ATI 3D Aneurysm

Preliminary Proposal

Ahmed Almutairi

Scott Barberii

Jackson Tomaszewski

Austin Vest

NAU Fall 2020

Project Sponsor: ATi

Faculty Advisor: Dr. Tim Becker, Chris Settanni

Instructor: Dr. David Trevas

DISCLAIMER

This report was prepared by students as part of a university course requirement. While considerable effort has been put into the project, it is not the work of licensed engineers and has not undergone the extensive verification that is common in the profession. The information, data, conclusions, and content of this report should not be relied on or utilized without thorough, independent testing and verification. University faculty members may have been associated with this project as advisors, sponsors, or course instructors, but as such they are not responsible for the accuracy of results or conclusions.

TABLE OF CONTENTS

Contents

DISCLAIMER I
TABLE OF CONTENTS II
1 BACKGROUND 1
1.1 Introduction 1
1.2 Project Description 1
2 REQUIREMENTS 2
2.1 Customer Requirements (CRs) 2
2.2 Engineering Requirements (ERs) 2
2.3 House of Quality (HoQ) 3
2.4 Functional Decomposition
2.4.1 Black Box Model 3
2.4.2 Functional Model 4
3 DESIGN SPACE RESEARCH 4
3.1 Literature Review 5
3.1.1 Jackson Tomaszewski 5
3.1.2 Scott Barberii 5
3.1.3 Ahmed Almuatiri 6
3.1.4 Austin Vest 7
3.2 Benchmarking 7
3.2.1 System Level Benchmarking 7
3.2.1.1 Existing Design #1: MARACUS 7
3.2.1.2 Existing Design #2: Object Detection Programs 8
3.2.2 Subsystem Level Benchmarking 8
3.2.2.1 Subsystem #1: Training Data 8
3.2.2.1.1 Existing Design #1: Image Labeler 8
3.2.2.1.2 Existing Design #2: Video Labeler 9
3.2.2.1.3 Existing Design #3: Ground Truth Labeler 9
3.2.2.2 Subsystem #2: Convolutional Neural Network 10
3.2.2.2.1 Existing Design #1: R-CNN 10
3.2.2.3 Subsystem #3: DICOM Application 10
CONCEPT GENERATION 12
4.1 Full System Concepts 12
4.1.1 Full System Design #1: 3d Aneurysm Location and Measurements 12
4.2 Subsystem Concepts 12
4.2.1 Subsystem #1: Locate Aneurysm 12
4.2.1.1 Design #1: Neural Network 13
4.2.1.2 Design #2: Aneurysm as a Function of Volume 13
4.2.2 Subsystem #2: Graphical User Interface 13
4.2.2.1 Design #1: User Friendly GUI 13
4.2.3 Subsystem #3: Volume Finder 14

4.2.3.1 Design #1: Integrate Area Over Length 14
4.2.3.2 Design #2: Use Sphere Approximations 14
5 DESIGNS SELECTED – First Semester 15
5.1 Technical Selection Criteria 15
5.2 Rationale for Design Selection 15
6 REFERENCES 16
7 APPENDICES 19
7.1 Appendix A: QFD 19
7.2 Appendix B: Intensity Finder 19
7.3 Appendix C: Dicom Processor 23

1 BACKGROUND

1.1 Introduction

This is the preliminary report for the ATI 3D Aneurysm Capstone project to present current progress to the sponsor Dr. Becker and Christopher Settanni at ATI (Aneuvas Technologies Inc.) as well as for credit for the fall semester of 2020, ME 376C Capstone class at NAU (Northern Arizona University). The ATI Aneurysm project's objective is to generate a program / algorithm utilizing MATLAB that will allow the user to input a stack of Dicom files (industry standard for medical images) and output label images of potential aneurysm location which could also be analyzed in the same software program. The user or client is considered to be ATI and medical professionals who do not necessarily have a background in coding or programming. The program will involve using functions, applications, and machine learning components built into or created in Matlab.

The importance of this project is its potential to be used by medical professionals who can efficiently use this program to locate all potential aneurysms and then analyze them. This would decrease time spent by a highly paid radiologist or doctor looking at hundreds of Dicom files. Specifically, ATI is a lab that is trying to and already has developed new ways to cure aneurysms, so this program could decrease valuable time spent by the lab locating aneurysms they need to find in order to make a cure for that specific to the aneurysm. For patients who might have a mild or life threatening aneurysm, they can visit a hospital or clinic that has this computer program and find out if they need to see a radiologist and or doctor minutes after their MRI or CT scan. This would be in future if the team's program works well enough to double check a medical professional, radiologist, or doctor. This decrease in time spent ultimately results in decreased cost for patients who are worried about a potential aneurysm and provide more confidence to radiologists and doctors.

1.2 Project Description

Below is the project description provided to the team by the client that the team will follow closely. They will continue to meet with the client throughout the project so that both parties can confirm that the work being done is in fact the work that is wanted.

> "The scope of this design project is to develop a standalone algorithm to determine the size of aneurysms. The algorithm will upload medical imaging (DICOM) files containing MRI data into MATLAB (or a similar program). Voxel images will be processed to extract the aneurysm space from the surrounding brain scans."

2 REQUIREMENTS

This section will detail both the customer and the engineering requirements. The customer requirements are the goals for the team to hit that will satisfy the client. They can be as simple as the design must fall within the budget provided to the team. The customer requirements are heavily related to the engineering requirements, in that the customer determines the engineering requirements of the design, which are then polished by the team. The team does not have many because this project is computer based with less available physical measurements. The engineering requirements will allow us to measure the performance of the program as the team progresses.

2.1 Customer Requirements (CRs)

1. Cost within budget
2. Reliable program
3. Input is DICOM files
4. Output location and volume
5. User friendly interface / simple enough for most to use

Above is a list of the customer requirements that the team and client created. To start the team's priority is to stay within the budget. If they cannot create a system that is within their budget, then the system is unusable or unreasonable as it can not be properly made.

Next requirement is that the program and UI must be reliable. If the program is likely to malfunction or produce incorrect information, it is once again useless. As engineers, the team values reliability and believe that their customers do as well. This will be measured by the engineering requirements of Time and Accuracy.

The program must be able to accept and process DICOM files. This is because these are the types of file extensions that medical professionals use and will be expected of the code. The team would like to keep the original DICOM file in use rather than convert it into a TIFF or other file type. This is in order to keep quality which could be lost in a conversion from one file format to another.

Once the DICOM is accepted the program needs to perform the tasks in the project description and produce a location and volume for the aneurysm. Is this the main purpose of the project and is one of the most important requirements that the team has.

Finally, the last requirement is that the program must be usable for people who are not necessarily tech savvy or knowledgeable in coding. The team wants anyone who has access to a DICOM file and believes that there is a chance of an aneurysm to be able to use the program with ease and produce useful information to the user and/or patient.

2.2 Engineering Requirements (ERs)

1. Volume: mm^3
2. Location: y,x,z mm
3. Time: s
4. Accuracy: ratio

Above are the compiled engineering requirements for the program the team is creating. The first being volume. This is referring to the volume of the aneurysm that the program is identifying. The programs are only scanning DICOM files of brain tissue, which means that the size of the vessels and vessel defects will be small, on the scale of millimeters cubed.

The next requirement for the program is location. This is the location of the aneurysm within the brain. The team plans to use millimeters for this application due to the size of the parts being dealt with. They also want to create a reference frame in the x,y, and coordinates so that the aneurysm can be easily identified from a known perspective and not arbitrarily is space.

