Scholars' Mine

Masters Theses   Student Theses and Dissertations

Spring 2015

Computer aided diagnosis of oral cancer: Using time-step CT images

Jonathan T. Scott

Follow this and additional works at: https://scholarsmine.mst.edu/masters_theses

Part of the Bioimaging and Biomedical Optics Commons, Electrical and Computer Engineering Commons, Nuclear Engineering Commons, and the Radiology Commons

Department:

COMPUTER AIDED DIAGNOSIS OF ORAL CANCER:

USING TIME-STEP CT IMAGES

by

JONATHAN T. SCOTT

A THESIS

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE IN NUCLEAR ENGINEERING

2015

Approved by

Dr. Hyoung Koo Lee, Advisor
Dr. Ayodeji Alajo
Dr. Xin Liu

ABSTRACT

In medical imaging it is a very common practice to use a technique known as Time-Step imaging in patients who might develop cancer. Time-Step imaging it a very powerful technique, however it can lead to unmanageable amounts of image data. Previously the only way to search all of this data was to manually look through all of the files. This had to be done by trained professionals who knew what to look for within the images and make a judgment about the patient based on the images. This paper discusses the development of an algorithm to have a computer search through these data files, and only surface the patients and images that have a high probability of having cancer. This allows the trained professionals to spend more time closely analyzing the few images that were marked as high risk.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

1. INTRODUCTION

A common practice in medical imaging is the use of a technique known as Time-Step imaging in patients who might develop cancer. Time-Step imaging starts with taking an initial scan of a patient and having a medical processional review the image, thus establishing a baseline. Subsequently more images can be taken and then reviewed using other techniques to routinely monitor the patient, looking for signs of cancer or other abnormalities. This leads to numerous images being created that a practiced oncologist must then search through looking for the development of any abnormalities in the time between images. The algorithm discussed in this paper was created so that a trained oncologist or radiologist need not search every image for signs of cancer, but instead be able to only review those images which are flagged as suspicious.

The design of this algorithm is to take and compare two sets of Computed Tomography (CT) data. Even though this process initially seems trivial, it takes a multitude of stems before final comparison can be completed. First, the data set needs to be imported into the program. Next, the data sets are registered, that is stretched, skewed, rotated, and translated, in order to ensure that they can be compared appropriately. After registration, the image data sets are subtracted using a point-by-point method. The last step in this algorithm is to compare the subtracted images and report whether a cancerous region exists. Figure 1.1 shows a flow chart of the algorithm.
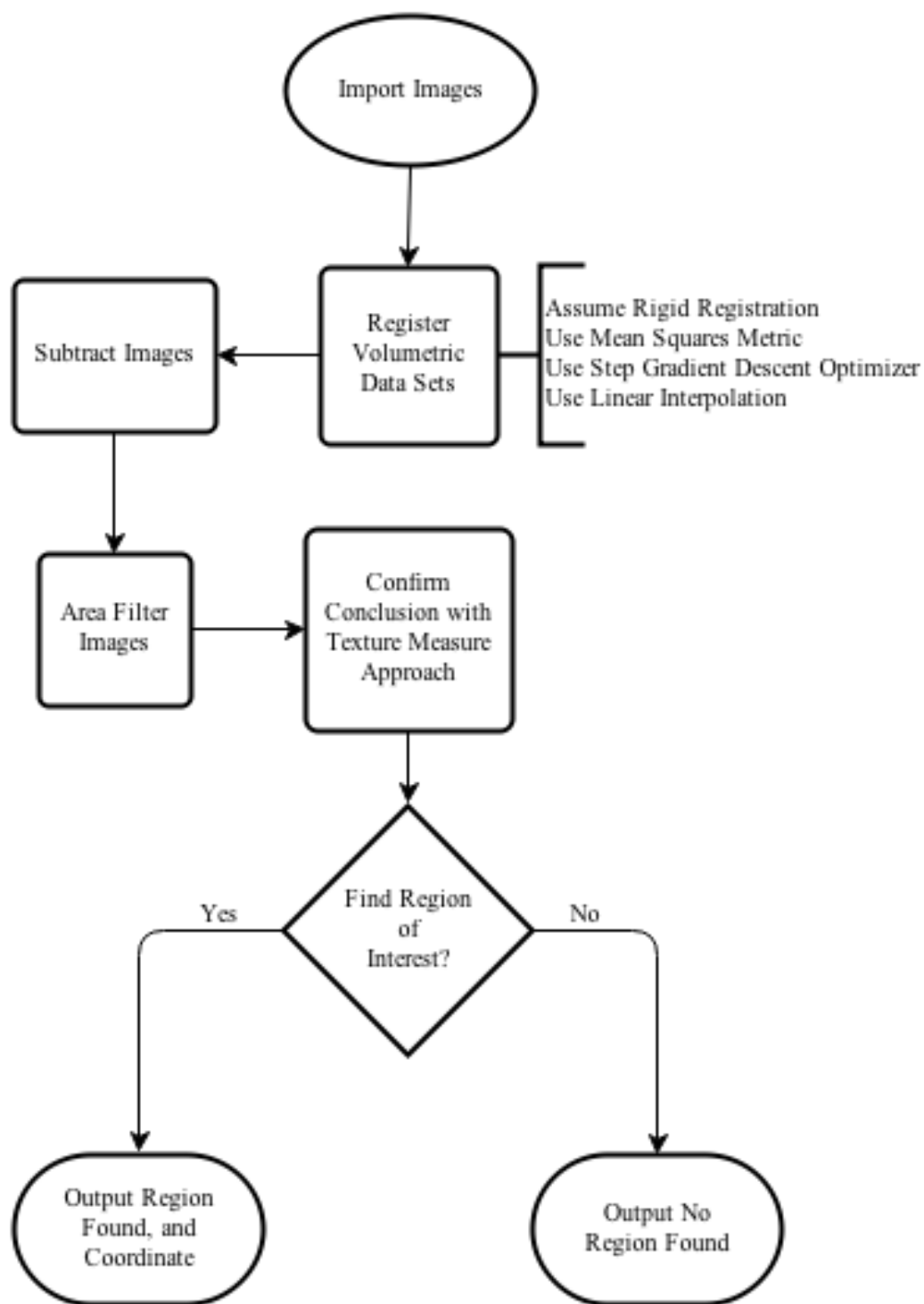
Figure 1.1: Algorithm Flowchart

