Wesley Chen, Brandon Sim

Applied Math 120 Final Project

Introduction

The MRI (Magnetic Resonance Imaging) is one of the most accurate and available types of medical imaging that is used to detect diseases throughout the body, ranging from severe bleeding to cancerous growths (SOURCE A). The prevalence of MRI imaging as a diagnosis is well represented by the sheer number of the expensive yet critical equipment. In the United States, there are over 10,000 MRI machines (SOURCE B). When a doctor administers an MRI imaging (we will use the brain as the organ of study from this point onwards), the machine stores many 2D slices of the brain from multiple angles to provide the most accurate representation of the brain. From this collection of slices, the doctor must look through each frame for possible irregularities that could confirm the existence of a tumor. The person reading the MRI image must flip through each frame systematically – there is no shortcut to knowing about where a tumor could be. The requirement of such a methodical but tedious search inspired us to develop approaches to automate the reading of a set of MRI images. For the scope of this project, we explored a couple independent techniques to identify possible tumor candidates from a given 2D image (one slice of the brain). At the bare minimum, the techniques we explored could reduce and simplify the number of images that a doctor would have to peruse in search of a possible growth.

Method

We were given a complete data set of a MRI brain scan for one patient with a relatively small tumor, courtesy of Dr. Steven Shufflebeam of Massachusetts General Hospital. Using a small tumor as our test case, we believe that larger tumors will only be more obvious to identify.

We tried two independent approaches: the first was using watershed segmentation to filter various foreground objects – hoping to identify the tumor as one of the fewer remaining objects. A brain tumor usually appears as a denser, white area which should be filtered out by the segmentation algorithm

The second method was to simplify the MRI image through edge detection algorithms. Theoretically, a tumor would generate a closed object in our image. With this abstract picture, we wanted to use the natural symmetry of the brain to detect the tumor. A tumor would be an irregular growth and would not have a symmetric counterpart.

Watershed Segmentation

The goal of the watershed segmentation method is to map the image to a topological equivalent, and “flood out” various “basins” to represent different levels of foreground. The first step would be to convert to grayscale. The intensity of the grayscale code would be mapped to relative height. Conceptually, once this grayscale mapping is complete, the “whitest” pixels would represent areas of maximum height (or intensity or density in the case of an MRI image). Using a gradient magnitude should result in proper segmentation. However, due to small differences in gradient, this results in an overly segmented product (see results section below).

To eliminate such detail, a set of image erosion and dilations must be conducted to eliminate some foreground detail. The foreground is selected through a morphological structuring element provided by the strel() function of Matlab’s image processing toolkit. We chose to use “disk” shaped elements as the round object best represents the structures we would find in the brain. We then calibrated the size of this element and this is where we controlled the level of segmentation. This element is used by the image manipulation functions below.

After this element is set, we used other Matlab image processing functions (namely imerode() and imdilate() ) to help remove foreground detail and provide a map of topological height “most prominent foreground areas.” Imerode() looks at every subset of the disk shaped element within the full image matrix to find areas with very similarly colored pixels – and combines them into one uniform value. Imdilate() compliments this step by closing off what was eroded so that the eroded area is filled as it was before – within the same structuring element.

After the establishment of the foreground markers, the markers were superimposed over the original image. Another sequence of erosion-dilation was applied to help smooth out the foreground markers superimposed onto the original image. Then, a grayscale thresholding function was run to select the background areas. Then, the watershed() function was called to create ridges (the valleys of the basin area – also our background). This background marker image was then superimposed onto the image with the foreground marker to create the fully marked image. In our MRI case, because the background is uniformly black, this step was not necessary but was kept in our program for comprehensiveness (will still work on any image).

Finally, we visualized the result by using a colored watershed visualization which filled each object with a different color. This color map was then superimposed with a 50% transparency over our original image. Then, each color would be a different component to identify as a possible tumor candidate.

Edge detection and Symmetry

The first step to edge detection involved picking an edge detection algorithm. We found various edge detection algorithms including Canny and Sorbel methods. Canny is generally considered to be the more optimal algorithm. It uses hystresis tresholding over a fixed threshold and this allows the algorithm to be more robust to noise, provide thinner edge lines and avoid treating 1 thicker edge as 2 separate ones (SOURCE C).

Mention smoothing, thresholds

Region functions like the ellipse things

CENTROID,

The dude’s edge list library – CITE BELOW

Results

For both cases, we started with this 2D MRI snapshot <original.jpg>. Because the watershed segmentation is more powerful and can handle more complex pictures, we will also show the output for a given colored image, this lily pad landscape <lilypad.jpg>.

Watershed Segmentation

(IM VISUALIZING TWO COLUMNS, ONE FOR THE MRI BRAIN IMAGES, ONE FOR THE LILYPAD SEQUENCE

ALL OF THE LILYPAD PICTURES ARE IN THE WATERSHEDOUT FOLDER WHEREAS ALL OF THE MRI PICTURES ARE IN THE MRIOUT FOLDER)

MAYBE THE COUPLE OF SENTENCES ABOVE EACH SET OF IMAGE COULD BE WRITTEN AS A CAPTION INSTEAD – CHECK OUT HOW IT LOOKS

Our results are best represented in a stepwise process – showing the result of every image after the image transformation.

First, we started with the most conceptual approach – calculating the gradient and then the watershed directly. You will notice the over-segmentation caused by using an unmodified image.