Another engineering requirement is time. This applies to both the time it takes for the program to process

and produce the location and volume, as well as the time it takes for the user to interact with and run the program. Both of these are trying to be minimized, as time is very important and the faster the process can be done the better.
The last engineering requirement is accuracy of the program once it is trained based on how many potential aneurysms were found compared to how many aneurysms there were in the given DICOM stack. The level of this accuracy is still being debated amongst the team. The team hopes to get the number as high as they can, within 90% accuracy. This engineering requirement relates to the customer requirement of reliability. These will allow for measurement of performance in the testing phase when we input new DICOM files/ stacks into our program.

2.3 House of Quality (HoQ)

The QFD is the aid to the team in deciding what were the most important aspects of the project. The values seen in the QFD are tabulated from what the team assumed would be the general consensus as well as what other medical professionals are saying. It was found that accuracy is the most important aspect of the engineering requirements. This follows through with the scope of the project. It needs to be accurate otherwise it will not be useful to medical professionals. The least important engineering requirement was found to be the time it takes to process information and use the program. While it is still important to not be a time consuming process, the team would much rather have it take 3 minutes but provide accurate information than have it take 5 seconds but produce incorrect results. This relates back to accuracy in that the team values accuracy the most. The full QFD can be found in the appendix of the report.

2.4 Functional Decomposition

2.4.1 Black Box Model

In Black model team focused on design and developed a standalone algorithm program in MATLAB that can identify boundaries of aneurysm and measure vascular defects. For the design inputs there are two types, the first one is information input which imports a DICOM file to the MATLAB program. Second type is physical input Click bottom / run information in GUI this process untrained medical professionals who are not familiar with MATLAB will be able to import their own DICOM file that has an aneurysm case to display these images in the program. Results that can be obtained from the program are location and volume of aneurysm users will be able to display MRI scan images in 3D format which help them to find location of aneurysm.

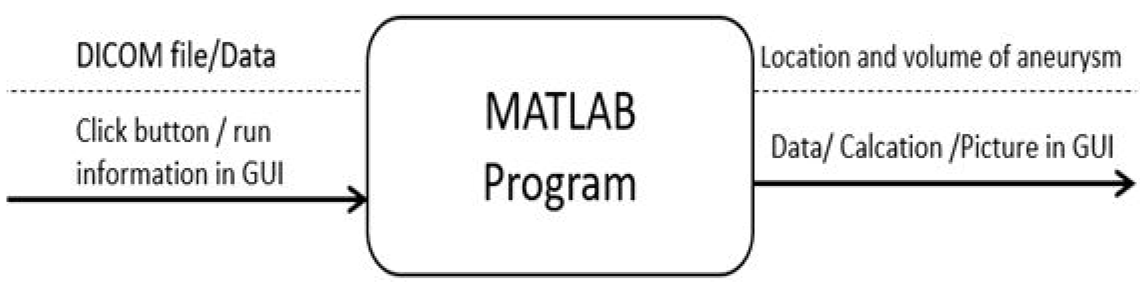


Figure 1: Black box model

2.4.2 Functional Model/Work-Process Diagram/Hierarchical Task Analysis

Below is the functional model that the team created to represent what the program will be accomplishing. It is split into two different parts, the code which can be thought of as the behind the scenes portion of the program, and the graphical user interface component, which is the parts that the user interacts with. The code can be thought of as three distinct different functions. The first being the database of DICOM files and processing those files into something that can be used by the second function: the aneurysm locator. This function will use the provided, read DICOM files and locate the aneurysm if there is one within the files. Finally, once located the volume finder can calculate the volume of the aneurysm. This is all out of sight to the user, who will only see the final products from each of these functions: the 3-D DICOM file displayed, and the location and volume of the aneurysm if there is one. The GUI is what the user will provide inputs to and receive outputs from. It should be simple to use and uncluttered as is a requirement mentioned in the above customer requirements. It will provide the displays of DICOM files and where the aneurysm is to the user. A great way to think about the code and GUI and their relationship is that the code performs all of the engineering requirements which are then transformed into understandable visuals and data by the GUI.

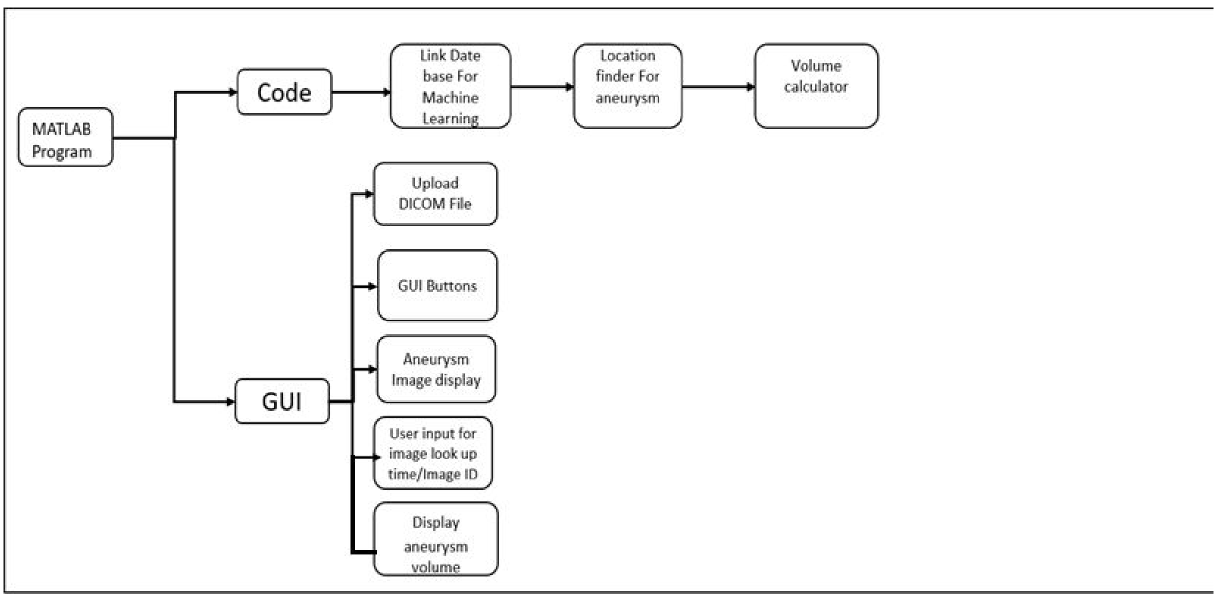


Figure 2: Functional Model

3 DESIGN SPACE RESEARCH

3.1 Literature Review

Literature Review is an important part of any project to gain a base understanding of a topic. This section will go over the research done by each member of the team.

3.1.1 Jackson Tomaszewski

Jackson was tasked to research how to calculate volume and design a graphical user interface. It is important to calculate the volume of an aneurysm because it will allow the doctor to choose the correct size coil for the application. The coil will fill the ballooned portion of the aneurysm and allow the blood to flow through its normal path. Our group found that the easiest way to calculate volume is through the process of integration. The entire process to calculate volume of the aneurysm first starts with a DICOM file. This stack of images is created from either a CT scan or an MRI. Each of these images has a scale included in its metadata. With this information the bounds of integration will be able to be set. Next the equation for integration has to be found [2]. This equation is going to be different for every aneurysm. It is important to note that there are already programs that do this. Where many of which have errors in the calculations. Some that are used in the medical industry range from 1.7-2.6% error [1]. So to find the equation for the specific shape a Lagrange interpolation polynomial will have to be used [3]. The scale from the DICOM images will allow for the polynomial to be created. Our MATLAB program will have an algorithm that will allow this to be completed automatically.