2. BACKGROUND

2.1. CT IMAGING

CT images are specialized X-ray images that produce cross-sectional images of the patient. Then each slice can be assembled to construct a 3D image of the section of the patient that was scanned. This process is similar to taking bread slices and stacking them up to create a loaf. Each slice is one cross-sectional sample of the whole loaf [1]. Figure 2.1 shows a diagram of a CAT scanner, which uses the same principal as a CT scanner.
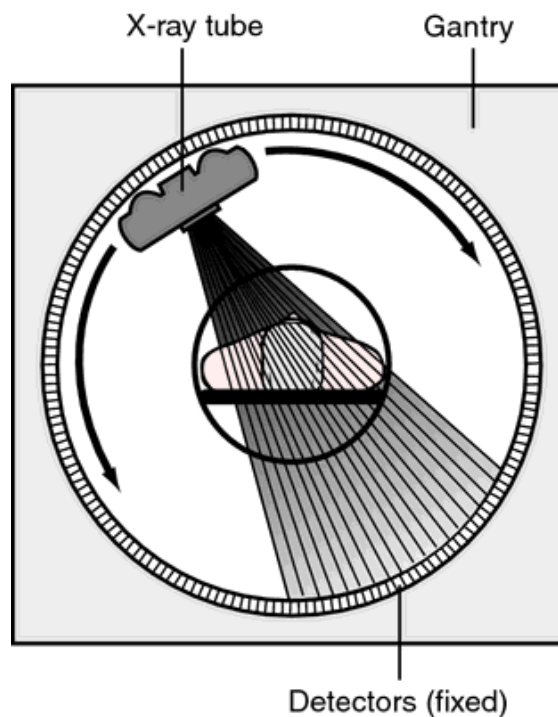
Figure 2.1:CT Unit [1]

