2635047 [06 30241]

Introduction

Brain MRI tissue segmentation is a crucial technique in medical image analysis, allowing the analysis of different brain tissue types. By segmenting brain MRI scans, we can detect abnormalities, monitor disease progression, isolate brain tissues, and monitor brain functions.

In this report, we use a combination of boundary-based and region-based segmentation methods to identify five distinct tissue layers within a T1-weighted brain MRI image of dimensions 362 x 434 x 10, where the third dimension represents 1 mm of depth intervals. We apply these methods to both individual 2D slices and the complete 3D brain volume. We compare the results of these techniques using various performance metrics and discuss about the optimal strategy for Brain MRI Segmentation.

Method

Task 1 : Our Algorithm begins with a Preprocessing Phase. We normalise the 2D slices using min-max normalisation to enhance image contrast, which benefits subsequent thresholding techniques. Then, we apply a Gaussian filter with a standard deviation of 1 to reduce noise while preserving boundary information. Next, we use the Triangle Threshold algorithm to remove the surrounding air from the image. The resulting mask is refined using morphological opening to further reduce noise. The morphological closing/opening operations implemented in these methods utilises a 3x3 kernel with 9 active members and performs this operation until there are 2 distinct regions in the image, i.e. without any holes or noise respectively.

Triangle Threshold algorithm utilises the Triangle Algorithm [1] to strip the surrounding air from our object of interest. This algorithm establishes a line between the peak intensity bin and the farthest bin and considers the point of maximum distance from the line to the histogram as the Triangle Threshold. This threshold effectively differentiates the brain (as the object of interest) against the uniform background air. The Triangle Threshold is robust to various normalisation techniques due to its direct operation on the image histogram.

After Preprocessing we perform skull stripping, we apply Otsu's thresholding to the MRI image to obtain a rough estimate of the combined skin and brain masks. Then, we utilise contour finding to identify the skin region within the Otsu mask. A morphological closing operation is then applied to fill small holes in the skin mask. The contour finding method uses an marching squares algorithm [2], which locates curves of constant intensity values (isovalues) within an image. In this algorithm, each 2x2 pixel block is analysed by a cell, with its four corner intensity values represented by 4-bits, the 4-bits form a pattern when a value overlaps with a region in the image. Using a lookup table, these patterns correspond to specific contour line configurations within the block. Linear interpolation ensures accurate contour placement between pixels. We leverage this flexible method to extract skin boundaries from the Otsu mask; the first two contour indices reliably delineate the skin contours, allowing us to create a rough skin mask.

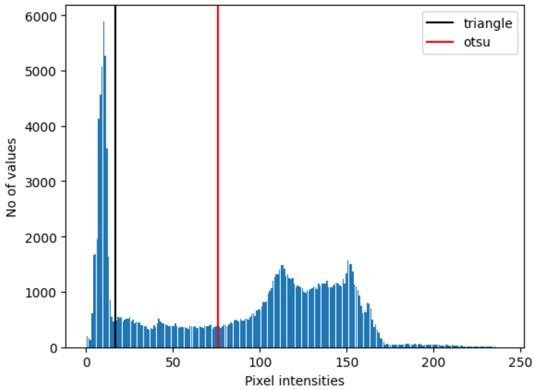

Figure 1: Comparison between the Triangle and Otsu Threshold methods

Table 1: Analysis of different normalization techniques for the 2D algorithm

Normalization	JI	DSC
Min-Max	0.86602	0.92732
Z-Score	0.37307	0.45909
Histogram Norm	0.75977	0.84871

Removing the skin region from the Otsu mask, we further remove small holes from the resultant brain mask using a morphological operation, The remove_small_holes morphological operation from scikit-image targets contiguous holes smaller than an area of 100^2 pixels (chosen based on prior domain knowledge). This method identifies connected components within the binary mask, filling these holes with neighbouring pixel values. Finally, a morphological opening operation removes residual noise from the brain mask. We identify and remove the noise region around the skin by applying contour finding to a binary version of the image with skin and brain masks applied. The resultant region consists of the skull and the CSF regions. We use K-Means clustering to separate the skull region by sorting the centroid indices and indexing into the first label; this method is robust due to the skull region having a lower intensity compared to the neighbouring regions.