An interesting article by Boston university students discusses how volume of bones were calculated. It first started with a process similar to ours where a medical image was inserted into MATLAB. This image was then processed and volume was eventually calculated. To calculate volume a meshgrid was displayed over the image to implement a scale. This scale then allowed for the volume to be calculated [4].

A graphical user interface, or GUI, is a very important aspect for any program. It is the front end and the only visual representation for the code. It will allow the user to upload a DICOM file and run the actual MATLAB code that will find the location of the aneurysm and calculate its volume. MATLAB has its own GUI builder but it doesn't have as many features as Visual Studio. This software program will allow the program to have a unique and modern look. In the GUI there will be various features that will allow the user to upload a DICOM file, run the program, have a display of the aneurysm, and display the volume of the aneurysm. All of which will meet the customers needs and more.

To find out more about the diagnosis and procedure for treating an aneurysm Dr. Arthur Duberg was contacted. He discussed the reasoning behind which type of imaging process was used during certain scenarios. He explained that the majority of doctors use a CT scan, while an MRI is only necessary if the patient is allergic to dye that is injected to show the vessels. He also explained that a DICOM file is the universal image format for the medical industry [5].

3.1.2 Scott Barberii

For Scott's technical analysis and research he chose to look into the processing of DICOM files and how MATLAB is able to read them. First, it is important to understand what a DICOM file is [6], because it is not just an average picture file like a .jpg or .png. It is much like a stack of images that are compressed to create a 3-D image of the real world. This is why they are so applicable for medical procedures. They can provide a 3-D image of a specific part of the body, in the team's case, the brain. This means that opening the file like any other is not possible. You would only get one of the many slices that are needed to fully understand the picture. This means that the image can either be shown as a montage, where all of the slices are laid out in order so that the differences between them can be seen. Or it can be rendered in 3-D. The 3-D rendering would provide the best view of what the system actually looks like, but might be difficult to navigate and if the system has many small interior parts, they could be left out of sight.

Matlab is one of the programs capable of processing a DICOM file in both of the ways discussed above. This is where research came in handy. The goal was to learn about how the DICOM file works and how MatLab processes the file to get either a montage or 3-D rendering. This was done through tutorials though mathworks as well as third parties who have looked into medical imaging and put the information online for free. Youtube is a great place to start, many professors and computer engineers post video tutorials for free that the public may use [7,8,9]. This is where I got a general idea of how to create a program that could take a DICOM file and turn it into a visual medium. From here I could create my own MatLab code and create a montage of DICOM slices. As of now the team is still working on 3-D rendering.

Another aspect of the code that Scott looked into was intensity finding. One of the things the client mentioned was that within DICOM files, materials much as bone mass are intense on the gray scale. Bone mass cannot contain aneurysms and therefore can be deleted. This requires less processing power from the system running the code, as there will be less it has to comb through to try and locate the aneurysm. Matlab provided a basic overview of the imaging toolbox [10], a set of functions that are very useful for managing images. From this video Scott created code that could identify the high and low intensity areas within the image and highlight them so that the user can easily see them.

The code and images that the code produces can be found in Appendix 6.2 and 6.3

3.1.3 Ahmed Almuatiri

Ahmed started focusing on MATLAB concepts that are required for analysis images and display them in the program. The MATLAB program has multiple image toolboxes that already exist in the app's features. These tools are designed to analyze and visualize images in one, two and three dimensions[11]. The MATLAB community website has useful basic information about processes that are required to read images in the program such as examples on writing functions in window command [12]. The team focused on visualizing the DICOM file in MATLAB that contained a series of MR scan images. So most of my research was about image toolbox features that have the ability to visualize images in two and three dimensions.

So my technical aspect is to read the DICOM image format in the MATLAB program, which is the important part for the project. From tutorials that are available in YouTube and the MATLAB community I found out in MATLAB there are two different ways to visualize series of MR, CT bone scan images compressed in a DICOM file. The normal way is to write a function in the command window using a function called Dicomread(filename), by writing down this function the program recognizes file format format and has the ability to run these images properties. Information about this way was also provided by the sponsor Chris. Second way is using a DICOM browser that is already existent in the app feature and can read the DICOM file without writing any function in the command window. DICOM browser display series images in montage which mean display images separately [14]. Also from the browser can get more details on patient information and series images properties such as row, columns and frames.

Visualization three-dimension images in MATLAB after reading a series of images from a DICOM file in the program. These series images need to be compressed into one image with three dimensions to achieve this goal we need to use volume viewer apps [14] . Volume viewer is one of useful image toolboxes that exist in MATLAB; it has the ability to transfer these series images of two dimensions to three dimensions.From DICOM browser we can open volume viewer and visualize images in 3 dimensions using a volume viewer which helps us to locate and visualize an aneurysm part.

3.1.4 Austin Vest

Austin initially focused on defining medical terms the team did not know or understand since we are studying engineering not medicine. The team had an initial learning curve for recognizing all the medical language. General information websites like reference [15] provided definitions and background information for the medical processes like MRI (magnetic resonance imaging) and X-ray that generated the Dicom files the team was interested in. This brought the team up to speed on medical language and united the team on definition allowing us to communicate effectively moving forward. This student then looked into the DICOM (.dcm) file format. The Standard for Dicom files website was found [16] which has important information about Dicom files in the industry and the standard that all Dicom users must conform to. This makes sure medical facilities around the world can use every Dicom file just like most other file formats. The information on this website as well as specific standards provided the team with the background and structure of Dicom files needed to use them in a coding application.

Team members were initially provided a couple articles by the client contact Chrisotpher Settanni regarding the state of the art of other programs and applications that existed and how they were made as seen in reference [17]. These reports explained the process of inventing computer programs through hard coding that would locate a blood vessel like an artery or vein and the program would follow the vessel wall until it determined that the vessel wall diameter had increased in size "too quickly" indicating an aneurysm was present. These articles provided an introduction to begin to think about how to mathematically in a computer locate and measure aneurysm. They showed that this was possible. Christopher also gave a presentation on his progress in reading dicom files in Matlab as seen in reference [18]. This gave the team images and 3D models of brain aneurysms to keep in mind.

Mathworks is the parent company that continuously updates Matlab and provides terabytes of information on every aspect of Matlab and its possibilities. Vary articles and explanation on Mathworks provided the team with the bulk of the needed information applicable to Matlab. As seniors in the mechanical engineering program at NAU, the team has experience with Matlab, but not Dicom image processing and machine learning. References [19] and [20] These explanations give clear ideas about the process that is machine learning or deep learning in Matlab. Several Neural Networks already exist and the team can create their own using Deep Learning Design. References [21] and [22] discuss the first step in machine learning which is creating training data by manually labeling the ROI (region of interest) that the programer wants the program to recognize automatically. Currently the "Image Labeler" app is the correct choice. This capstone team realized the potential and usefulness of machine learning after our advisor Dr. Trevas made us aware of the current importance in the real world industry. Mathworks pages will be the main source of information as the team dives into Matlab.

3.2 State of the Art - Benchmarking

The team has free access to Matlab through NAU. Matlab is focused toward calculation and image processing, the team has decided to use Matlab as the primary coding language. This initially saves the sponsor money since they do not have purchase access to a coding language and because the sponsor also has free access to Matlab. If and when this program is functional and accurate enough to be used by others, this program can try to be transferred into a more available coding language for public use.

