Generation of Fiducial Coordinates from DICOM Data

Group - 3

Overview

- According to given datasets and the cases considered in them, we have proposed an Image Processing algorithm for fiducial localisation.
- There is no change in the algorithm for the given datasets apart from some minor changes in the parameters and operators for better results.
- The algorithm has been implemented in Python and the image processing has been done using skimage and scipy.

Processing a slice

Step #1: Intensity based thresholding

- Intensity threshold was found using Otsu thresholding algorithm.
- This method produces a threshold which minimizes the intra-class variance or equivalently, increases the inter-class variance.
- This helps in foreground-background segmentation.

Original Image

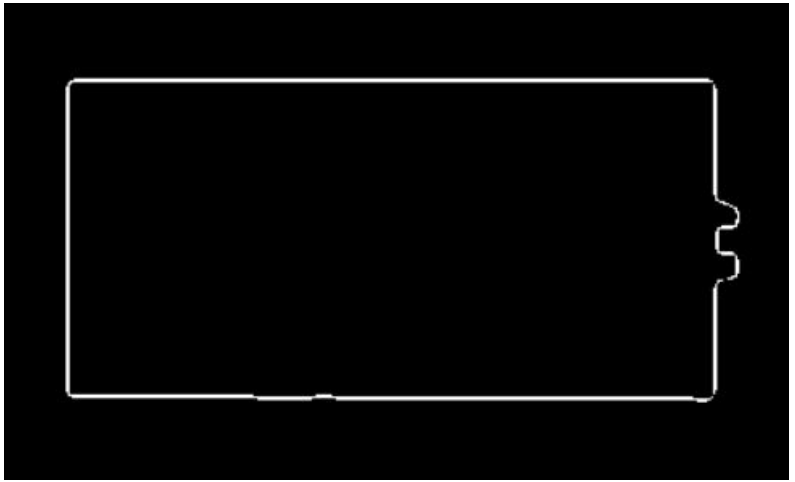
After thresholding

Processing a slice

Step #2 : Edge detection

- We use Canny edge detection algorithm for the purpose.
- This significantly reduces the number of data points that need to be processed further.

Thresholded Image

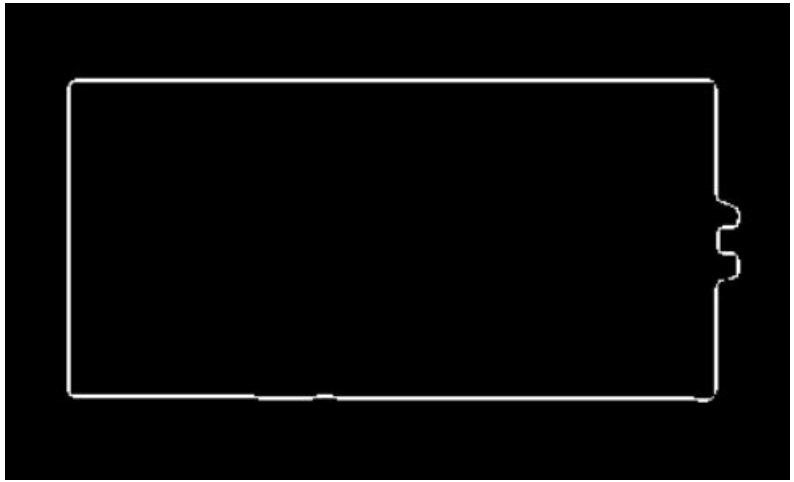


After edge detection

Processing a slice

Step #3: Processing the edges found

- A binary closing operator is first applied to remove small gaps in the edge, if any.
- Morphological skeletonize operator then turns the edges into 1 pixel width.



After edge detection

After edge processing

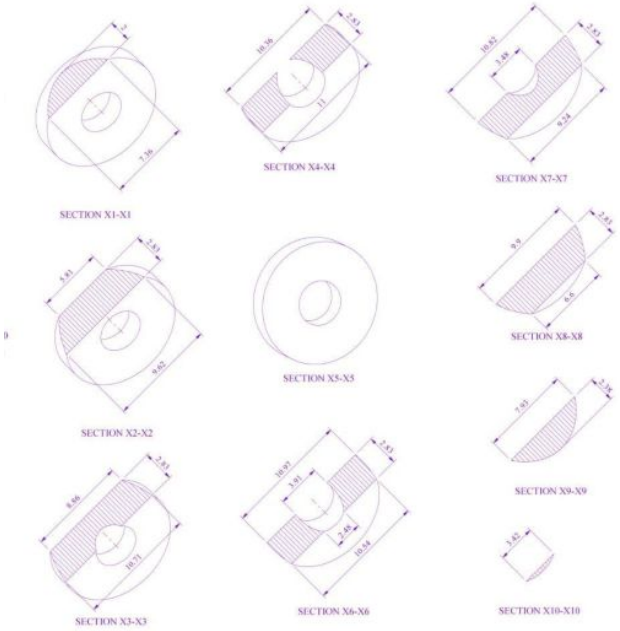
Processing a slice

Step #4: Corner detection

- Why to detect corners?

A plane cuts the fiducial in these ways.

Thus, detecting the corners would be a good way to localise the important points which can help in localising the centre.



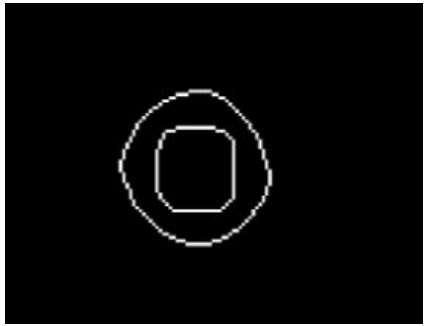
Processing a slice

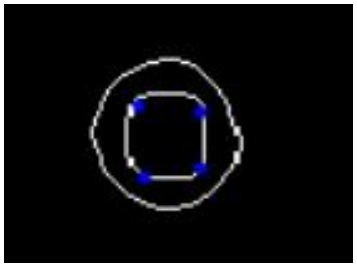
Step #4: Corner Detection

- Harris corner detection algorithm is used as the base algorithm for corner detection.
- The detected corners are filtered by peak corner detection which keeps only the corners which represent peaks.

Image with edge of 1px width

After corner detection





Interestingly, corner detection also works for circular cross-section of fiducial marker.

Processing a slice

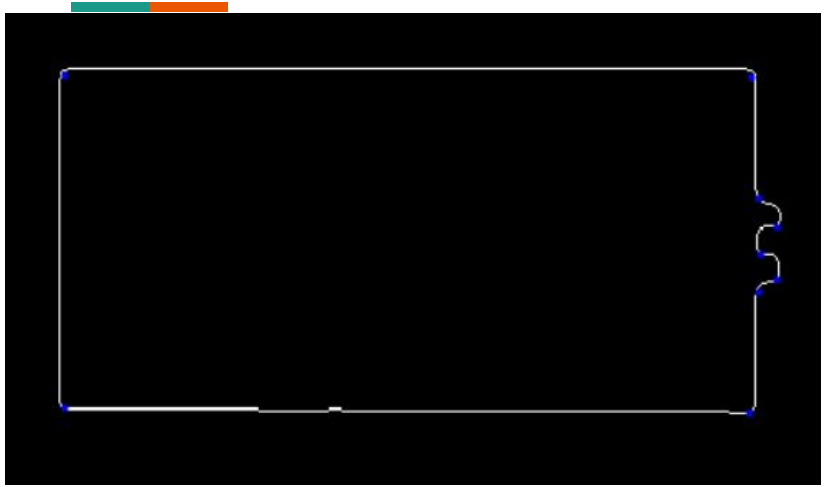
Step #5: Filter unwanted corners using distance.

- Not all of the corners obtained are from fiducials. But, wherever a fiducial is present, two corners are detected which have distance less than or equal to outer diameter of a fiducial.
- So, only those corners which have distance less than outer diameter of a fiducial are kept and the rest are filtered out.

Processing a slice

Step #5: Filter unwanted corners using distance

- Sometimes, fiducials are placed close to geometrical corners of an object. In that case, the corner detected at that geometrical corner is not filtered out.
- So, we detect the geometrical centres using a base image and all the detected corners at the geometrical corners are filtered.



Unfiltered corners

Corners filtered by distance

Processing a slice

Step #6: Cluster the final corners for fiducial localisation in a 2D slice.

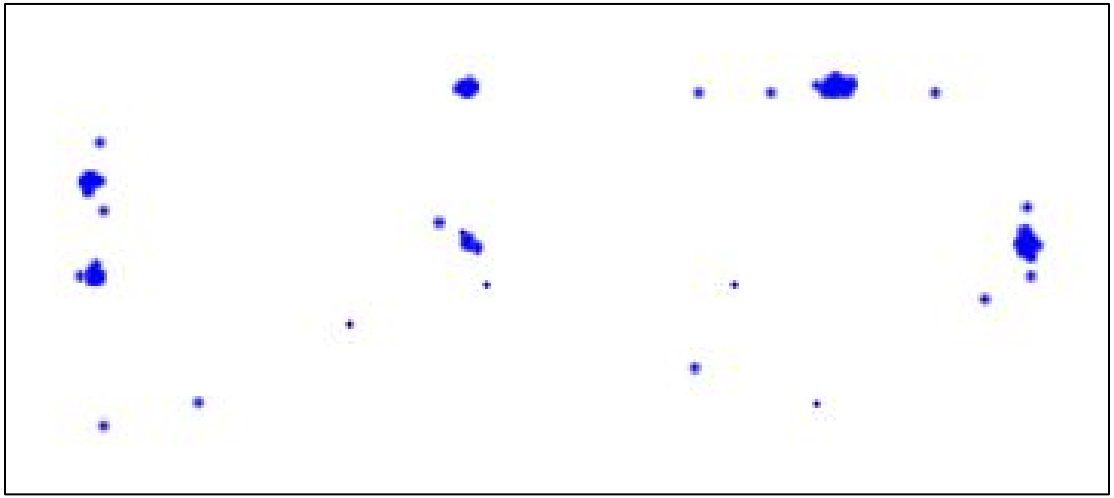
- We use hierarchical clustering method based on distance to form clusters of corners which belong to the same fiducial.
- Then, we find the coordinates of centre point for each cluster to localise the centre of the fiducial.

Filtered corners

After clustering and locating centres of clusters

Processing the fiducials

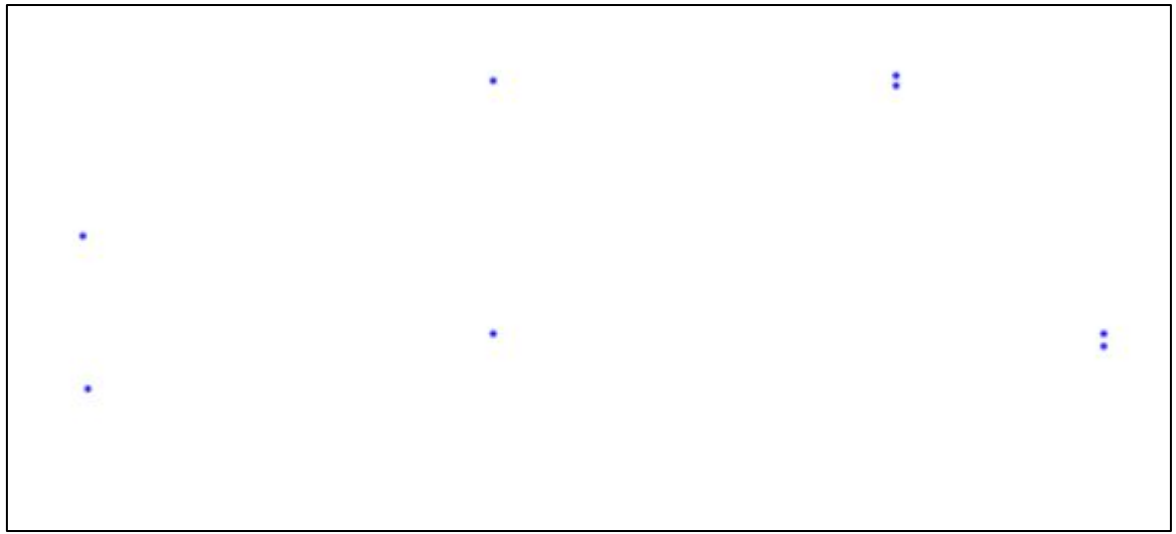
- Fiducials obtained in each slice are stored in an array.
- A fiducial is detected multiple times. This happens because a fiducial is localised in a continuous set of slices.
- So, when these fiducials are plotted, they form definite clusters with some noises.



Different clusters are formed and there are noises

Processing the fiducials

- These clusters are separated from each other and the noise using DBSCAN (Density-based spatial clustering of applications with noise) algorithm.
- DBSCAN algorithm has a special feature of detecting the noise and clusters using the minimum number of points that compose a cluster and maximum distance between two points to be placed in a cluster.
- Finally, the coordinates of the cluster centres are determined, the noises are removed.



After DBSCAN algorithm and locating centres of fiducials. Notice that, noises have been removed.

Determining the third coordinate

Approach #1

- DICOM slices are arranged according to third coordinate. So, we can extract the third coordinate if we know the spacing between the two slices.
- So, the third coordinate is found using the index of the slice, total number of slices and pixel spacing in along that third axis (obtained from metadata).
- This is done at the same time when other coordinates of the fiducials are localised in a slice.

Determining the third coordinate

Approach #2:

- We find the 2D coordinates of the fiducials along the Axial, Sagittal and Coronal planes.
- Now, each triplet of 2D coordinates along the three planes are taken. Three duplets (formed by coordinate in the same axis from each triplet) are matched with some minimal error.
- If all of the three duplets match, we find a fiducial at the point.

Determining the third coordinate

- Our initial model used approach #2 as it reduces the chances of errors to creep in.
- But, as per the problem parameters, we were supposed to use DICOM images along only one plane for different datasets. So, we have produced the results obtained by approach #1 in the report submitted.

Future Work

- The PVC Skull data that was given to us had an incomplete outer contours for most of the images whereas CT/MRI scans of actual brain have full outer contours.
- With the DICOM data of real brain, we plan to adopt an algorithm for outer contour detection which will remove all possible noisy points which can arise due to inside structure of brain.
- This will improve the accuracy of our algorithm.