After Skull Stripping, we prepare the final mask. We remove the skin, noise, and skull regions from the original binary image and apply morphological opening to eliminate any remaining noise, the resultant processed mask enables us to perform segmentation effectively on the brain region. We use the processed mask and perform K-Means clustering with four clusters to segment the brain region into Background, CSF, Gray Matter, and White Matter. We assign labels to each cluster based on their intensity values, resulting in separate masks for CSF, Gray Matter, and White Matter. All the masks are added as different labels into a new 2D image.

Task 2: For the evaluation of the algorithms, we use the Jaccard Score (JI) and the Dice coefficient (DCS) as they are widely used as segmentation metrics in many applications. Both metrics consider the overlap with respect to the ground-truth label allowing for different algorithm and parameter settings. Additionally, both scores range between 0-1, with 1 indicating a perfect overlap with the ground truth.

Jaccard Score (JI) : Also known as intersection over union (IoU), Calculated as the ratio between the intersection of predicted and ground truth to their union.

Dice Coefficient (DSC) : Approximated with the F1 score, calculated as the two times the intersection of sets divide by the sum of their sizes.

The Chosen 2 metrics are robust for our application of Brain MRI Segmentation. We used the 'macro' averaging method for both metrics, which calculates the average score across all classes (tissue types) in the segmentation. The F1 score with the ‘macro’ averaging method becomes equivalent to the Dice Coefficient. The results were presented in tables and graphs to facilitate comparisons between different algorithms and parameter settings.

Task 3: Our 3D segmentation approach shares similarities with the 2D method, with modifications to accommodate volumetric data. Preprocessing begins with min-max normalization of the 3D image. Subsequently, the Triangle Threshold Algorithm isolates air region, followed by a 3D morphological opening operation. The morphological closing/opening operations implemented in the 3D method utilises a 3x3x3 ball kernel with 7 active members and iterating until we obtain two distinct regions within the 3D image.

In the Skull Stripping Phase, Otsu's thresholding is applied to the entire 3D volume that yields a combination of the skull and the brain masks. Here, we utilise a morphological operation called connected component labelling algorithm [3] to find distinct regions in our otsu 3D mask. The algorithm works by labelling connected pixels similarly, resulting in background, skin and the brain masks. Both skin and brain masks are refined using a morphological closing operation to mitigate any noise. To address holes within the 3D brain mask, we use remove_small_holes with an adjusted area threshold of 100^2 x 10 to account for image depth. Similarly we identify and remove the noise region by using the connected component labelling algorithm to the binary 3d image with skin and mask applied. Further, applying the noise mask we can strip out the skull mask using KMeans algorithm.

After the Skull Stripping the algorithm follows the same procedure as the 2D Segmentation, The K-Means clustering algorithm segments individual brain tissues, producing distinct masks. These masks are combined as labels within a new 3D image, completing the 3D segmentation process. The application of connected component labelling from 3D segmentation was further extended to the 2D segmentation and compared against the original method.

Results

Task 1 : We have implemented 2 different algorithms for 2D Segmentation, 2D and 2DX.

2D : The algorithm that utilised the contour finding method achieved a Jaccard score of 0.87 and a Dice coefficient of 0.93 on average across all the 10 slices. The Average time taken by this algorithm is 2.9388 seconds per each slice. The algorithm has an acceptable accuracy, however, is computationally expensive and takes longer than the other method.

2DX: This algorithm utilises the connected component labelling instead of contour finding and achieved a Jaccard score of 0.89 and a Dice coefficient of 0.94 on average. The Average time taken by this algorithm is 1.2296 seconds. This algorithm has slightly more accuracy and even has faster execution time compared to the 2D algorithm.

Task 2 : From the analysis in the Table 1, Min-Max Normalization seems to be the more valuable normalization for our segmentation task.

From Table 2, We can analyse the kernel size hyperparameter of the morphological closing and opening operations and conclude that the 3D algorithm is more robust to size of dilation whereas the 2D algorithm fluctuates a fair amount.

Table 2: Analysis of kernel size

Kernel Size	JI		DSC	
	2D	3D	2D	3D
NIL	0.82545	0.83477	0.90124	0.90635
3	0.86602	0.88634	0.92732	0.93861
5	0.86834	0.88348	0.92851	0.93700
7	0.80589	0.88174	0.89043	0.93563

Task 3 : Like the 2DX algorithm the 3D algorithm utilises the connected component method and achieves a Jaccard score of 0.89 and a Dice coefficient of 0.94 on average across all the slices. The Execution time of this algorithm is 10.8611 seconds, which is at least 10% faster than the 2DX algorithm considering all 10 slices. This speedup can be attributed to the 3D algorithm's ability to leverage information from multiple slices simultaneously, reducing redundant computations. The specific implementation of the algorithms and the hardware used can influence the results. However, this improvement in speed can be significant for real-time applications.

Table 3: Segmentation results for all the algorithms

Algorithms	JI	DSC	Time(sec)
2D	0.87	0.93	29.388 ± 0.569
2DX	0.89	0.94	12.296 ± 0.747
3D	0.89	0.94	10.8611

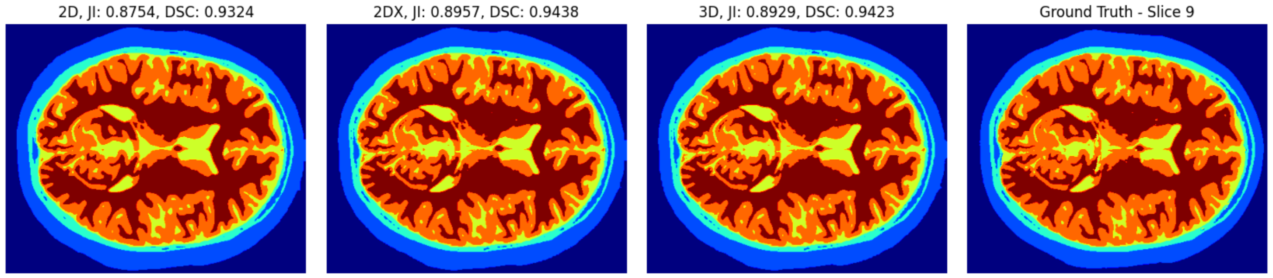

Figure 2: Results from each algorithm with the Ground Truth of Slice 9

Conclusion

This report emphasises the effectiveness of various 2D and 3D segmentation algorithm for segmenting brain MRI data into five tissue layers: air, skin/scalp, skull, CSF, gray matter, and white matter. We implemented and evaluated two 2D algorithms and a 3D algorithm. The Jaccard score and Dice coefficient were used to assess the accuracy of each algorithm by comparing the predicted segmentations with the provided ground truth labels. The results showed that the 3D algorithm achieved the highest segmentation accuracy, outperforming the 2D method in terms of Jaccard Index, Dice coefficient and even the Execution time. This suggests that leveraging information from multiple slices simultaneously in the 3D approach leads to more accurate and robust segmentation. Additionally, the 3D algorithm exhibited a 10% improvement in computational efficiency compared to the best-performing 2D algorithm, making it a promising choice for processing large datasets or time-sensitive applications.

References

[1] “Automatic measurement of sister chromatid exchange frequency.” Accessed: Mar. 17, 2024. [Online]. Available: https://journals.sagepub.com/doi/epdf/10.1177/25.7.70454

[2] C. Maple, “Geometric design and space planning using the marching squares and marching cube algorithms,” in *2003 International Conference on Geometric Modeling and Graphics, 2003. Proceedings*, London, UK: IEEE Comput. Soc, 2003, pp. 90–95. doi: 10.1109/GMAG.2003.1219671.

[3] K. Wu, E. Otoo, and A. Shoshani, “Optimizing connected component labeling algorithms,” Jan. 2005, Accessed: Mar. 18, 2024. Available: https://escholarship.org/uc/item/7jg5d1zn