3.2.1 System Level State of the Art - Benchmarking

3.2.1.1 Existing Design #1: MARACAS

As seen in Reference [17], MARACAS is a software designed in 2002 that fulfilled this team's basic customer requirements. It is a post processing program that visualizes blood vessels in 3D from MR angiography and automatically makes measurements of vessel defects. This article was read early on in the project since it was initially provided by the sponsors. As mentioned before, it gave the team a good vision for the project but it also provided possibilities for engineering requirements and areas of emphasis. From the work of this other team, we noted the good and bad ideas about their program. Like MARACAS, our program should be able to be used on standard PCs, create a 3D model for the user, and measure any aneurysms. After seeing the struggle of this other team,we wanted our program to be easy to code/understand and use a more applicable coding software. This relates directly to this team's requirements because Matlab is a better calculation based coding language, it has advanced 3D modeling, and Matlab can be used on most PC platforms.

3.2.1.2 Existing Design #2: Object Detection Programs

As seen in reference [23], Object Detection is a type of program that has been made in several different neural networks. This type of machine learning has been applied to facial recognition and vehicle detection meaning there are specific programs that have already been made several times. Without going into specific programs, the team decided to follow a similar path since the objective of this project is to find a non uniform shape among a complicated background. This format of utilizing deep learning and neural networks in Matlab also gives the team many possible avenues to try to locate aneurysms if one of the paths ends up not working for this design problem. This approach also allows for customization as seen below in the Neural Network subsystem section and the Concept Generation section where the team can create their own neural network.

3.2.2 Subsystem Level State of the Art Benchmarking

3.2.2.1 Subsystem #1: Training Data

Again as seen in reference [24] and [25], creating training data is the first step in machine learning and deep learning in Matlab. Training data is created by the programmers of this ATI Aneurysm team to teach the neural network what objects of types of attributes to look for. In this project, aneurysms will be manually labeled, highlighted or emphasized. The several tools built into Matlab in the Image Processing and Deep Learning "Tool Boxes" that facilitate this labeling will be discussed below.

3.2.2.1.1 Existing Design #1: Image Labeler

Image Labeler is a way to label a stack of images for the purpose of creating training data as seen in reference [26]. This is an application in the image processing toolbox in Matlab and allows for several options and variations. An ROI(region of interest) can be drawn around the object of interest or specific pixels of the object can be highlighted. This App can also follow objects through multiple frames which will increase efficiency greatly not having to draw a box around 50 different aneurysms. Since the team is dealing with DICOM images this is currently the primary method of generating training data.

Figure 3.2.2.1.1a: ROI (left) and Pixel Highlighting (right) in Image Labeler

3.2.2.1.2 Existing Design #2: Video Labeler

This Application is similar to Image Labeler in that it can label ROIs or highlight pixels on individual frames of a video. Video Labeler was used for the vehicle detection problem as seen in reference [27]. Because Dicom files do not come in video format and are not usually used by medical professionals in video format, this application will not be used but is kept in mind for a frame of reference.

3.2.2.1.3 Existing Design #3: Ground Truth Labeler

Ground Truth Labeler is more intricate in that it can label information from multiple sources over time. As seen in reference [28] and figure 3.2.2.1.3a, multiple sources can also be thought of as multiple inputs. In a video it was seen that the user labelled a vehicle based on video input and lidar input after linking the time for both inputs. Since the team is only given Dicom files, there will not be multiple inputs to utilize. Again, this existing available subsystem is good to know in case we get multiple inputs like MRI and CT scan in the distant future.

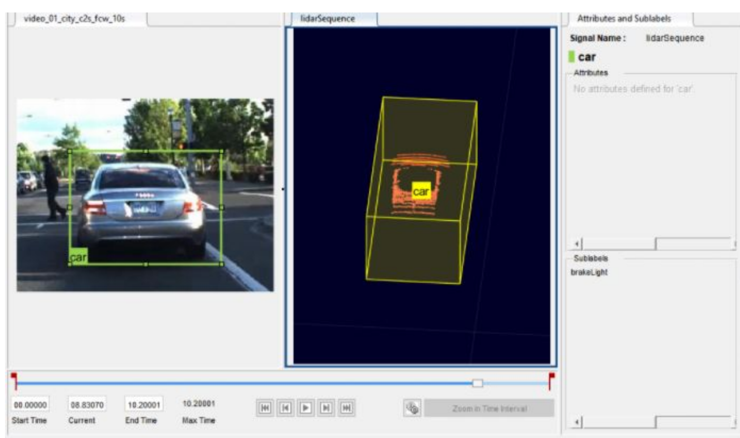


Figure 3.2.2.1.3a: Ground Truth Labeler utilizing Video and Lidar

3.2.2.2 Subsystem #2: Convolutional Neural Network

A convolutional Neural Network is a trained program that is able to learn how to perform specific tasks. It is a great tool for trying to emulate human understanding and application of knowledge. It uses a set of functions to determine what in a given input is important based labeled data that it was trained with.

3.2.2.2.1 Existing Design #1: R-CNN

R-CNN stands for Region-Convolutional Neural network. This a form of deep learning locates a region of interest. It then categorized that region with a given label. Finally it resized the region to fit the wanted object. It does this all based on what it learned from its training data. This is part of what our program needs to do for the locating aneurysms objective.

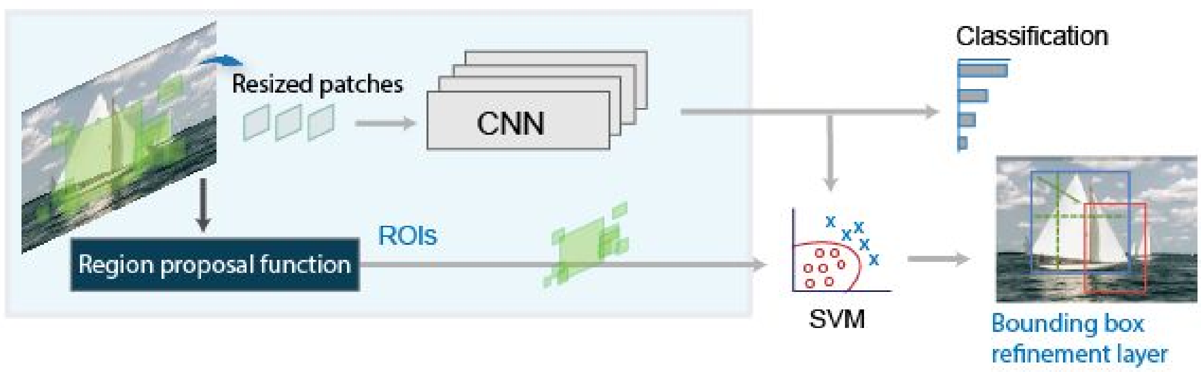


Figure 3.2.2.2.1a: Region Convolutional Neural Network.

3.2.2.3 Subsystem #3: DICOM Application

DICOM application has the ability to store the date from DICOM file as volume with separate variables into matlab program [29] .The application is designed to read these series images that are compressed in DICOM file and visualize them as separate images. Also can read properties that are attached in file such as patient information and series images properties like numbers of row, columns and frames quality as it shown in figure 3.2.2.3a . It is important to use this application to view DICOM because one of our customer requirements is time processing. Time processing to read a DICOM file using this application is very short, it only takes a few seconds to visualise images. And also allow us to export these series images to volume viewer and video volume viewer applications as it shown in figure 3.2.2.3b .