(No captions needed as these images have the “figure” style)

<MRI\_Gradient\_Mag.jpg> <Gradient\_Mag.jpg>

<MRI\_Overseg\_Watershed.jpg> <Overseg\_Watershed.jpg>

To solve oversegmentation, we needed to erode and dilate the image. Here we show the results of two pairs of functions: imopen() followed by imclose() and imerode() followed by imdilate(). We noticed that the erode-dilate pair preserved the original edge and shape of the images over the open-close combination.

<MRI\_Open\_Disk4.jpg> <Open\_Disk6.jpg>

Cap: imopen() with 4-pixel disk structuring element imopen() with 6-pixel disk structuring element

<MRI\_Close\_Disk4.jpg> <Close\_Disk6.jpg>

Cap: imclose() with 4-pixel disk structuring element Imclose with 6-pixel disk structuring element

<MRI\_OpenReconstruct\_Disk4.jpg> <Open\_Reconstruct\_Disk6.jpg>

Cap: imerode() with 4-pixel disk structuring element imerode() with 6-pixel disk structuring element

<MRI\_CloseReconstruct\_Disk4.jpg> <CloseReconstruct\_Disk6.jpg>

Cap: imdilate() with 4-pixel disk structuring element imdilate() with 6-pixel disk structuring element

From the reconstruction of the image, a maxima was calculated and then superimposed on the original image to mark the foreground.

<MRI\_Maxima.jpg> <Maxima.jpg>

Cap: Calculated maxima from the erode-dilation pair

<MRI\_MaximaSuperimposed.jpg> <MaximaSuperimposed.jpg>

The superimposed image was the processed with a erosion-dilation to smooth out the markers.

<MRI\_MaximaSuperSmooth.jpg> <MaximaSuperSmooth.jpg>

To detect the background, a grayscale threshold was applied to discriminate.

<MRI\_Background\_Threshold.jpg> <Background\_Threshold.jpg>

The background was reduced to its contour, which would be the location of the background, as well as serve as the lowest points for the watershed.

<MRI\_Watershed\_Ridge\_Lines.jpg> <Watershed\_Ridge\_Lines.jpg>

The background markers were superimposed along with foreground markers on the original image.

<MRI\_All\_Markers\_Super.jpg> <All\_Markers\_Super.jpg>

The image can now be properly segmented which can be visualized by colors.

<MRI\_Color\_Watershed.jpg> <Color\_Watershed.jpg>

Overlaying the above colored watershed image with 50% transparency helps orient in relation to original image giving our final result.

<MRI\_result.jpg> <result.jpg>

Edge Detection and Symmetry

YOUR RESULTS GO HERE

Discussion

Our results show promising images that a more refined automated algorithm could be designed. We recognize that individually, both algorithms need to be very well calibrated to give us the desired best output. In general, the watershed segmentation method has a tendency to provide false positives due to the fact that the tumor is not the absolute brightest structure recorded by the MRI. It would be difficult to “teach” our algorithm to ignore such areas – because sometimes, the tumor could in fact be the whitest area. However, this problem can be reduced by running the algorithm at various thresholds for the morphological structure element used to simplify the image. Regardless, providing false positives is more acceptable than a false negative – as a doctor would be able to quickly filter through the highlighted possibilities for tumor candidates.

The edge detection algorithm requires a similar calibration step. The same process to test various thresholds for repeated results would help reduce these errors. Because of the simplicity of this design, this algorithm seems to be favored with our current results. However, with border cases, the edge detection may create false negatives (if the tumor is bordering another edge).

Because the algorithms are different in their accuracies, successive application of both would only improve our accuracy. Through such cross-checking, the agreed parts would have the highest probability of being the area of interest. Combining both methods would increase our detection rate while minimizing false positives.

Future Work and Conclusion

Our project has demonstrated two possible independent directions to the automation of MRI reading. We believe that these two algorithms will work well with each other and when coordinated, will provide the best results. Possible next steps include coordinating between the two algorithms. This could be done by storing the coordinates of each “identified” section. Once this is accomplished, we will have developed an accurate method to pick up irregular growths in a brain for any given 2D MRI image. Then, we would need to read multiple slices and process them together. This would give us the tumor located in 3D space isolated from a collection of 2D slices. To perform these tasks, we would need to run the 2D algorithm on every slice and then store the results in a 3D matrix represent the position of possible tumor candidates. Finally, a check could be done by running the same 3D algorithm from all the different angles that the MRI takes. This would conclude in a fully-functional, automated MRI interpreter. Even if it is unable to conclusively diagnosis a tumor, this program could at least reduce the number of frames that a doctor would have to sift through to identify a possible tumor.

MENTION A POSSIBLE SCORING SYSTEM IN INREPRETATING THE 3D IDENTIFICATION MODEL  
AND PLEASE ADD IN REFRENCES WITH FOOTNOTES THEY”RE NOTED AS (SOURCE A, B,C) etc.

ADD IN SOME MORE MATH FOR ALL THE COMPONENTS

ORGANIZE INTO POWERPOINT

PUT IN PICTURES DICTATED BY <FILE NAME HERE>

References

<http://www.mathworks.com/products/image/examples.html?file=/products/demos/shipping/images/ipexwatershed.html#2>

SOURCE A <http://www.medicinenet.com/mri_scan/article.htm>

SOURCE B <http://www.magnetic-resonance.org/MagRes%20Chapters/21_03.htm>

SOURCE C <http://12000.org/my_courses/UCI_COURSES/CREDIT_COURSES/fall_2004/EECS_207A_UCI_FALL_2004/my_notes/nabbasi_EECS_207A_notes.pdf>