

Tumor segmentation and isolation analysis



School of Computer Science and Engineering

November – 2018

**Author - Agrim Nautiyal, School of Computer Science and Engineering, Vellore
Institute of Technology, Vellore.**

Abstract

Image processing is a vast area with numerous applications in our day to day lives. The main motive of image processing is to prepare images obtained from image sensors for further analysis and ensure that operations on these said images can occur accurately. The techniques of image processing vary from different use cases depending upon the needs of the person employing the same. In this project we demonstrate the use of Image Processing in health care by taking inputs of MRI scans and locating the presence of any tumor in them if the same exists.

The use of such a project is to create a new algorithm that can help in tumor segmentation

and enable a software to intelligently predict based on the automated isolation of the tumor, whether the tumor is benign or malign in nature. Upon accurate prediction a lot of human intervention and effort can be reduced as the process of segmentation with prediction becomes automated. To make the software more reliable we need to use as many training data as possible. Such a system would drastically reduce the scope of human error and prevent misdiagnosis and help in preventing wrong treatment or not being able to diagnose a serious issue like cancer in time.

Through this project a proof of concept is tried to be provided according to which an example of an MRI scan is taken and worked upon. Since each image is different, to make the custom algorithm proposed in this report work, some techniques of image normalization and standardization will have to be employed to make the operations relevant to the local context of pixels of the particular use case. The final outcome of this project is to segment the tumor area from the scan of a body part and isolate it. Then apply the trapezium method to find area via selective integration and the perimeter respectively. As a part of the future work the scan image, area and perimeter data can be stored in a file and used as training data that will be fed to a model. This model can make predictions upon the nature of the segmented tumor depending on the area and perimeter it calculates.

Such a project can have long lasting medical benefits and completely revolutionize and automate the healthcare industry.

Problem statement -

An efficient and innovative segmentation approach for brain tumor detection in MRI

Problem definition -

Segmentation in image processing terms is defined as the process of separating out an area of interest from a given image and acquiring meaningful information out of it. The problem statement asks us to make critical analysis of a patients' condition from their MRI scan of a body part of concerned. Depending on how accurate our isolation and segmentation is, we will be able to determine the condition accordingly.

In this era of increased computational performance and ease of human-computer interaction, it has become mandatory than ever for students, research scholars and computer engineers to come up with more innovative and accurate algorithms for specific use cases.

Motivation -

Having personally worked with Artificial, Convolutional and Recurrent Neural Networks; understanding the concepts of image processing and applying it on given images piques my interest.

All networks perform only as well as their input has been preprocessed. Hence it is vital to understand how best to process any image to overcome some general motives like

- Reducing noise
- Enhancing Quality of image
- Controlling brightness of image as per use case
- Enhancing to view higher detailing
- Normalizing image pixel values

And some user specific motives like

- Edge detection
- Segmentation
- Image recognition
- Object detection
- Image inversion
- Object elimination

Along with understanding the above elementary yet important components of image processing, I have a strong interest in innovating and coming up with new algorithms to perform some tasks like edge detection of area of interest, segmentation, enhancement etc. Finally the main objective is to combine all the above, encapsulate it into one project and come up with an industry relevant solution to a problem statement such as this.

Project description -

This project aims to isolate and segment a tumor highlighted in a patients MRI scan. Calculating the tumor's area and perimeter remains an important objective of this project.

Rather than strictly following a research paper or an already published methodology, the project supports a new approach to edge detection. However, this edge detection is not based on any known method and is applicable for specific use cases. In our case, it isolates just the tumor (lit) area in the MRI scan to a good accuracy and highlights only its presence in the correct location of the image, blackening everything out.

This same approach can be used to carry out segmentation and finally with the activated pixels we can calculate values of perimeter and area in terms of pixel units for the particular scan used and eventually apply to other scans as well (after resizing and normalizing).