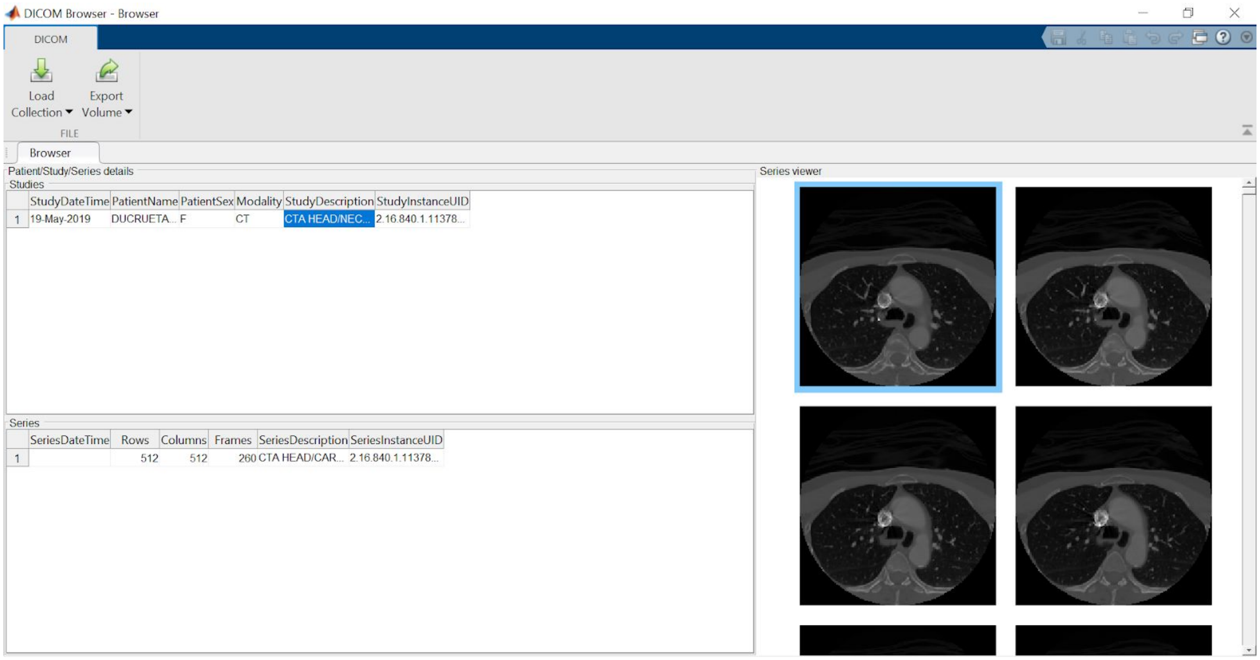


Figure 3.2.2.3a: DICOM browser

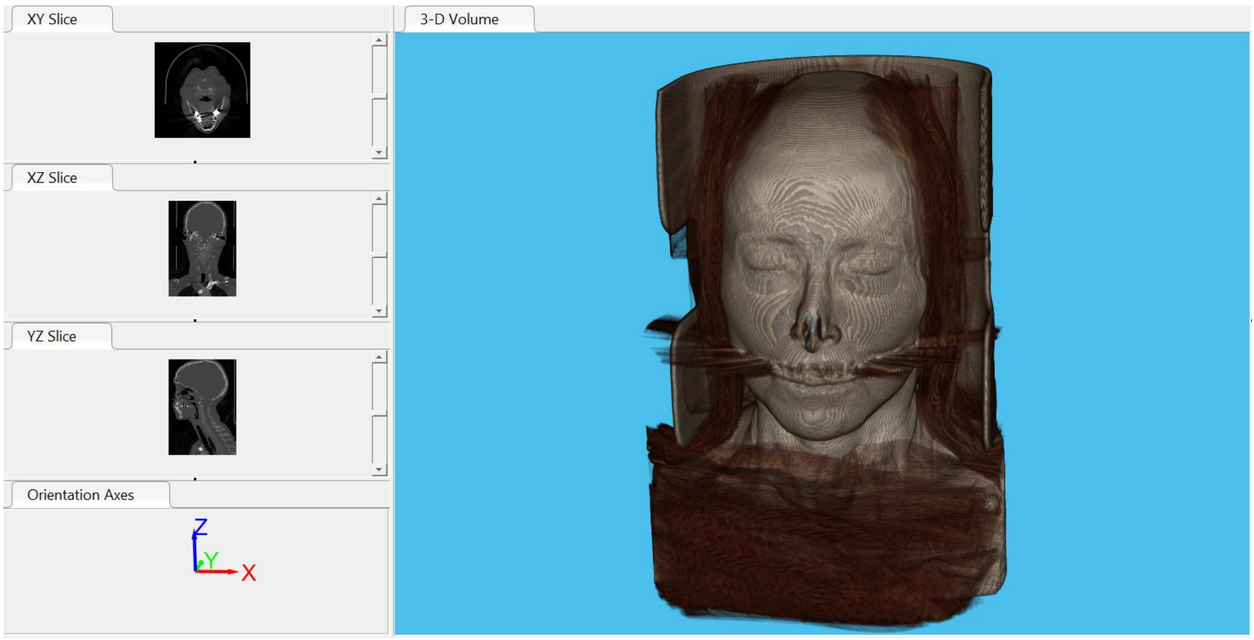


Figure 3.2.2.3b: Volume viewer

4 CONCEPT GENERATION

For this project the team is using mostly concepts that are already existing, as mentioned above in the benchmarking section of the report. The new concepts that they would be creating would be new convolutional neural networks specific to this application.

3.3 Full System Concepts

3.3.1 Full System Design #1: 3D Aneurysm Location and Measurements

This concept is a machine learning format with iterative and customizable potential. It will be a downloadable program in Matlab with a GUI that uploads Dicom files and outputs potential vessel defect / aneurysm locations and size of that aneurysm. It will also display a 3D model of the aneurysm. This design was based on the scope given by the sponsor and then focussed by the research done by the team. All the research, advice from our advisor, and guidance from our sponsor all pointed to making a computer program to automatically complete the objective. The main point of this concept is being able to iterate as we go without cost. We can test this concept and then make small adjustments as needed making this concept the most versatile.

This paragraph will describe step by step use of the intended concept program. The user will be able to download this program. Open the program on their standard PC and then a GUI will open. The user will then be able to upload a stack of Dicom images from their PC using a GUI button. Then our Matlab program will run the images through the neural network in a Deep Learning and locate all potential aneurysms. Then the program will make preliminary measurements on the aneurysms. The next thing the program will do is compile a 3D image from the stack of dicom images through a concatenation process. During this internal process, a loading bar and anticipated time left. Finally the program will display the model to the user and allow the user to make any extra measurements. A download link will also appear for annotated Dicom files.

Some positives of this concept are its easy of use and potential availability. It will be able to be used by almost anyone with a PC. This also makes it easy to use in conjunction with the user friendly GUI. Another positive about this concept is that because it uses machine learning, the accuracy will only increase as more data is imputed and feedback is given to the program. This Concept is also low cost. The team does not currently foresee any monetary costs. Another positive is its potential for customization. The main point of this concept is being able to iterate as we go without cost.

Some negatives to this concept are limited background knowledge, difficult conceptualization, and it not being tangible. The team has worked with Matlab but not specifically with its image processing or deep learning applications. This has already and will require much research time to gain an idea of the various subsystems. This leads into machine learning being a new and abstract concept to comprehend. Finally because this concept is a computer program and not a machine, it is both more difficult and less difficult to work on as a team.