2.2. 3-D REGISTRATION OVERVIEW

In this algorithm, the CT image will consist of a 3-D object that will need to be rotated and moved in order to ensure that the comparison can be adequately performed. This process is called registration, and it is used in many image analysis algorithms throughout the world. In Figure 2.2 an image of a box with a three-quarter pie extruded from the side is shown. This image could be rotated 90 degrees in order to produce Figure 2.3. Since these images are of the same object, but one is rotated relative to the other, we can register the two 3-D objects to perfectly overlap each other. This is a good way to visualize the 3-D registration process. The specifics of the algorithm and the images used for testing are explained and shown in the following sections.
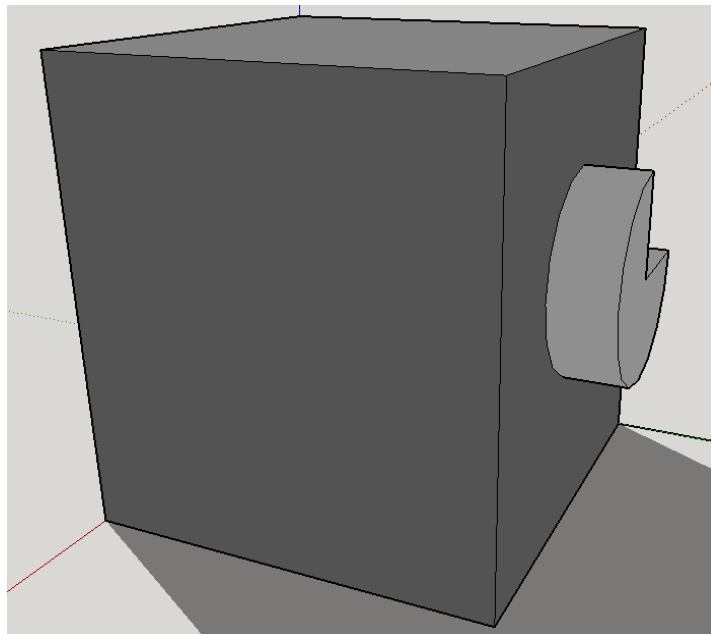
Figure 2.2: 3-D Registration Example Object
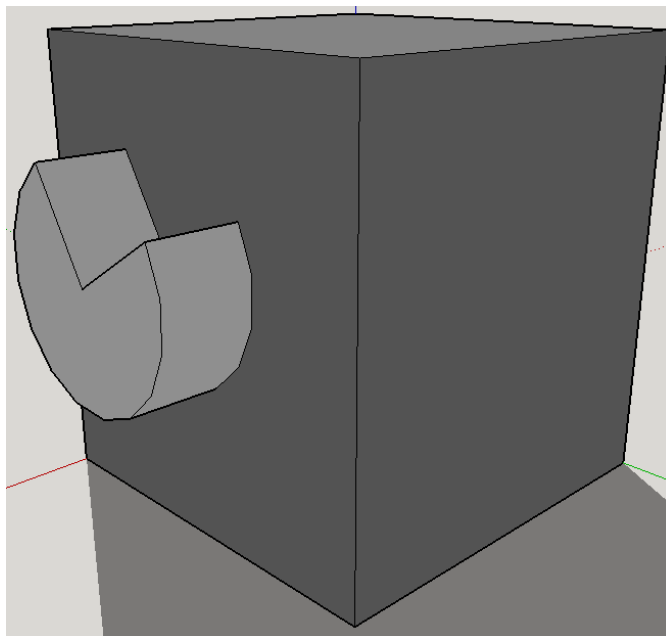
Figure 2.3: Rotated 3-D Image Registration Example Object

3. MATLAB METHODS

3.1. IMPORTING IMAGES

In an algorithm of this nature, the first step is to import the images that are being worked on. Initially, this seems to be a simple problem with a straightforward solution. However, CT images are three-dimensional (3-D) are thus require using an equivalent dimensional array. This causes problems in MATLab due to the limited number of cells available to be allocated at a given time. To circumvent the limitation MATLab presented, the initial algorithm was written using a 128x128x128 3-D Shepp-Logan Phantom [2].This allowed for the test of the algorithm to be developed while still maintaining compatibility with MATLab.

Once the 3-D phantoms were created, one for the "healthy" image and one for the "cancerous" image, they were saved as 128 separate images. These were then imported using a for loop that read each sucessive file and save it as the next level of the 3-D array.

3.2. 3-D REGISTRATION

Image registration has had many algorithms and implementations written over the years, leading to a well understood process for current registration algorithms. General classifications for image registration include Intensity, Feature, and Frequency based techniques. Feature and Intensity based registration techniques are very similar in their theory with Intensity being slightly easier to implement. Frequency based registration can only be applied to two-dimensional images and is therefore inapplicable to this endeavor. Thus, Intensity based registration techniques will be discussed.