This project uses only the skimage python library to input and output images. All the other functions and implementations are hard coded and based on raw logic. None of the implementations are used/published and remain completely novel as per the authors work.

Objectives -

This project has the following objectives

To apply noise cancellation filters - This can be done by the use of the any normal average or median filter. This is an important aspect of image processing as this reduces the probability of false analysis and in turn ensures that images do not contain any false information encoded in its pixels. This step smoothens the image and helps in bringing out the gross details of its components.

To make contrast enhancements - Contrast enhancement here is done either through normal contrast stretching (threshold method) or by applying the hilbert space in 2D gray level images. The reason why the second method is more preferable is because the second method caters to local context of image pixels.

The Hilbert space of the gray level images

Let f and g be two integrable functions from $F(D,E)$. We define the scalar product by: for $\forall f_1, f_2 \in F(D,E)$

$$(f_1 | f_2)_{L^2(E)} = \int_D (f_1(x,y) | f_2(x,y))_E dx dy$$

With the scalar product the gray level images space $F(D,E)$ becomes a Hilbert space.

Further on, we will define the norm.

$$\forall f \in F(D,E), \|f\|_{L^2(E)} = \left(\int_D \|f(x,y)\|_E^2 dx dy \right)^{\frac{1}{2}}$$

The relative contrast between two pixels

The relative contrast between two distinct pixels $p_1, p_2 \in D$, for an image, $f: D \rightarrow E$ is a logarithmic gray levels image noted $C_R(p_1, p_2)$, defined by relation:

$$\forall p_1, p_2 \in D, p_1 \neq p_2,$$

$$C_R(p_1, p_2) = \frac{1}{d(p_1, p_2)} \langle \times \rangle \frac{f(p_1) - f(p_2)}{1 - \frac{f(p_1) \cdot f(p_2)}{M^2}}$$

where $d(p_1, p_2)$ is the Euclidean distance between p_1 and p_2 in the R^2 plain.

To sharpen produced image with appropriate selection of filter - This objective is fulfilled by a simple application of the laplacian sharpening filter where a high boost mask is used to bring out the finer details of the image after contrast enhancement. The reason why this works the best is because it doesn't need a lot of fine tuning and the gross details are already out in the first step of removal of noise by median and average filters.

The Laplacian $L(x,y)$ of an image with pixel intensity values $I(x,y)$ is given by:

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

This can be calculated using a convolution filter.

Two commonly used discrete approximations to the Laplacian filter.

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

Because these kernels are approximating a second derivative measurement on the image, they are very sensitive to noise. To counter this, the image is often Gaussian smoothed before applying the Laplacian filter. This pre-processing step reduces the high frequency noise components prior to the differentiation step.

In fact, since the convolution operation is associative, we can convolve the Gaussian smoothing filter with the Laplacian filter first of all, and then convolve this hybrid filter with the image to achieve the required result. Doing things this way has two advantages:

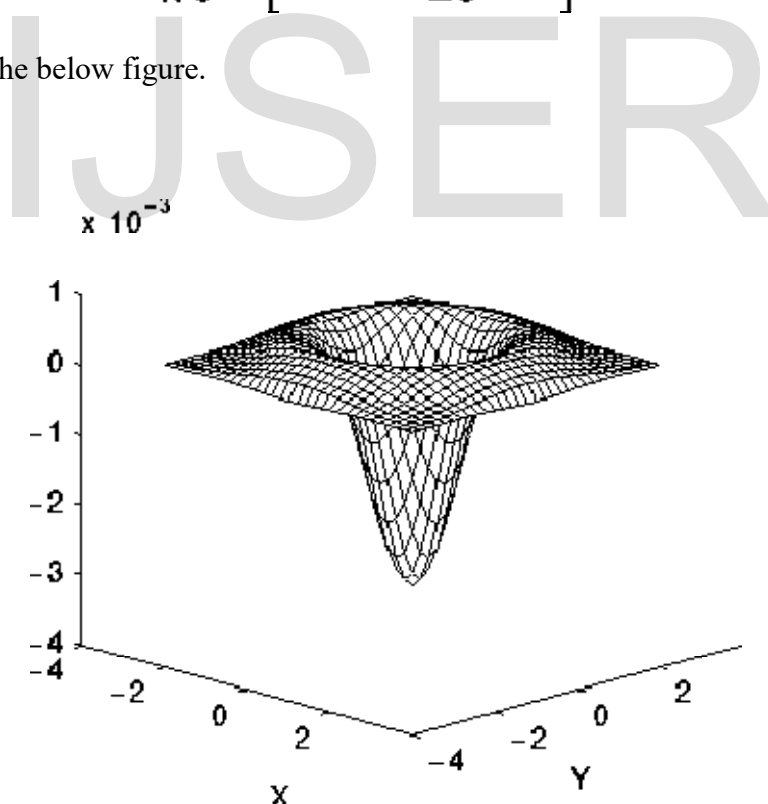
Since both the Gaussian and the Laplacian kernels are usually much smaller than the image, this method usually requires far fewer arithmetic operations.

The LoG ('Laplacian of Gaussian') kernel can be precalculated in advance so only one convolution needs to be performed at run-time on the image.

The 2-D LoG function centered on zero and with Gaussian standard deviation σ has the form:

$$LoG(x, y) = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

and is shown in the below figure.



To apply right algorithm steps to isolate and segment the tumor area - The algorithm employed to segment the tumor area is made such that it first removes the unnecessary details of the image like brain coverings, fluid space and finer granules within the brain area. The biggest reason is that the final outcome depends on how much finer details were eliminated to bring about the tumor (i.e; the activated pixels and hence uncover the area and perimeter).

To find perimeter of tumor area in pixel units - The perimeter of the tumor area can be easily found out by adding up the number of activated pixels with a certain threshold intensity and multiplying with a constant for better generalization.

To find area of tumor in pixel unit -

The definite integral of $f(x)$ is equal to the (net) area under the curve $y=f(x)$ over the interval $[a,b]$. Riemann sums approximate definite integrals by using sums of rectangles to approximate the area.

The trapezoid rule gives a better approximation of a definite integral by summing the areas of the trapezoids connecting the points

$$(x_{i-1}, 0), (x_i, 0), (x_{i-1}, f(x_{i-1})), (x_i, f(x_i))$$

for each subinterval $[x_{i-1}, x_i]$ of a partition. Note that the area of each trapezoid is

$$(x_i - x_{i-1})f(x_{i-1}) + \frac{1}{2}(x_i - x_{i-1})(f(x_i) - f(x_{i-1})) = \frac{1}{2}(f(x_i) + f(x_{i-1}))(x_i - x_{i-1})$$

For example, we can use a single trapezoid to approximate:

$$\int_0^1 e^{-x^2} dx$$

The final area is calculated under the curve by multiplying the horizontal as well as the vertical waves (in directions of X and Y axis by taking differentials along Y and X axes respectively and integrating them over a loop).

Along with these objectives we can set some long-term objectives that need not stay within the scope of this project

- To test the above algorithms on different MRI scans

- To create a dataset of MRI scans, respective tumour areas and perimeters and classification (either malign or benign)
- To create a predictive model that will predict if a tumor is benign or malign depending on the MRI inputted

