the input image. (b) The interface presents all five outputs with image-upload and clear controls.

- [7] The MathWorks, Inc., “dicomread—Read DICOM image,” *MATLAB documentation*. [Online]. Available: <https://www.mathworks.com/help/images/ref/dicomread.html>.
- [8] The MathWorks, Inc., “conv2—2-D convolution,” *MATLAB documentation*. [Online]. Available: <https://www.mathworks.com/help/matlab/ref/conv2.html>.
- [9] P. Soille, *Morphological Image Analysis: Principles and Applications*, 2nd ed. Springer, 2003. doi: 10.1007/978-3-662-05088-0.
- [10] The MathWorks, Inc., “imfill—Fill image regions and holes,” *MATLAB documentation*. [Online]. Available: <https://www.mathworks.com/help/images/ref/imfill.html>.
- [11] The MathWorks, Inc., “uicontrol—Create user interface control,” *MATLAB documentation*. [Online]. Available: <https://www.mathworks.com/help/matlab/ref/uicontrol.html>.
- [12] L. R. Dice, “Measures of the amount of ecologic association between species,” *Ecology*, vol. 26, no. 3, pp. 297–302, 1945. doi: 10.2307/1932409.
- [13] A. A. Taha and A. Hanbury, “Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool,” *BMC Medical Imaging*, vol. 15, art. 29, 2015. doi: 10.1186/s12880-015-0068-x.
- [14] N. I. Tasfee, “Tumor-Detection-from-MRI-image,” *GitHub repository*. [Online]. Available: <https://github.com/n-tasfee21/Tumor-Detection-from-MRI-image>.