Image registration starts with two different images of the save object that are comparable. Once the algorithm has the two images, it starts a loop that will decide how

to modify the moving image and how much to modify the moving image. The moving image being the one chosen to be modified to match the other image. The fixed image, the image remaining the same, is then compared to the moving image after it has been modified. This loop continues until the images are overlapping.

A metric (similarity measure) decides whether the images are close enough to quit the loop or send the moving image through the loop again. Figure 3.1 shows a diagram of this process.
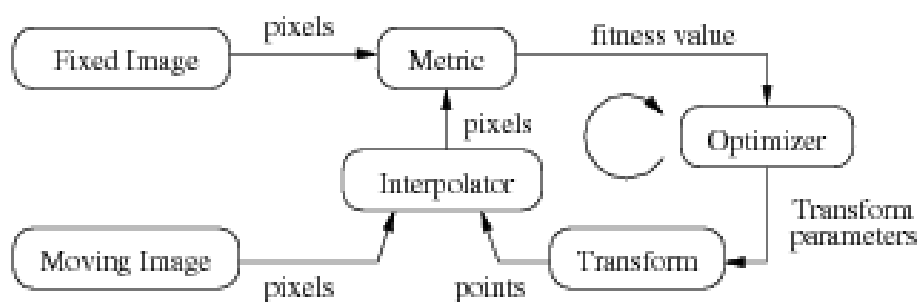
Figure 3.1: Image Registration Algorithm Overview

The algorithm works by piecing all of the different operations together and iterating until the moving image is within tolerance of the fixed image, or a maximum number of iterations is reached.

3.2.1. Similarity Measures. The metric has many different types and equations that can compare the intensity values in intensity based registration. These range from Mean Squares, where the sum of squared difference between intensity values are compared [4]. To Mutual Information, where the uncertainty of knowledge that can be assumed about the fixed image from the moving image is calculated [4]. This allows for a measure of the difference between the data sets, volumetric data sets in this case.

3.2.2. Optimizers. The Step Gradient Descent, a good example of an optimizer based on the gradient, calculated the gradient at a point and changed the moving image to translate that point toward the global minimum [5]. An example of an optimizer that relies on both gradient and attempts to apply that knowledge more intelligently would be the Quasi-Newton Method. This method is based on Newton's Method and is a generalization of the Secant Method. It assumes a function can be approximated as a quadratic function around the global minimum for the gradient, and attempts to calculate the global minimum from the current point [6].

3.2.3. Interpolator. The Interpolator types are just as varied as the other segments of the image registration algorithm. The types range from Linear Interpolation to Spline Interpolation. For linear interpolation, the function between two points is assumed to be a straight line; it as fast computational time, but can have high error rates and large deviations from the actual answer if the correct function is not close to linear [7]. Spline interpolation uses low degree polynomials over small intervals to fit a curve close to the correct function, having higher accuracy but also higher computation times.

3.2.4. Volumetric Registration. Many different algorithms were tested for each of the different segments of the image registration, and some were found to be the optimal for this application. Table I previews the algorithms chosen for this application.

Table 1.1. Algorithm Choices

Metric	Mean Squares
Optimizer	Step Gradient Descent
Interpolator	Linear Interpolation

3.3. SUBTRACTION

Once the images were registered, correct comparison can be accomplished. The simplest method for comparing images is point-by-point subtraction. The goal of this application is to find a cancerous region that has formed since the first image. Subtracting the images should leave just the cancerous region being non-zero and the rest of the image should be zeroes. Once completed, it should be a simple matter of parsing the array and finding the non-zero pixels. This however is not the case.

Once the columns are registered, there is usually come noise introduced by pixel interpolation and other approximations in the registration. These estimations lead to extra pixels that a simple algorithm might "see" as a cancerous region. This, the comparison algorithm needs to be more adaptive and intelligent than a simple search function looking for non-zero pixel values.

3.4. COMPARISON

A comparison algorithm more advanced than looking for non-zero pixels had to be utilized. If the assumption is made that the pixels that are not cancerous are merely a result of approximations made in the transformation and registration process; it follows that the noise pixels should be individual pixels or small groups of pixels located throughout the image. This leads to the conclusion that calculating the area of each object in the image and comparing the area to a threshold value should remove most of not all of the noise pixels. Once done, a simple comparison can be completed looking merely at the presence of non-zero pixels. This approach can be determined whether cancer exists in a registered image and greatly reduce false positives compared to a simpler approach.