Methodologies/Algorithm

The execution of this project occurs in the given below process flow

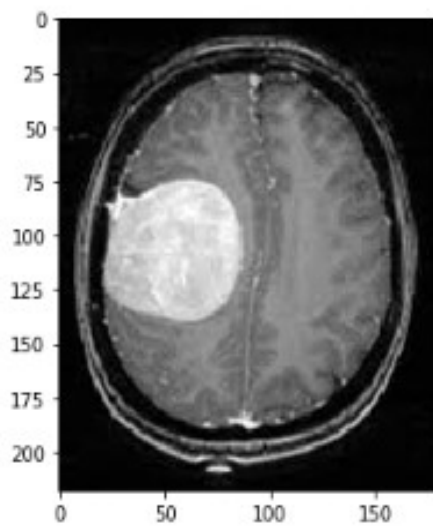
- Input image
- Add contrast enhancements
- Apply a high boost sharpening filter (Derived from second order laplacian filter)
- Apply sobel filter on the inputted image
- Smoothen sobel filtered image by means of a 3x3 average/median filter
- Multiply the obtained smoothened image with the sharpened image produced in step 3. Store as mask
- Add the above mask obtained to the originally inputted image. Store as temp_sharp
- Apply average filter on temp_sharp repeatedly for 6 times (Value obtained by trial and error). Store as t1
- $\text{test_res1} = \text{original_picture} + \text{t1} * \text{sharpened_image}$
- $\text{test_res2} = -1 * (\text{test_res1})$
- $\text{new_result} = \text{test_res1} * \text{test_res2}$
- $\text{final_result} = \text{inverse}(\text{new_result})$
- The final_result obtained is the array representation of the final image we get that contains the isolated tumor area depicted by a thin black edge.

Additional steps :

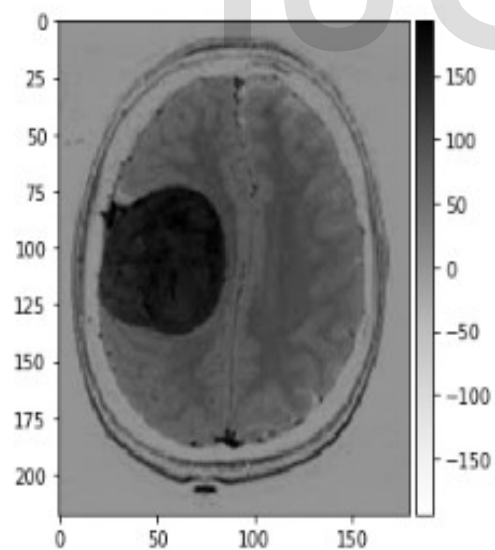
- Enhance final_result , edges get better defined
- Calculate no of activated pixels - store as perimeter
- Activate area under enclosed region - segmentation
- Calculate no of activated pixels, store as area
- Store data for MRI scan locally

Snap shots of solved objectives/components of process flow

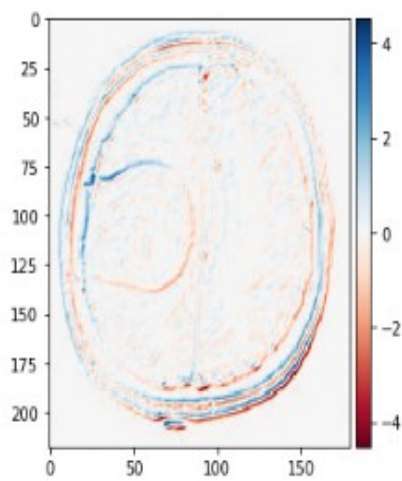
Input Image (A)



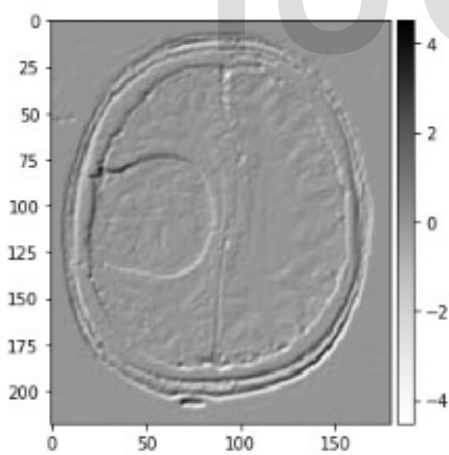
High-boost filtered + Inverse (B)