3.4 Subsystem Concepts

3.4.1 Subsystem #1: Locate Aneurysm

The neural network is the system within the locate aneurysm function of the program. It is responsible for learning how to identify aneurysms within a DICOM file with high accuracy. It is crucial that the network is working properly as it is a foundation that the rest of the functions interact with.

3.4.1.1 Design #1: Neural Network

The neural network is the system within the locate aneurysm function of the program. It is responsible for learning how to identify aneurysms within a DICOM file with high accuracy. It is crucial that the network is working properly as it is a foundation that the rest of the functions interact with. Its pros are that it has the possibility of high accuracy. To achieve this however, the network will need a lot of time and effort to be trained. This also requires a lot of training data.

3.4.1.2 Design #2: Aneurysm as a function of Volume

Another way to determine the location of the aneurysm is to find out where the size of the vessel diverges and is no longer uniform. This is likely due to an aneurysm causing a bulge in the vessel wall. This design is able to locate aneurysms with much less processing power than a neural network. This comes at the cost of accuracy. The volume does not have to be due to an aneurysm, it could simply be a vessel getting bigger, or branching off into a bigger parent vessel.

3.4.2 Subsystem #2: Graphical User Interface

A GUI is the front end for any program. It sits in front of the code and allows the user to not view any sort of raw code or information. The GUI must be appealing to the user and functional at the same time. The ease of use will allow anyone to run the code without having experience in programming.

3.4.2.1 Design #1: User Friendly GUI

There are endless possibilities when it comes to creating a user interface. The only aspect that needs to be included is that there is a run button which actually runs the MATLAB code, a button to upload the DICOM file, The aneurysm is displayed on the screen, and the volume is displayed. A few design additions that will be added is a section to save the information found through the code to a database. This information will be saved using an id number and time. All of which will allow the user to look up past information. The basic form for the GUI can be found below in Figure 3.4.2.1.

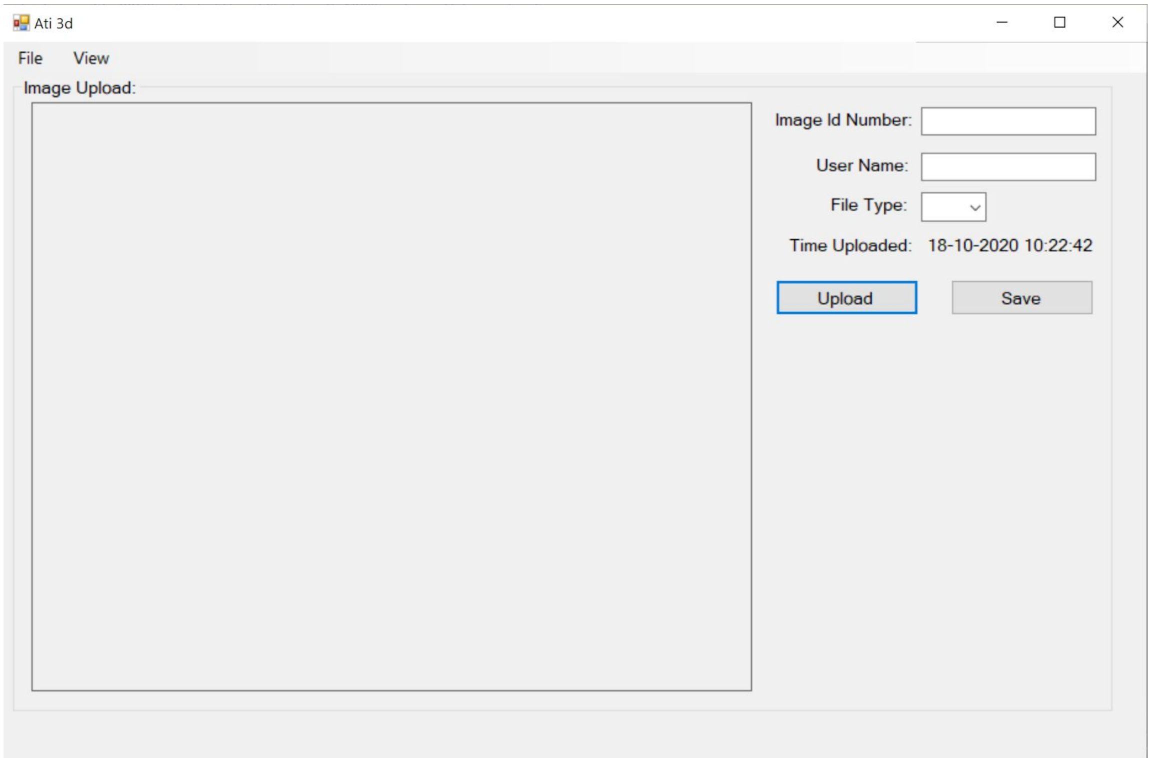


Figure 3.4.2.1: General GUI Form

In Figure 3.4.2.1 above it is important to note that the GUI will always be changing throughout the entire process of building the application. If a new button has to add this can be done at any point in time. This is also true when a button or a function becomes obsolete in the process. It will then be deleted. The process of building a GUI is iterative and new buttons and features will be added when needed.

3.4.3 Subsystem #3: Volume finder

Finding the volume of the aneurysm is a very important feature of the program. It is directly stated in the scope of the problem statement that it must be a function of the program. Over the years there have been many ways that the team has been taught on how to find volume of shapes.

3.4.3.1 Design #1: Integrate area over the length of the DICOM file.

The first method that the team considered was to use integration of area over the total range of slices where the aneurysm is present. This would most likely be a very accurate way of calculating the volume. This would also take much more processing power. On top of the power required the integration also needs the exact or close to areas of the aneurysm at each given slice. This would only be possible if the location finder is able to provide these details once the subject is identified.

3.4.3.2 Design #2: Use sphere approximations

Most aneurysms are likely to be a sphere shape or close to it. By making this assumption the team can take the radius of an aneurysm and use that to create a rough calculation for the volume, It could even be possible to create a coefficient to multiply this volume by to get a closer approximate. For example, if the aneurysm is more pill shaped the coefficient could be 1.2, an increase in overall volume from just a sphere. This design is fairly easy to execute as only the radius and general shape are necessary. Because of this it sacrifices accuracy greatly.

4 DESIGNS SELECTED – First Semester

4.1 Technical Selection Criteria

The selection process for our program has been done through extensive research and talking to the customer about their needs. A QFD helped rank the importance of features and needs for the project. While trial and error was used to see if these topics could even be met. Trial and error also helped our team choose what would actually work within the program and the capabilities of MATLAB. This led to the idea of using a CNN and machine learning to find the aneurysm. Lasty the GUI for the program must be user friendly, which is rather hard to come up with a final design at the moment. To create the most user friendly GUI it will be created throughout the process of making the program. This will allow us to add and take out what features that we do and don't need.

4.2 Rationale for Design Selection

The team has chosen to process DICOM files directly rather than converting them into .tiff or other file extensions. This decision was made to ensure that the integrity of the information is correct. The information lost by converting from file to file is too important for this application. As for the location of the aneurysm the team has chosen to use/create a neural network. This is because of the accuracy that the network has to offer if done correctly. It should be able to process images as many applications of CNNs are image based, the team will just have to find one that works for them, or create their own. They believe that the ability to train the system even while it is working and in the hands of users is very appealing to the team. This leaves the Volume finder function out of the three main functions of code. It has yet to be determined due to the fact that the team is still researching and coming up with the CNN. Depending on the CNNs output the volume will be calculated differently. Currently the teams function for the volume is to integrate the area across the slices where the aneurysm is present. This is to increase accuracy and the team does not believe that finding the area of the aneurysm in each slice will be hard if the aneurysm has been correctly identified.