The comparison function of the algorithm can also use the Gray Level Cooccurance Matrix to calculate the uniformity of the image. A unique characteristic of cancer in CT images, and all radiographic images, is cancerous regions are more uniform than normal tissue. If the uniformity of an image is calculated then it can be compared with statistical data from other images and can then be determined to be cancerous or non-cancerous.

4. C++ METHODS

4.1. LIBRARIES USED

In order to be able to analyze images in C++ libraries needed to be included in order to obtain the basic functions of image editing and analysis. These libraries are open source collections of C++ code that people have created in order to share and allow others to build more advanced algorithms based in simpler structures. There are a lot of libraries that contain image processing algorithms, and several of them were tried in order to discover the best option for this algorithm. These libraries were evaluated for their ability to analyze 3D images and their implementation of functions that can work with CT image file formats, commonly known as Digital Imaging and Communications in Medicine (DICOM) images.

4.1.1. OpenCV. OpenCV was the first library attempted due to its common use in image processing programs in C++, and its similarity to MATLab code. Because of this it was very simple to convert the majority of the MATLab code into C++ code using OpenCV. However, OpenCV was constructed around processing 2D images and has only recently been expanded to be able to analyze 3D images. Due to this the 3D image processing techniques were not able to be used to analyze DICOM images. Thus OpenCV was abandoned in search for another set of libraries that would provide adequate functionality for the creation of the analysis of CT images program.

4.1.2. VTK. Visualization Toolkit (VTK) was then used to create the algorithm. This library was not created with the same similarity to MATLab, however the structures were similar enough to convert the basic structure of the algorithm. Unfortunately this library focuses on the visualization parts of image processing, as the name implies, however this leaves the other parts of image processing less developed. So similarly the 3D techniques implemented in VTK were not adequate enough to create the desired algorithm.

4.1.3. ITK. Insight Toolkit (ITK) is the other side of the VTK library, so it contains plenty of image processing algorithms that can be used for creating an image processing algorithm intended for 3D images. In fact, ITK was created in order to analyze medical images including CT volumes. So the support for many common image formats, and analysis was superbly supported. As with VTK however, ITK was not written to be similar to MATLab in syntax. Nevertheless converting the basic structures created in MATLab into ITK structures was completed successfully. Thus ITK was the library chosen to be used for this project.

4.2. IMPORTING IMAGES

Like the MATLab code, the first step to analyzing an image is to import the image. Since ITK was created for analyzing medical images, the function to import DICOM images was used to allow for simpler importing. It also allows for the use if DICOM slices, or a single file containing the whole volumetric data set.

4.3. 3-D REGISTRATION

The algorithm for 3D registration in C++ is very similar to the MATLab implementation. This is by design as they are supposed to be the same procedure, simply implemented in different languages. Figure 5 shows the structure of the registration algorithm.

4.3.1. Registration Metric. The image registration metric used in the C++ implementation is the Mean Squares Metric. This uses a set if points to estimate the difference between the two sets of images. The algorithm uses the registration method to calculate the metric for any two data sets during the registration process.

The registration metric defines how well the two image data sets match. This is one of the most critical elements of the image registration process. Mean squares metric types are generally the simplest and most reliable registration methods that can be used. It uses the intensity values in the image to compare the two images; however, it requires the images to be of similar intensity values. This is a perfectly valid assumption for use with medical images, since there is a standard pixel intensity range for medical images. [9]

4.3.2. Registration Optimizer. The image registration optimizer that is used in the C++ implementation is the Regular Step Gradient Optimizer. This optimizer calculates the gradient in a parametric space that represents the image. This gradient allows the moving image to be modified in the correct manner in order to be properly registered. The Regular Step Gradient Optimizer also possesses some other enhancements that makes it more advanced than a standard gradient optimizer. The main enhancement possessed by the Regular Step Gradient Optimizer is its ability to modify the step size after every change in direction. [4]

The registration optimizer not only encompasses the simplicity and reliability of the standard gradient optimizer algorithm. The Regular Step Gradient Optimizer also possesses advanced features that allow it to be more efficient and less time consuming. This leads to a very elegant, and reliably algorithm that is able to register a range of image types that could be encountered.

4.3.3. Registration Interpolator. The image registration interpolator that is used in the C++ implementation is the Linear Interpolate Function. This is a simple linear interpolation function to estimate the pixel value between pixels when registration occurs. The optimal parameters for an interpolator to possess is the ability to calculate and recalculate pixels quickly and efficiently. To this end, the simple, yet perfectly effective interpolator was chosen. This allows the rest of the components of the algorithm to complete their task as quickly as possible to generate the best possible registration. [4]

4.3.4. Registration Transform Type. The image registration transform used in the C++ implementation is the Affine Transform. This is the counterbalance to the less advanced interpolator. The Affine Transform can handle almost any situation in an image registration problem. It can perform rotations, translations, shearing, and scaling. In this problem, one major assumption is the absence of shear transformation required due to the lack of skeletal change in a patient. This minor issue with the Affine Transform is easily mended with a factor change in the code, informing the transform algorithm that the shear transform is not required. [10]

4.3.5. Subtraction. Once the registration is completed on the images, the subtraction can be easily performed. This procedure simply subtracts the individual pixels in one image from the corresponding pixels in the other image. This allows a simple comparison to be performed that only takes into account the differences in the images, and any minor noise from the registration process.

4.3.6. Comparison. Once subtracting the images is complete, the comparison can commence. This entails several steps in order to minimize noise from the registration, and ensure that the results reported are the most accurate possible, with the given information. To that end, the subtracted image is thresholded into a binary image. This has two purposes, first it allows for much faster computation of the results, and it allows the use of binary image functions on the image. The binary image functions that are used is an area filter. From the known information about the images being processed, it can be assumed that very small regions are not cancerous, or indicating of a cyst. This means that they can be filtered out and the remaining areas can be analyzed. During the MatLAB testing however, the area filter was concluded to catch a high percentage of the regions of interest, so this was the limit of the C++ implementation. However, any future work could include the implementation of a Co-Occurrence matrix that would compare the uniformity texture measure of the image against other images. This would create a second layer of testing, which would make this algorithm even more reliable.

5. RESULTS

Due to the nature of this project, suitable CT images could not be obtained for use in the C++ implementation. However since the C++ and MATLab implementations use the same basic algorithm, it can be assumed that the C++ results would be similar to the MATLab results. Therefore, the following section only contains the results from the MATLab algorithm, but they can also be viewed as the results of the C++ algorithm.

5.1. PHANTOM APPROXIMATIONS

The initial algorithm was created using 3-D phantoms, which are very common for testing new algorithms dealing with CT and radiographic images. For the purposes of this paper, the phantoms that are shown will be the 64th slice in a 128 slice volume. Figure 5.1 and Figure 5.2 display the fixed image and moving image respectively.
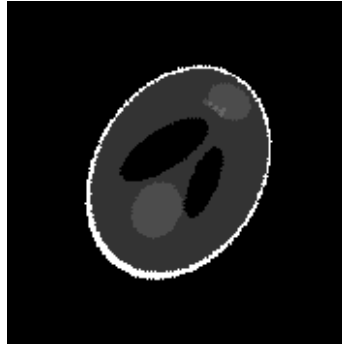

Figure 5.1:Fixed Shepp-Logan Phantom

Figure 5.2: Moving Shepp-Logan Phantom

5.2. REGISTRATION

Image registration was performed with the assistance of MATLab's built-in functions. The image registration was based on the assumption that ridged registration would be sufficient due to the constant bone structure of the patient. Figure 5.3 shows the paired images of the moving and fixed phantoms.

Figure 5.3: Pre-Registered Phantoms

During the image registration, the moving image is constantly shifted and rotated to align with the fixed image. Figure 5.4 is an image during the registration at

approximately halfway through the registration process. It can be clearly seen that the moving image (green) is closer to the fixed image (purple) that it was prior to the registration shown in Figure 5.4.
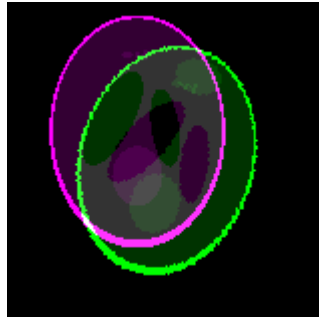

Figure 5.4: Partially Registered Phantom

Once the image registration was performed, the images were compared to check the accuracy of the registration. Figure 5.5 shows the overlaid registered images. The green oval shows the cancerous region only present in one of the two images.

Figure 5.5: Registered Phantom

5.3. SUBTRACTION

After the images were registered they were subtracted in order to determine if a cancerous regions exists. The ideal subtracted image would have the cancerous region left without any other pixels being non-zero. Figure 5.6 shows the subtracted image.

Figure 5.6: Subtracted Phantoms

Unmistakably, the subtracted image is not idea. Having a registration technique that leads to a non-ideal subtracted image will require a more advanced comparison method than simply looking for non-zero pixels.

5.4. COMPARISON

Once the image is registered and subtracted, the resulting image can be analyzed. This is done primarily through the application of a thresholding algorithm that looks at the area of each object and removed those that are too small to be a region of interest (ROI). In addition to this thresholding algorithm, a more advanced algorithm is used to double-check the findings, requiring the use of a Gray Level CoOccurance Matrix and a texture measure of uniformity. A measure of smoothness was also tried and works just as

reliably as the uniformity texture measure. Figure 5.7 shows the thresholded image multiplied by the original image that has had the smaller noise objects filtered out.

Figure 5.7: Area Filtered Subtracted Phantom

The texture measure search can look at either the area filtered subtracted image or merely the subtracted registered image in order to determine the presence of a cancerous region. The extra time was devoted to performing the area thresholding because it is important to be able to report the exact location of the cancer as opposed to merely confirming the presence of cancer.

6. CONCLUSION

The ability for medical professionals, and specifically oncologists, to look at and analyze radiographic images is severely limited and the need always exceeds the ability. Because there is a great need for fast reliable analysis of radiographic images, this project was started to create an algorithm for automatically searching images for a cancerous region. This algorithm looks at CT volumetric data sets; more specifically at the case of time-step CT images. This means that the first CT images was taken, analyzed by a medical professional and declared to be non-cancerous. Afterward, this algorithm can take future CT images and analyze them for signs of cancer.

The algorithm created during this project first imports the volumetric data sets, and then registers the data sets. Once the images are registered, the images can be subtracted to find any ROIs. This is done by filtering the image by an area measure, and then by looking at a texture measure that calculates the uniformity or smoothness texture measures. When the whole algorithm was implemented, the end result was very successful

REFERENCES

[1] United States Food and Drig Administration, "Computed Tomography (CT)," 07 08 2014. [Online]. Available: http://www.fda.gov/Radiation-EmittingProducts/RadiationEmittingProductsandProcedures/MedicalImaging/MedicalX-Rays/ucm115317.htm.

[2] M. Schabel, "3d Shepp-Logan phantom," 20 September 2006. [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/9416-3d-shepp-logan-phantom.

[3] ITK, "Insight Segmentation and Registration Toolkit," 21 April 2014. [Online]. Available: http://www.itk.org/Doxygen/html/ImageSimilarityMetricsPage.html.

[4] M. Chao, "Insight Segmentation and Registration Toolkit," 2 May 2006. [Online]. Available: http://www.itk.org/pipermail/insight-users/2006-May/017685.html.

[5] NAG Library, "Quasi-Newton," [Online]. Available: http://www.nag.co.uk/numeric/fl/nagdoc_fl23/pdf/E04/e04jyf.pdf.

[6] R. C. Gonzalez, R. E. Woods and S. L. Eddins, Digital Image Processing, Gatesmark Publishing, 2009.

[7] Insight Segmentation and Registration Toolkit, "Image Similarity Metrics," 2 Nov 2014. [Online]. Available: http://www.itk.org/Doxygen/html/ImageSimilarityMetricsPage.html.

VITA

This research was completed by Jonathan T. Scott who was born on October 29th 1990. Thus far, he has obtained his Bachelor's in May of 2013 in Nuclear Engineering at Missouri University of Science and Technology, along with a minor in Mathematics. With the completion of this Thesis and the project discussed herein, he will also obtain his Master's in May of 2015 in Nuclear Engineering at Missouri University of Science and Technology, focusing on computer analysis of medical images. This research was supported by the Nuclear Regulatory Commission, through the NRC Graduate Fellowship program, and the Chancellor's Fellowship through Missouri University of Science and Technology.