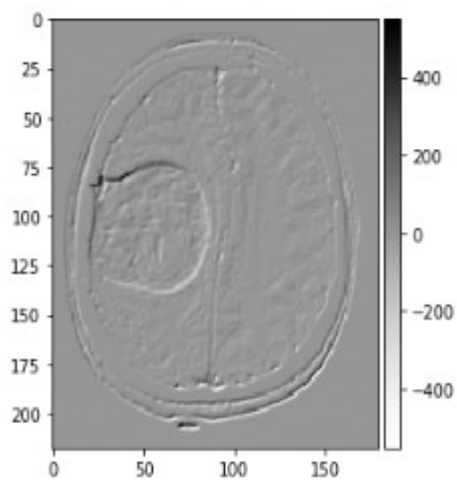
Sobel filtered image on input (C)



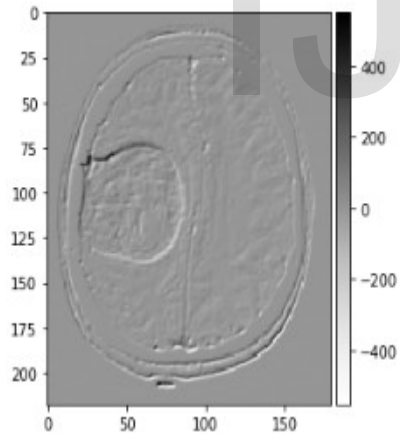
Mask 1



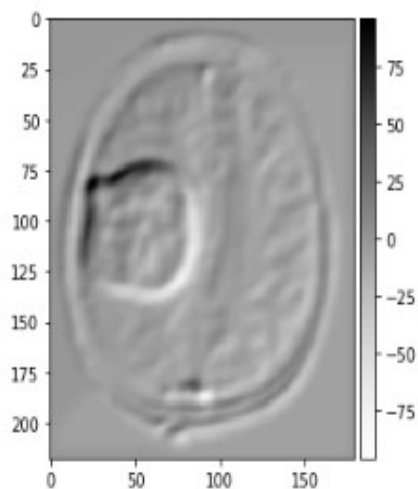
Mask 2



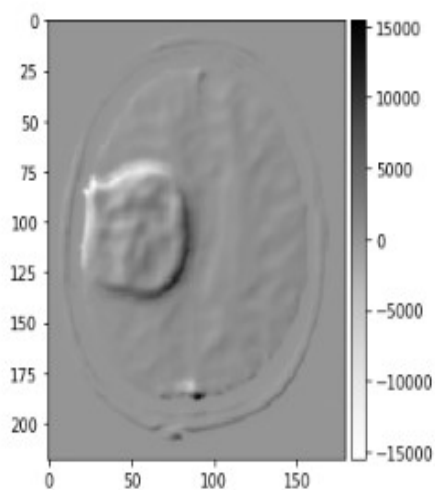
Mask 3



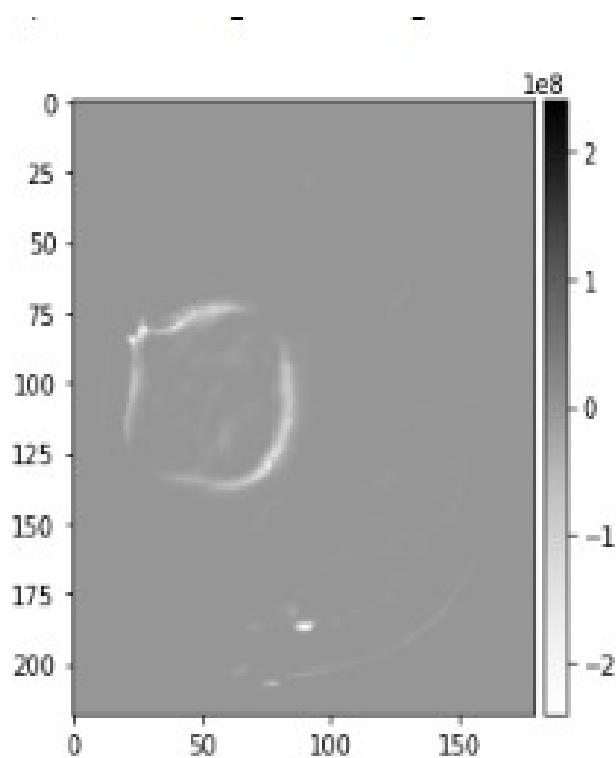
Average filtered image - Making the lower part of tumor area more prominent (D) - notice the lower right portion of tumor edge



Inverted image to make upper part of tumor more prominent (E)- notice the upper left portion of tumor edge

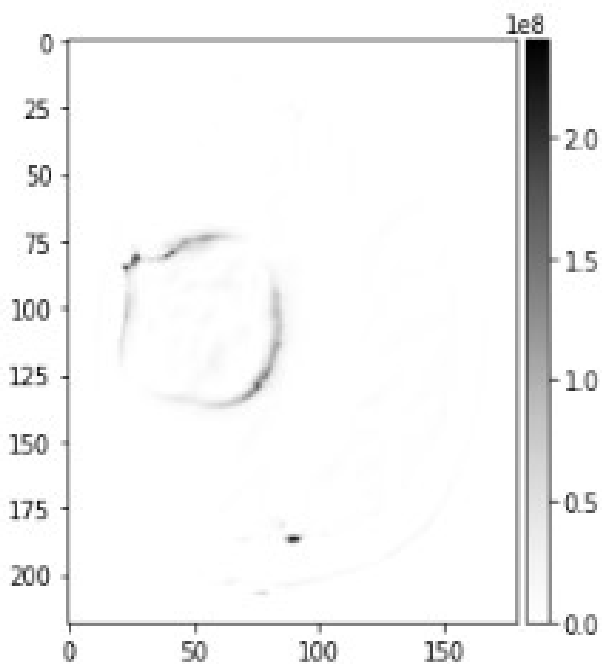


Multiplying the above two prominent areas with each other- Tumor edge (F)

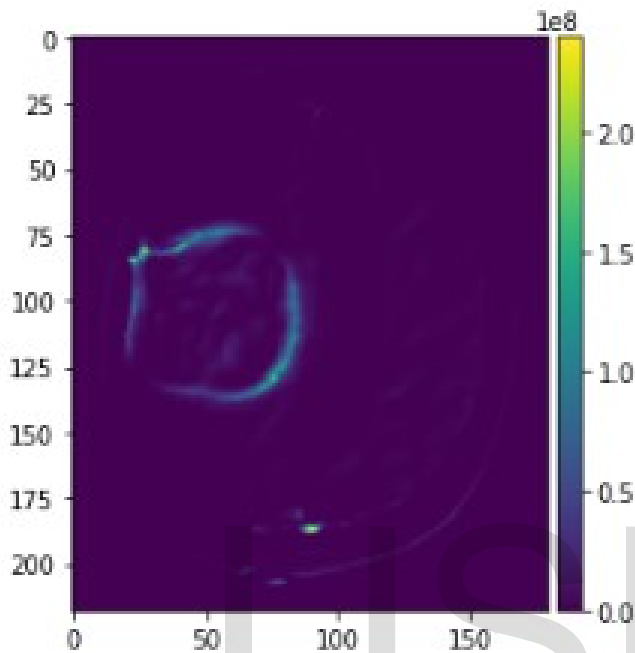


IJSER

Inverting obtained result (G)



High boost filter + heat map :



