

Medical Imaging Final Assignment

Refo Ilmiya Akbar

1 Introduction and Problem Statement

In this project, the objective is to load, visualize, and perform coregistration of DICOM images, specifically focusing on the HCC-TACE-Seg dataset and the assigned patient HCC XYZ. The project involves using PyDicom to load the DICOM images and rearranging the pixel arrays based on relevant headers such as 'Acquisition Number', 'Slice Index', 'Per-frame Functional Groups Sequence', and 'Segment Identification Sequence'. Additionally, a third-party DICOM visualizer like 3D-Slicer is recommended for visualization. The goal is to create an animation showcasing a rotating Maximum Intensity Projection on the coronal-sagittal planes. Furthermore, the project includes coregistration of the images using landmarks or a custom similarity measure, without relying on external libraries. The Thalamus region in the input image space is specifically visualized. This project addresses the need for efficient loading, visualization, and coregistration of DICOM images, contributing to enhanced understanding and analysis of medical imaging data.

2 Algorithm Description and Implementation

In this section, the implementation and algorithms used in each aspect of the project will be discussed. The focus will be on the steps taken to execute the project, including the loading, visualization, and coregistration of DICOM images. The algorithmic approaches employed for each task will be elaborated upon, highlighting the key techniques utilized to achieve the desired results.

2.1 DICOM loading and visualization

The dataset assigned HCC_001 was downloaded using NBIA Data Retriever. Subsequently, it was visualized using 3D Slicer to gain a better understanding of the dataset. Multiple acquisitions of the CT-scan were observed, indicating that the segmentation data is only valid for a specific acquisition, namely the 3.000000-C-A-P-42120 DCM files taken on November 30, 1999. Additionally, it was determined that there are four segmentation sequences in the dataset: liver, tumor, vessels, and aorta.

Using Python, the DCM files corresponding to the second acquisition were loaded along with their segmentation data. The three-dimensional array of the

DCM files was also loaded. The information from the DCM files of the segmentation dataset was examined to identify the slices in the array corresponding to each segment. To ensure correct orientation, the segmented image was flipped on the second axis. Further analysis revealed that the array indices for the tumor were within the range of 37-73.

To create a segmented image, alpha-fusion was performed iteratively throughout the first axis, with an alpha value of 0.2 for the mask and 0.8 for the image. Once the segmented images were obtained, they were visualized by plotting both the image and segmented image. The image was assigned an alpha value of 0.8 to create an overlay effect on the CT scan. Multiple visualizations were conducted using Median, Maximum Intensity Projection (MIP), and Average Intensity Projection (AIP) across the sagittal and coronal planes.

To generate a GIF file of the segmented image, axial rotation was applied. This involved using 48 projections with an interval of 30 degrees. For each projection, the image and segmented image were rotated by a certain degree, and the MIP sagittal plane of both images was plotted. A loop was implemented over the projections, and the resulting frames were combined using ArtistAnimation to generate the GIF file. The same process was repeated for the coronal plane to examine any differences in the results.

2.2 3D Rigid Coregistration

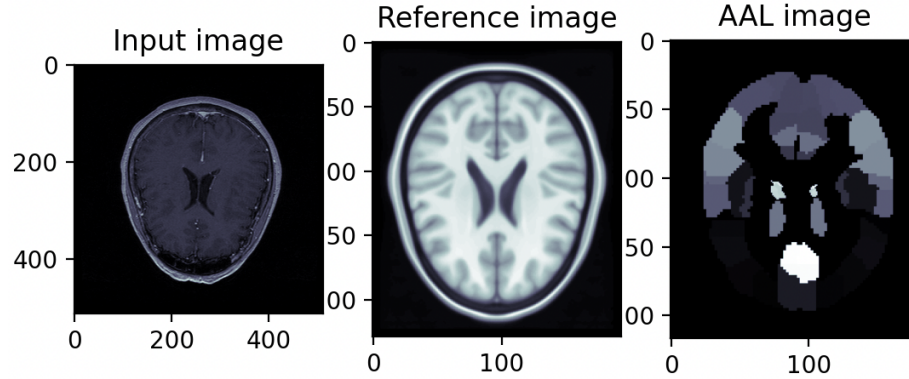


Figure 1: Initial visualization of the images

The three DCM files, namely the reference image (icbm avg152t1talnlin symmetricVI), the segmentation image (AAL31mm), and the input image (RM-Brain3D SPGR), were loaded and their pixel arrays were extracted for further analysis. Horizontal visualizations were performed to gain a better understanding of the dataset, revealing variations in size and features among the images as can be seen on Figure 1. Preprocessing steps were then carried out to address these differences.

The reference image and segmentation image were cropped to match the size of (181, 216, 181). As for the input image, it underwent several transformations. Firstly, it was cropped and zoomed to align its region of interest with the other images using the 'zoom()' function from the scipy library. A rotation of 3 degrees was applied to ensure the input image had the same orientation as the reference image. Lastly, intensity normalization was performed with respect to the reference image to achieve consistent intensity levels. Finally, normalization is done to the input image achieve similar range of values with the reference image. Horizontal, sagittal, and coronal visualizations were conducted to assess the effects of the preprocessing as shown on Figure 2.

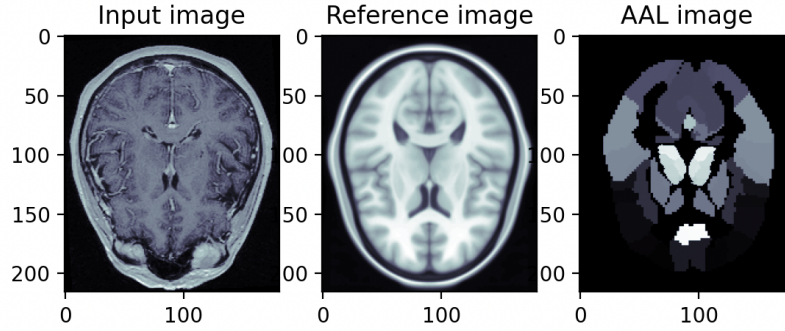


Figure 2: Preprocessed images

The coregistration part is then initiated. Firstly, the reference and input images are transformed into 3-dimensional arrays using the `reshape(-1,3)` function. The residual vector between each point is subsequently examined to observe the disparities. It is observed that there exists a significant difference between the two. Subsequently, a 3D visualization is performed on each point of the 3D arrays to facilitate comparison. Now, the data is prepared for coregistration by implementing the `coregister_landmarks()` function. The `coregister_landmarks()` function enables the coregistration of two sets of landmarks through a rigid transformation. Initial parameters, including a translation vector and an axis of rotation, are defined as the starting point for the transformation. The centroids of the reference landmarks and input landmarks are then computed to update the translation vector, aligning the centroids of both sets. A function named `function_to_minimize` is defined to calculate the residuals by transforming the input landmarks and comparing them to the reference landmarks. The optimization process is carried out using the `least_squares` function, aiming to minimize the residuals by adjusting the transformation parameters. Downsampled arrays of the input and reference images are created, and coregistration is performed between the two sets to obtain the optimal coregistration parameters. This downsampling is done to reduce the computation time as the original data size is considerable. The resulting parameters, $[-1.78, -2.00, -2.38]$ along the x, y, z axes and a rotation of 0.15 radians along the axis (0.62, 0.51, 0.62), are

obtained. Subsequently, the `translatino_then_axialrotation()` function is invoked to adjust the input image with the reference image. Another visualization is performed to observe the effects of the coregistration process, the comparison between the two can be seen on figure 3. At this stage, the coregistration part is completed.