As for the decision to pick a GUI, the team has chosen to go with what was dubbed the user friendly GUI. This is highly subjected to change as the GUI acts as the transformation from the code to understandable knowledge. This design will most likely be improved and tweaked as the code is created. As stated above the code builds on prior code, and the GUI is built from the code. This means that to fully finalise the GUI the team must first complete the code. For the time being the team has chosen the basic GUI to get the clients and sponsors.

Overall the team's design is focused on accuracy which falls in line with the highest engineering requirement: accuracy. The design is highly subjected to change and the team hopes to constantly change it for the better.

5 REFERENCES

[Include here all references cited, following the reference style described in the syllabus. There should only be one Reference list in this report, so all individual section or subsection reference lists must be compiled here with the main report references. If you wish to include a bibliography, which lists not only references cited but other relevant literature, include it as an Appendix.]

[1] V. Hugo Escobar-de la Garma, M. Zenteno, F. Padilla-Vázquez, D. San-Juan, and A. Cerón-Morales, “Comparative analysis of aneurysm volume by different methods based on angiography and computed tomography angiography,” 2018.

[2] Libretexts, “6.2: Determining Volumes by Slicing,” *Mathematics LibreTexts*, 27-Jul-2020. [Online]. Available: https://math.libretexts.org/Courses/Monroe_Community_College/MTH_211_Calculus_II/Chapter_6:_Applications_of_Integration/6.2:_Determining_Volumes_by_Slicing. [Accessed: 28-Sep-2020].

[3] B. Archer, “Lagrange Interpolating Polynomial,” *from Wolfram MathWorld*. [Online]. Available: https://mathworld.wolfram.com/LagrangeInterpolatingPolynomial.html. [Accessed: 28-Sep-2020].

[4] A. I. Hussein, P. E. Barbone, and E. F. Morgan, “Digital Volume Correlation for Study of the Mechanics of Whole Bones,” *Procedia IUTAM*, vol. 4, pp. 116–125, 2012.

[5] Arthur Duberg September 8th 2020

[6] Mather, J. (2002). Accessing data in DICOM files. Retrieved October 01, 2020.

[7] Matlab. (2013). “Medical Imaging Analysis and Visualization,” *YouTube*, 19-Apr-2013.

[8] Atul. (2018). “Working with MATLAB on DICOM images: An easy tutorial for beginners,” *YouTube*, 08-Jan-2018.

[9] (2016) “Brain Tumor Detection using Matlab - Image Processing + GUI step by step,” *YouTube*, 22-Feb-2016.

[10] The, A. (2014, March 26). Introduction to MATLAB with Image Processing Toolbox - Video. Retrieved October 01, 2020.

[11] 2020. [online] Available at: <https://www.youtube.com/watch?v=1-jURfDzP1s> [Accessed 19 October 2020].

[12] 2020. [Online]. Available: https://www.youtube.com/watch?v=olBl0KIMpAo&t=12s. [Accessed: 19-Oct- 2020]..

[13] “Image Processing Toolbox,” *Image Processing Toolbox Documentation*. [Online]. Available: https://www.mathworks.com/help/images/index.html?s_tid=srchtitle.

[14] “dicomanon,” *Explore collection of DICOM files - MATLAB*. [Online]. Available: https://www.mathworks.com/help/images/ref/dicombrowser-app.html?s_tid=srchtitle. [Accessed: 19-Oct-2020].

[15] "Common Types of Medical Imaging," *Revere Health*, 08-Mar-2018. [Online]. Available: https://reverehealth.com/live-better/common-types-medical-imaging/. [Accessed: 18-Oct-2020].

[16] *DICOM*. [Online]. Available: https://www.dicomstandard.org/. [Accessed: 18-Oct-2020].

[17] Herna′ndez-Hoyos, Marcela, M. Orkisz, P. Puech, C. Mansard-Desbleds, P. Douek, and I. E. Magnin, "Computer-assisted Analysis of Three-dimensional MR Angiograms." [Online]. Available: https://pubs.rsna.org/doi/pdf/10.1148/radiographics.22.2.g02mr03421. [Accessed: 18-Oct-2020].

[18] Settanni C. (2020) Patient 3D Image Scans-Analyzing the Volume of Vessel Defects [PowerPoint presentation]

[19]"Image Labeler," *Getting Started with Object Detection Using Deep Learning - MATLAB & Simulink*. [Online]. Available: https://www.mathworks.com/help/vision/ug/getting-started-with-object-detection-using-deep-learning.html. [Accessed: 18-Oct-2020].

[20] "alexnet," *Pretrained Deep Neural Networks - MATLAB & Simulink*. [Online]. Available: https://www.mathworks.com/help/deeplearning/ug/pretrained-convolutional-neural-networks.html. [Accessed: 18-Oct-2020].

[21] "Select a Web Site," *Choose an App to Label Ground Truth Data - MATLAB & Simulink*. [Online]. Available: https://www.mathworks.com/help/vision/ug/choose-a-labeling-app.html. [Accessed: 18-Oct-2020].

[22] "Image Labeler," *Get Started with the Image Labeler - MATLAB & Simulink*. [Online]. Available: https://www.mathworks.com/help/vision/ug/get-started-with-the-image-labeler.html. [Accessed: 18-Oct-2020].

[23] "Image Labeler," *Getting Started with Object Detection Using Deep Learning - MATLAB & Simulink*. [Online]. Available: https://www.mathworks.com/help/vision/ug/getting-started-with-object-detection-using-deep-learning.html. [Accessed: 18-Oct-2020].

[24] "Image Labeler," *Getting Started with Object Detection Using Deep Learning - MATLAB & Simulink*. [Online]. Available: https://www.mathworks.com/help/vision/ug/getting-started-with-object-detection-using-deep-learning.html. [Accessed: 18-Oct-2020].

[25] "Image Labeler," *Training Data for Object Detection and Semantic Segmentation - MATLAB & Simulink*. [Online]. Available: https://www.mathworks.com/help/vision/ug/training-data-for-object-detection-and-semantic-segmentation.html. [Accessed: 18-Oct-2020].

[26] "Image Labeler," *Get Started with the Image Labeler - MATLAB & Simulink*. [Online]. Available: https://www.mathworks.com/help/vision/ug/get-started-with-the-image-labeler.html. [Accessed: 18-Oct-2020].

[27] "Image Labeler," *Label video for computer vision applications - MATLAB*. [Online]. Available: https://www.mathworks.com/help/vision/ref/videolabeler-app.html. [Accessed: 18-Oct-2020].

[28] “Image Labeler,” *Label ground truth data for automated driving applications - MATLAB*. [Online]. Available: https://www.mathworks.com/help/driving/ref/groundtruthlabeler-app.html. [Accessed: 19-Oct-2020].

[29] “dicomanon,” *Explore collection of DICOM files - MATLAB*. [Online]. Available: https://www.mathworks.com/help/images/ref/dicombrowser-app.html?s_tid=srchtitle. [Accessed: 19-Oct-2020].