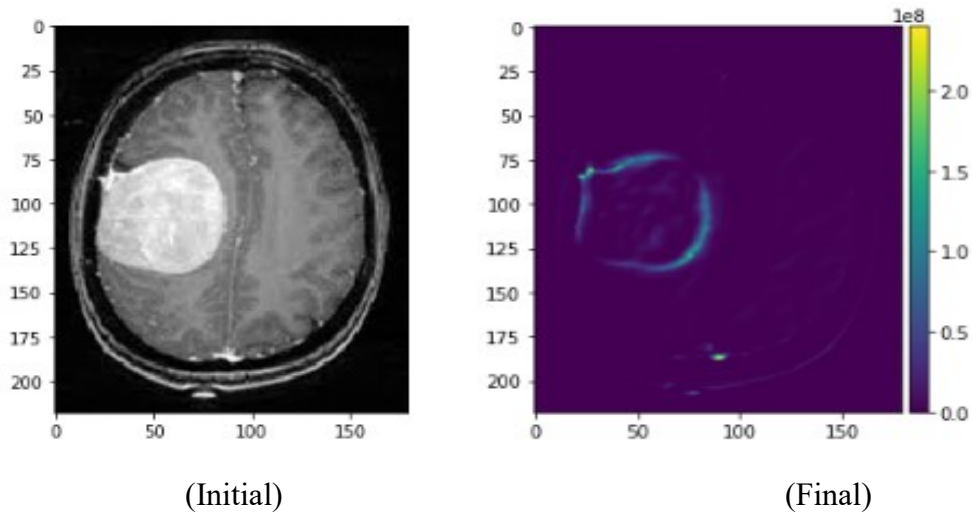
Solved objectives :

As depicted by the above snapshots/pictures ; the list of solved objectives are as follows

- To apply noise cancellation filters
- To make contrast enhancements
- To sharpen produced image with appropriate selection of filter
- To apply right algorithm steps to isolate tumor area
- To calculate perimeter of isolated tumor
- To segment tumor area
- To calculate area of tumor in terms of pixel units

Results

Until now the results of this project can be described as follows



Particular use case result (Area of tumor in the input image) :

43635 Pixel Units

Implementation :

```
In [116]: from numpy import trapz

dx=0
for i in range(1,len(res2[0])):
    diff = abs(res2[0][i]-res2[0][i-1])
    dx+=diff
area1 = trapz(res2[0], dx=dx/180)

dy=0
for i in range(1,len(res2[1])):
    diff = abs(res2[1][i]-res2[1][i-1])
    dy+=diff
area2= trapz(res2[1], dx=dy/180)
final_area = area1*area2

In [83]: final_area

Out[83]: 43634.81866929069
```

Conclusion

This project has been successful in implementing a new algorithm to extract a tumor area by identifying its edge. It also will be the forefront in crafting the algorithm for segmentation.

Having achieved the same, we can now branch out to other applications/use cases of this demonstration. We can use this project to calculate area, perimeter and classify each tumor detected in an inputted MRI scan.

This will prove to be an industrially relevant solution and hence promote the use of smart systems and in turn cause greater computational development.

References

Padhani AR, Husband JE. Dynamic contrast-enhanced MRI studies in oncology with an emphasis on quantification, validation and human studies. Clin Radiol 2001; 56: 607–620.

Knopp MV, Giesel FL, Marcos H, von Tengg-Kobligk H, Choyke P. Dynamic contrast-enhanced magnetic resonance imaging in oncology. Top Magn Reson Imaging 2001; 12: 301–308.

F. Mai, Y. Hung, H. Zhong, and W. Sze. “A hierarchical approach for fast and robust ellipse extraction,” Pattern Recognition, Vol. 41, No. 8, pp.2512–2524, August 2008.

Sarma Mousmita and Kandarpa Kumar, "Segmentation of Assamese phonemes using SOM," NCETACS, pp.121-125, March 2012J. Breckling, Ed., The Analysis of Directional Time Series: Applications to Wind Speed and Direction, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61

Resource of MRI scan images used in this project -

<https://blogs.mathworks.com/community/files/Brain-MRI-Tumor-Detection-and-Classification.jpg>