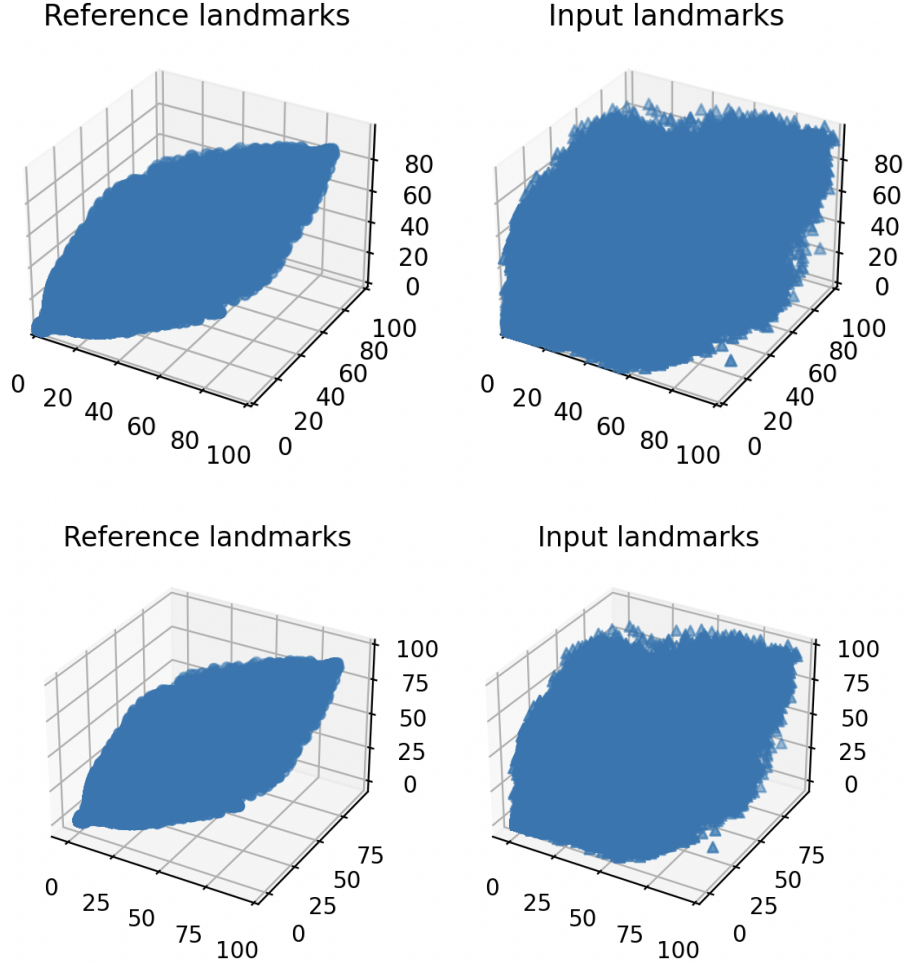


Figure 3: Comparison of the landmark points before and after coregistration

For the thalamus region masking, the `'getthalamusmask()'` function was implemented to generate a mask by selecting the thalamus indexes, specifically [121-150]. The `'visualaxialslice()'` function was then utilized to perform alpha fusion between the thalamus region mask and the input image. The same fusion was applied to the reference image for verification. To observe the masking

in the input space, a reverse transformation was performed. The fused image was sliced to obtain a 2-dimensional array, and its intensity was matched with the original image intensity after coregistration. The coregistration process was repeated with the inverted values of the best parameters, followed by intensity matching with the original image before coregistration. Subsequently, the preprocessing steps were repeated with the inverted values, including rotation, resizing, and zooming out, with padding added to restore the original size of (512, 512). Finally, a visualization of the mask in the input space was generated.

3 Results and Discussion

The results section of the report presents the outcomes of the conducted analysis.

3.1 DICOM loading and visualization

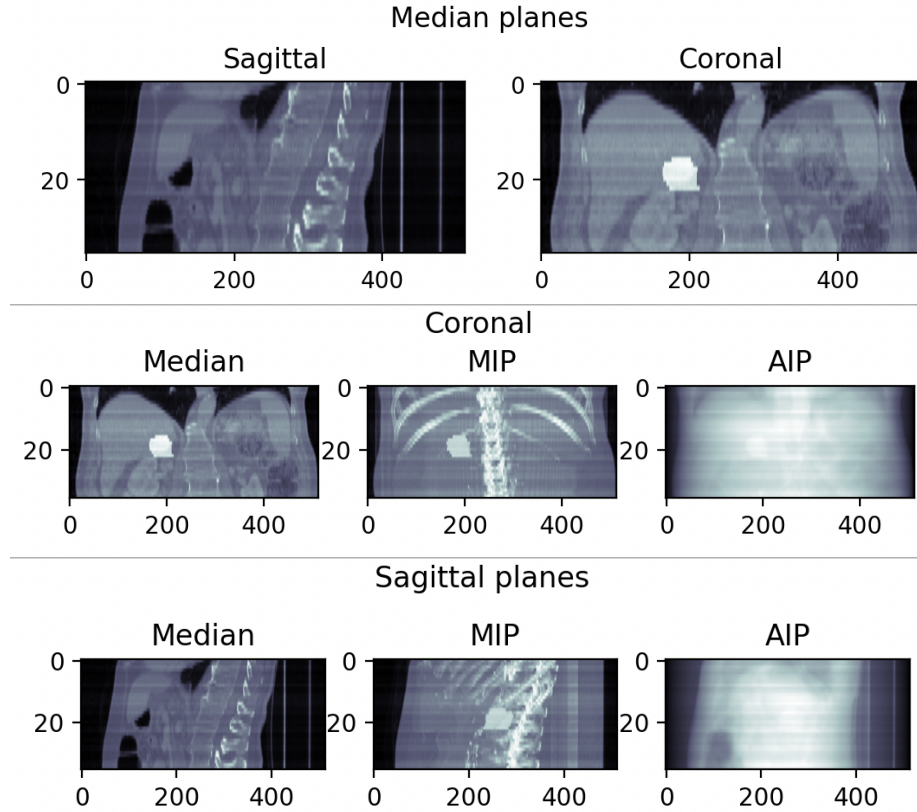


Figure 4: Visualizations of the tumor on median, coronal, and sagittal planes

The coronal and sagittal planes of the loaded DICOM images were examined

to highlight the segmented parts as shown in Figure 4. The coronal plane provided a cross-sectional view of the dataset, showcasing the segmentation regions in the liver, tumor, vessels, and aorta. This visualization effectively displayed the spatial distribution of these segmented areas within the scanned anatomy.

Furthermore, projections of the dataset were generated to gain additional insights. A maximum intensity projection (MIP) was created by projecting the pixel values with the maximum intensity along the axial, sagittal, and coronal planes. This MIP projection offered a comprehensive overview of the dataset, emphasizing areas of high intensity and providing a better understanding of the distribution and extent of the segmented regions. Additionally, an animated gif file was generated by performing axial rotation of the dataset. By rotating the images around the axial axis, a dynamic representation of the dataset was achieved, enabling a comprehensive visualization of the segmented regions from various angles. This animated gif file provided a more interactive and informative way to explore and present the segmentation results.

3.2 3d Rigid Coregistration

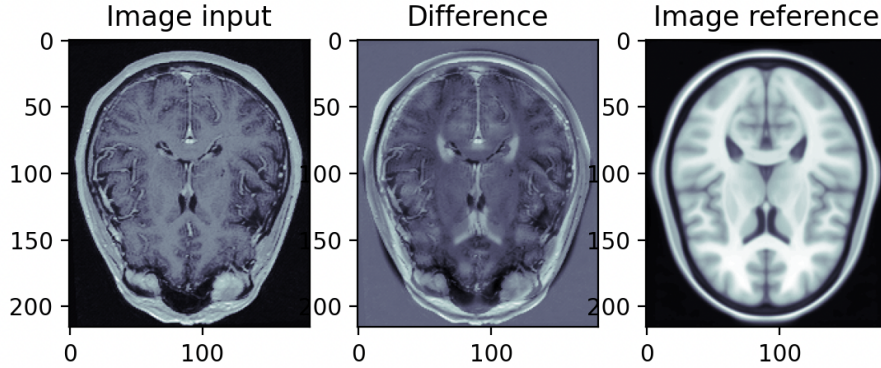


Figure 5: Visualizations of the differences between input and reference image

The coregistered input image was compared to the reference image by visualizing their differences, which involved subtracting one image from the other as shown on Figure 5. This allowed for a comprehensive understanding of the variations and discrepancies between the two images.

The table 1 shows the performance of the implementation. The Mean Absolute Error (MAE) results were computed to quantify the average absolute difference in the pixel intensities between the coregistered input image and the reference image. The obtained MAE value of 13.31 HU (Hounsfield Units) indicates the average magnitude of the differences between corresponding pixels in the two images. A lower MAE value suggests a higher level of similarity

Performance Measure	Value
MAE	13.31 HU
MSE	364.03 HU ²
MI	0.7 bits

Table 1: Performance of the coregistration algorithm

between the images. Furthermore, the Mean Squared Error (MSE) results were calculated to evaluate the average squared difference in pixel intensities between the coregistered input image and the reference image. The MSE value of 364.03 HU² represents the average of the squared differences between corresponding pixels. A lower MSE value indicates a closer match between the two images. Additionally, the mutual information between the coregistered input image and the reference image was measured. Mutual information quantifies the amount of shared information between two images. In this analysis, a mutual information value of 0.7 bits was obtained using (10,10) bins. This indicates the level of similarity or correlation between the images based on their joint histogram distribution. A higher mutual information value implies a stronger relationship between the images.

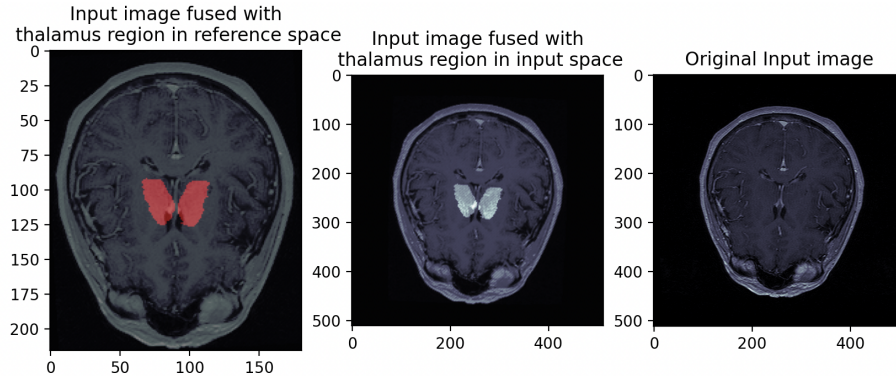


Figure 6: Final result of the implementation

This visualization provided a clear depiction of how the thalamus region was successfully masked and aligned with the input image after coregistration. By combining the input image with the thalamus mask, the specific region of interest was highlighted and visually represented. This allowed for a better understanding of the spatial relationship between the thalamus and the surrounding anatomical structures in the input image space. The visualization served as a visual confirmation of the accuracy and effectiveness of the thalamus region masking procedure, further validating the coregistration and segmentation process.

4 Conclusion

In conclusion, this project highlighted the challenges faced in understanding the DCM files, the long computation time required for coregistration, and initially unclear project objectives. However, these challenges were overcome through persistent efforts and learning from previous lab activities. The intricacies of DCM files were successfully grasped, and the computational efficiency of the coregistration process was improved. The project's objectives were eventually clarified and better understood. As a result, the project contributed to the enhancement of technical skills in medical imaging and provided valuable insights into the complexities associated with working with DICOM data.

Github link for the project: https://github.com/realr3fo/medical_imaging

5 Bibliography

- <https://wiki.cancerimagingarchive.net/pages/viewpage.action?pageId=70230229>
- https://pydicom.github.io/pydicom/stable/auto_examples/index.html
- https://mscipro.github.io/post/read_dicom_files_in_python/