6 APPENDICES

6.1 Appendix A: QFD:

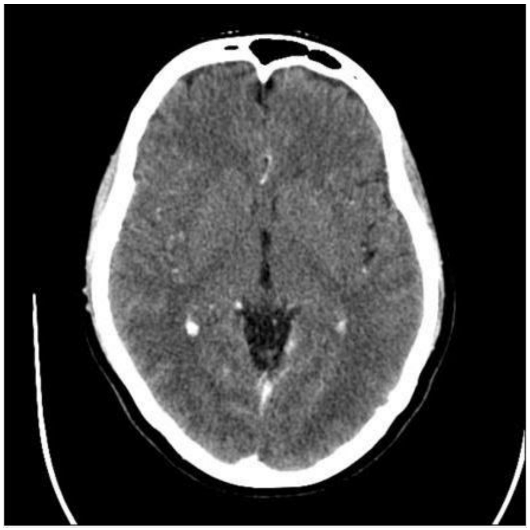
			EGR requirements		
Customer Requirements	**Importance (1-9)**	Volume	Location (x,y,z)	Processing Time	accuracy
Find location of aneurysm	9	3	9	1	9
Find Size of Aneurysm	7	9	3	1	9
Easy to Use	6	1	1	9	1
Accept DICOM Files	5	1	1	3	1
Cost within Budget	7	3	3	3	3
Technical Importance: Raw Score		17	17	17	23
Technical Importance: Relative Weight		0.2267657993	0.249070632	0.1970260223	0.3271375465
Techanical Target Value				60 sec	90%
Upper Target Limit				5 min	99%
Lower Target Limit				5 sec	60%
Units		mm^3	mm	seconds	ratio

6.2 Appendix B: Intensity finder

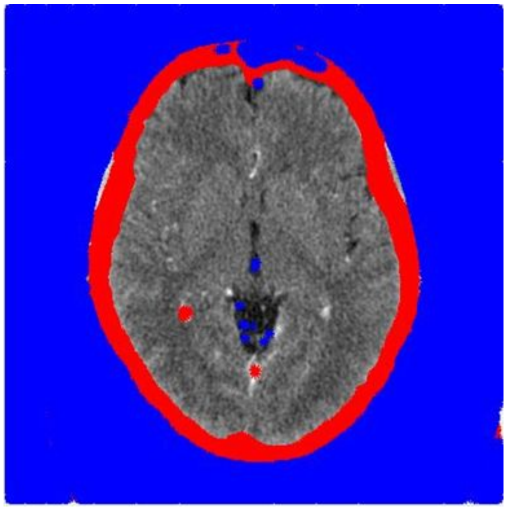
Code:

```
%% Read in Image
Img = imread ('Test Image 1.jpg');

%% Display Image
figure
imshow(Img)

%% Change to B&W
Img1 = rgb2gray(Img);
%% Find high Intensity areas
HImg1 = max(max(Img1));
[iRowH,iColH] = find(Img1 == HImg1);

%imtool(HImg1);
hold on
plot (iColH,iRowH, 'r*')

%% Find low Intensity areas
LImg1 = min(min(Img1));
[iRowL,iColL] = find(Img1 == LImg1);

%imtool(LImg1);
hold on
plot (iColL,iRowL, 'b*')
```

Original image:



New image:



6.3 Appendix C: DICOM Processor

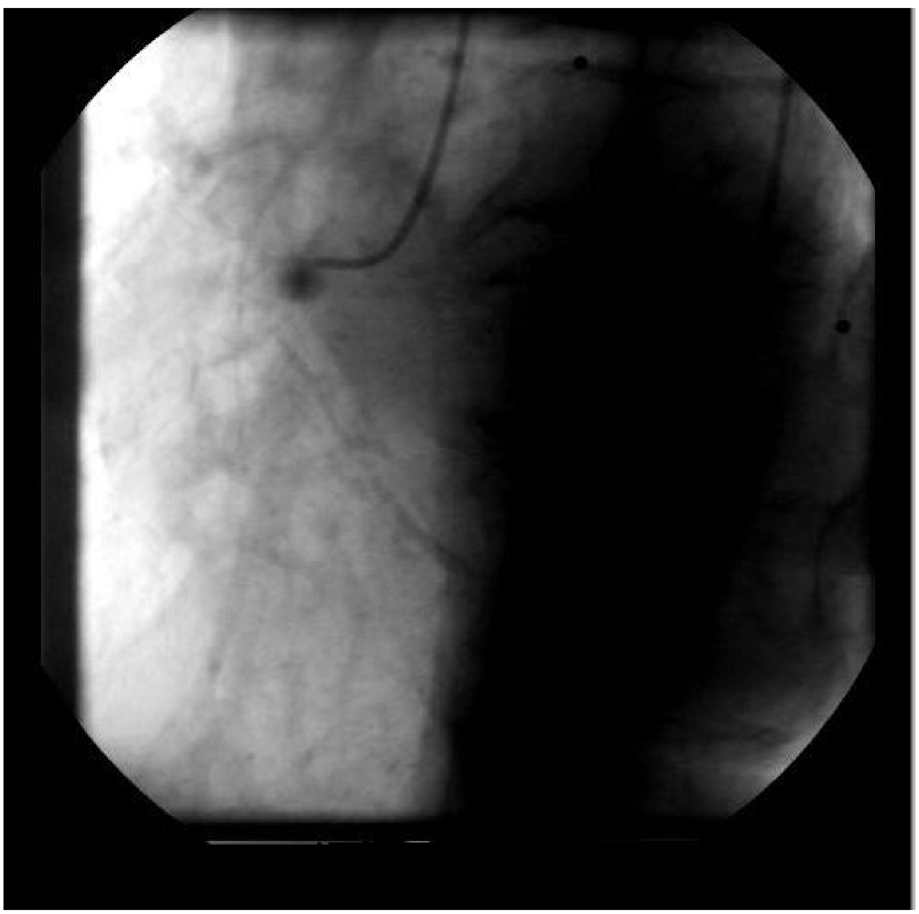
Code:

```
%% close all
clear all; close all; clc; imtool close all;

%% Go to folder

fileFolder = fullfile(pwd, 'DICOM folder');
files = dir(fullfile(fileFolder, '*.dcm'));
fileNames = {files.name};

%% Find info

Info = dicominfo(fullfile(fileFolder,fileNames{1}));

%% Find size from meta data

%voxel_size = [info.PixelSpacing; info.SliceThickness]'

%% Read one file and get size

I = dicomread(fullfile(fileFolder,fileNames{1}));
classI = class(I);
sizeI = size(I);
numImages = length(fileNames);

%% Create array

mri = zeros(sizeI(1), sizeI(2), sizeI(3), sizeI(4));
mri = uint8(mri);


%% Fill in array

hWaitBar = waitbar(0,'Reading DICOM files');
for i=length(fileNames):-1:1
    fname = fullfile(fileFolder,fileNames{i});
    mri = uint8(dicomread(fname));

    waitbar((length(fileNames)-i+1)/length(fileNames))
end

delete(hWaitBar)
```

```
%% Explore image

mri = flipdim(mri, 3);                    % face data forwards
im = mri(:,:,2);                          % pick middle slice for viewing
max_level = double(max(im(:)));
imt = imtool(im, [0, max_level]);
```

```
%% Explore as a montage

%imtool close all
minMRI = min(mri(:));
maxMRI = max(mri(:));
montage(reshape(mri, [size(mri,1), size(mri,2), size(mri,3), size(mri,4)]));
set(gca, 'clim', [minMRI, maxMRI]);
```

Single slice:



Montage: