Interactive software program for automated CT quality assurance testing

BME 402
May 3, 2017

Team Leader: Heather Shumaker
Communicator: Connor Ford
BSAC: Sam Brenny
BPAG, BWIG: Rachel Reiter

Client:
Prof. Timothy Szczykutowicz
Dept. of Radiology & Medical Physics

Advisor:
Prof. John Webster
Dept. of Biomedical Engineering

Abstract

Purpose: To create an open source software program capable of assisting a user in capturing the needed components of a computed tomography (CT) quality assurance (QA) report. The report is generated in a manner that expedites physicist to technician communication in cases where problems with the scanner are discovered.

Methods: The software program was created using MATLAB's graphical user interface platform. The graphical user interface (GUI) has fifteen panels, each facilitating a QA testing protocol by either collecting simple pass/fail or yes/no information, capturing physicist comments and measured values, or analyzing imported images. The GUI produces a LaTeX-formatted text file after the results are calculated which is compiled into a PDF final report. The program is designed to be open source and will accept new users adding new panels to enable customization for different tests.

Results: A graphical user interface was developed to assist in capturing CT report data, analyzing CT QA test results and generating a formatted report that details the testing protocols and results of the QA tests. The program is capable of analyzing images for artifacts, noise, CT number, CT uniformity, gantry tilt, beam width, slice width, and low contrast detectability (LCD), and accepting user input after providing basic information and testing the safety protocols, dose, protocol review, couch, lasers, and the monitor. The resulting data from the analyses is organized and exported into a LaTeX-formatted text file that is used to create the report. Images of laser alignment and couch travel are included in the report to aid in conveying issues to technicians.

Conclusion: The CT QA report-building software combines all the necessary tools for QA testing into a single program, automating calculations, and generating formatted reports with the results and testing procedures. This program aims to improve communication between the medical physicist and service engineers by generating consistently formatted reports with explanations of the testing procedures to improve the reproducibility of each test.

Abstract 2

Introduction 4

Methods/Materials 5

3. Results 5

3.A Quality Assurance Testing Capabilities 5

Basic Information 6

Safety 6

Artifacts 7

Noise 9

LCD - Low Contrast Detectability 9

CT Number 10

CT Uniformity 10

Monitor 11

Beam Width 12

Protocol Review 14

Gantry Tilt 14

Dose 15

Couch 17

Lasers 18

LaTeX 19

4. Discussion 19

5. Conclusion 20

1. Introduction

Computed tomography (CT) is a dominant medical imaging modality used for medical diagnosis and treatment. CT scans combine X-ray images accumulated from multiple angles to create cross sectional images of a target object through digital computer processing [1]. These CT images can provide physicians with valuable information regarding the anatomy and structure of human tissue without the need to make incisions. According to G.T. German, a renowned professor in the field of computed tomography, CT has “revolutionized diagnostic radiology over the past three decades" [2]. Though CT is a fast and revolutionary imaging modality, one caveat to this technology is the dose of radiation the patient is subjected to while receiving a scan. The radiation can be potentially harmful if the dose is too large or if a patient receives a large number of scans. Therefore, it is necessary to ensure accurate dose calculations of clinical CT systems on a daily basis [3].

CT machines require frequent quality assurance (QA) testing to ensure they are producing accurate images at the lowest dose possible for diagnostic and radiation therapy purposes. QA testing involves numerous tests conducted by a certified medical physicist to validate whether the machine is functioning properly or if a certain component requires repair. Various QA tests are required at different time intervals [3]. There are simpler CT QA tests performed daily, more complicated tests performed weekly and monthly, and rather extensive tests performed on an annual basis. Daily tests, for example, may include multiple series of helical and axial scan evaluations with parameters such as detector coverage, speed, rotation time, and slice thickness. Annual tests may include artifact testing, noise, CT uniformity, laser consistency, couch movement and levelness, dose, beam width, and gantry tilt, among others [4].

Once the medical physicist has collected the testing data, it must be compiled into a comprehensive report to be handed off to the vendor’s service technician to perform the necessary adjustments. As of now, the report is either generated manually by the medical physicist, or with the aid of expensive and limited software programs. By providing an open-source software package to automate this step in an otherwise tedious QA process, the time and error associated with QA testing will be reduced. This report details the implementation and testing of an automated report generation tool to improve and expedite communication between the medical physicist and the service technician.

2. Methods/Materials

The *CT Quality Assurance Automated Report Builder* was developed using MATLAB, a programming language developed by MathWorks, and LaTeX, a high-quality typesetting system. MATLAB requires a license to use, while LaTeX is free and readily available to the public. The framework of the program was created by following a spreadsheet of program requirements made by Dr. Timothy Szczykutowicz, an assistant professor in Medical Physics at the University of Wisconsin-Madison school of Medicine and Public Health. The spreadsheet contained information and requirements for the different types of quality assurance tests that need to be performed to ensure proper and accurate CT machine functionality. The program's user interface was developed using MATLAB's guide platform and consists of fifteen "panels" corresponding to each type of test that is performed on the CT machine. Each panel was made individually in MATLAB, and the completed panels were integrated into a user-friendly interface controlled by a drop down bar in the top left corner. An export button in the top right corner is used when all test results have been calculated to export a LaTeX-ready text file to be further processed into the final report.

3. Results

The final software package consists of a single graphical user interface (GUI) capable of loading and analyzing DICOM images, calculating distances and angles in images, and analyzing user input from test results. The program allows for direct user input, comments, and image analysis to iterate through the necessary quality assurance tests. At the conclusion of the program, the user is able to export a text file ready to be used as input into LaTeX, which then creates a final PDF of all the testing information and results. This final report can be passed on to service technicians to make any necessary repairs to the CT machine in question.

The program has combined the capabilities of several existing software programs (i.e. ImageJ and ROI analysis software) used in CT QA testing into a single program. The program also improves upon these capabilities by automating calculations. The program is capable of analyzing and storing information for fifteen quality assurance tests and exporting the results of each test along with a description of the testing protocol.

3.A Quality Assurance Testing Capabilities

The program has a “panel” for each testing procedure. Following are descriptions of each testing panel’s features and capabilities.

Basic Information

The basic information panel (Figure 1) requires user input relating to the basic information of the test. Examples of the information include the testing date, scanner type and location, the facility contact information, and the physicist contact information. This data is saved and incorporated into the testing report.

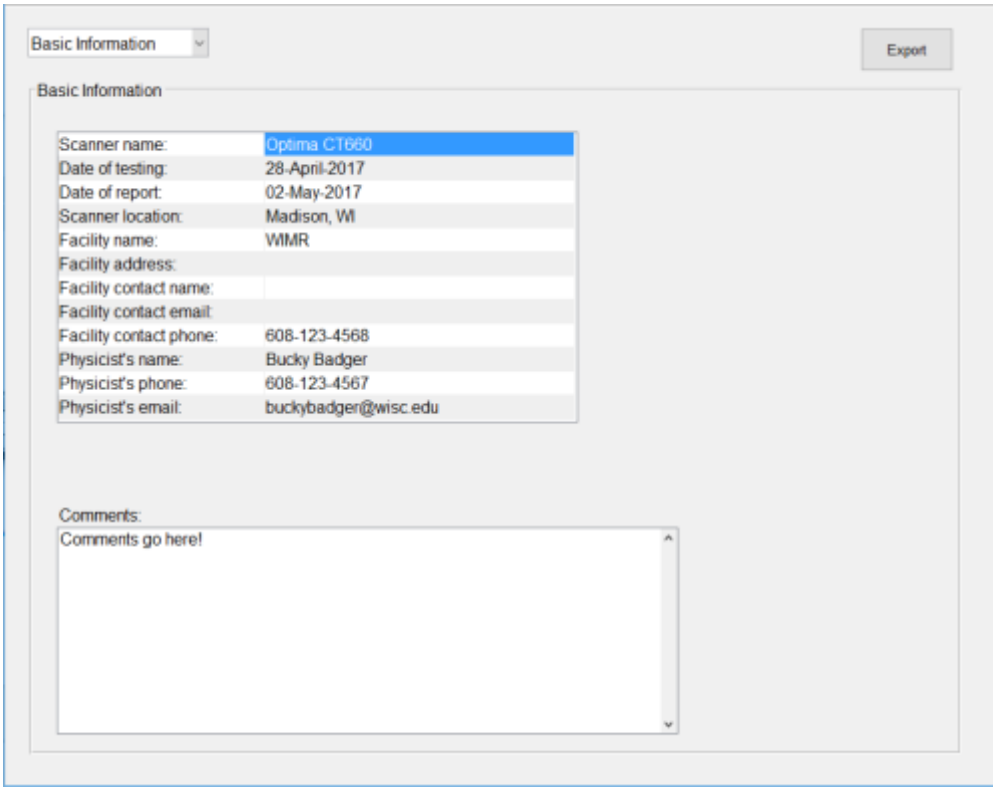

Figure 1. Basic Information Panel. The user enters information regarding the basic information of the QA test.

Safety

This panel (Figure 2) contains a series of questions regarding the room set-up safety precautions. For example, some of these questions confirm the functionality of the “X-ray On” light in various locations, the ability to disable couch movement, and proper intercom performance. Each question requires a Yes/No/NA selection with an optional comment box for additional observations to be noted. The input will appear in a section of the QA report.

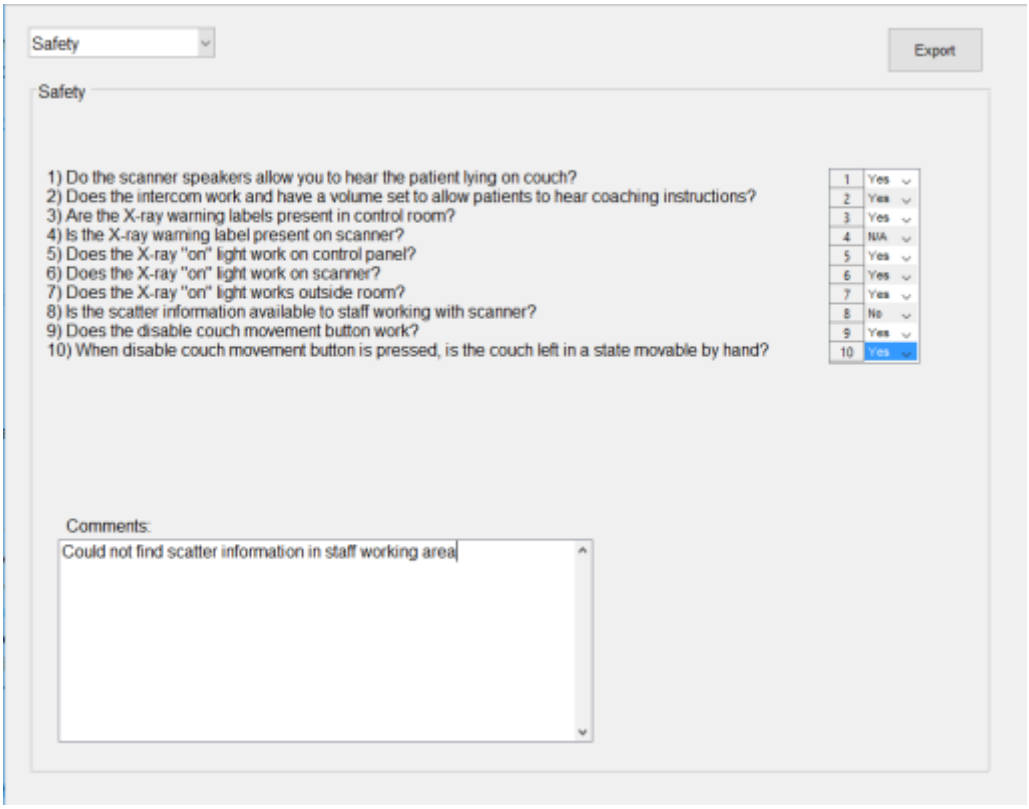

Figure 2. Safety Panel. The user selects Yes/No/N/A from drop down menus regarding safety checks of the machine and room.

Artifacts

The artifacts panel allows the user to view CT images to check for artifacts or enter the results from a previously conducted artifacts test. The user can either enter parameters such as phantom type, phantom scan mode, kV, mA, gantry period, slice thickness, and kernel (Figure 3) or can browse for a CT image from which these parameters are automatically extracted (Figure 4). Additionally, an ROI can be drawn if there is a noticeable artifact within the image. Once an ROI is selected, the ROI standard deviation and the artifact distance-to-isocenter are calculated, which are used for determining artifact sources, and is documented in QA report.

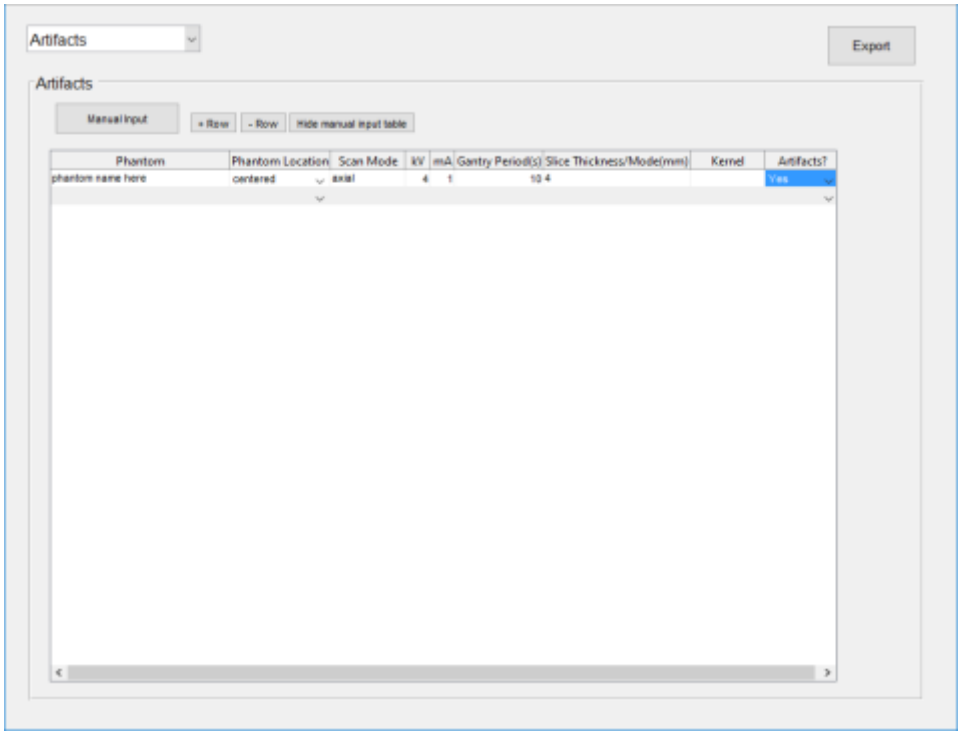

Figure 3. Artifacts Panel Manual Input. The artifacts panel can accept manual user input with a table. The user can increase or decrease the amount of phantoms testing by selecting the “+Row” and “-Row” buttons, respectively.

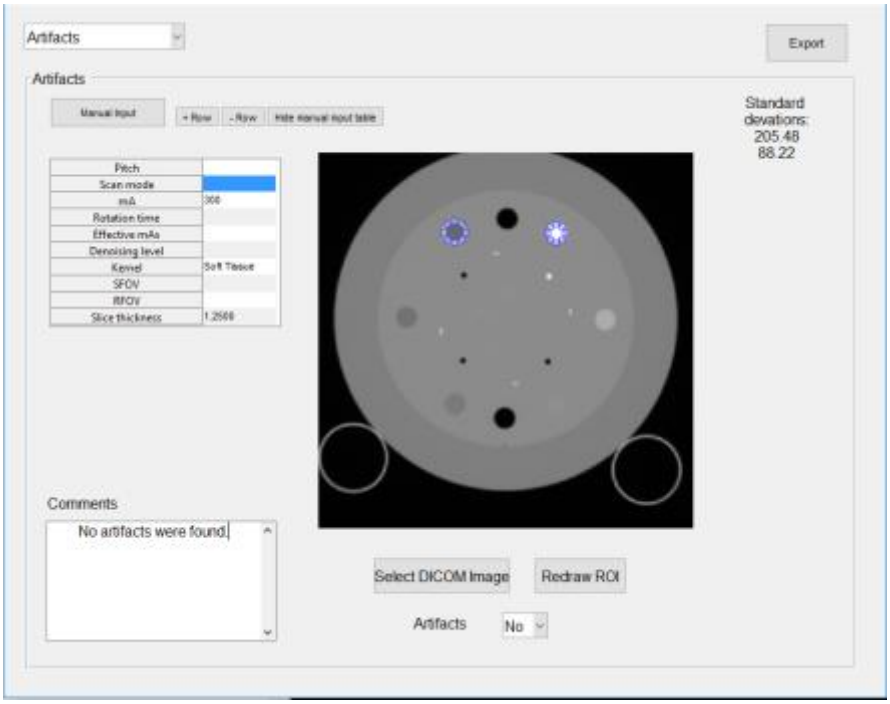

Figure 4. Artifacts Panel Importing Image. The artifacts panel can also import a DICOM image. The user can draw as many ROIs as necessary and the program calculates the standard deviation of each ROI. This panel also strips information from the DICOM headers automatically.

Noise

The noise panel prompts the user to select a CT image. From there, the user can draw ROIs on the image to measure the standard deviation of the pixels in each ROI. This section also features the same DICOM image tag extraction function as the artifacts panel. See the Artifacts section for panel layout.

LCD - Low Contrast Detectability

The low contrast detectability (LCD) panel allows the user to select two CT images (Figure 5). Additionally, the user can manually enter LCD data (Figure 6). Low contrast detectability calculations are performed using the images and the results, along with a plot of the LCD, is saved. The LCD panel also calculates the signal-to-noise ratio (SNR) (Figure 7), which is an important factor that can be used for artifact and noise correction.

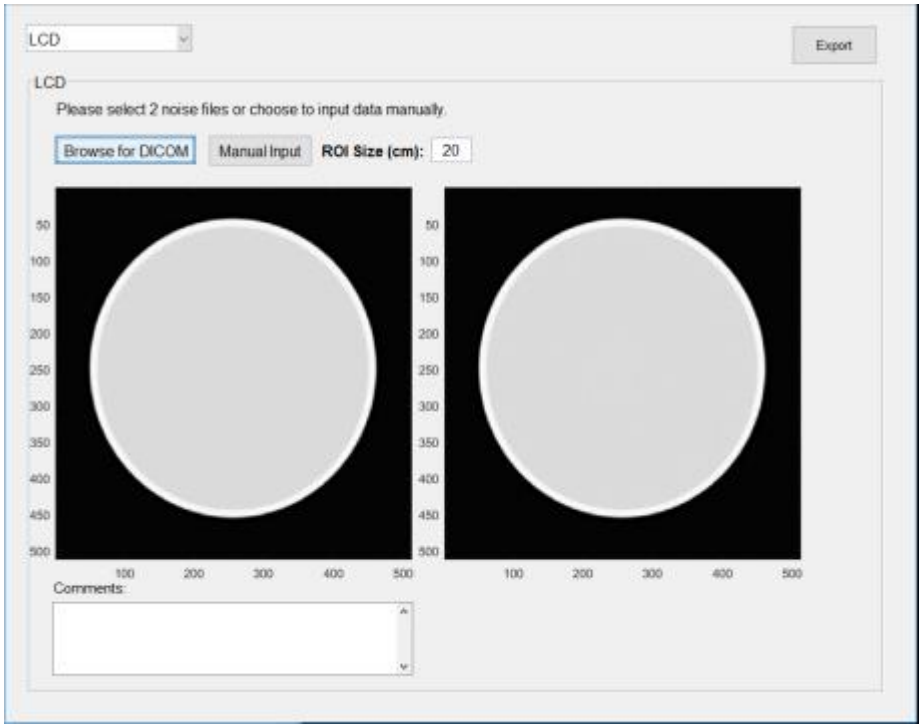

Figure 5. LCD Panel. The user loads two DICOM images to analyze.

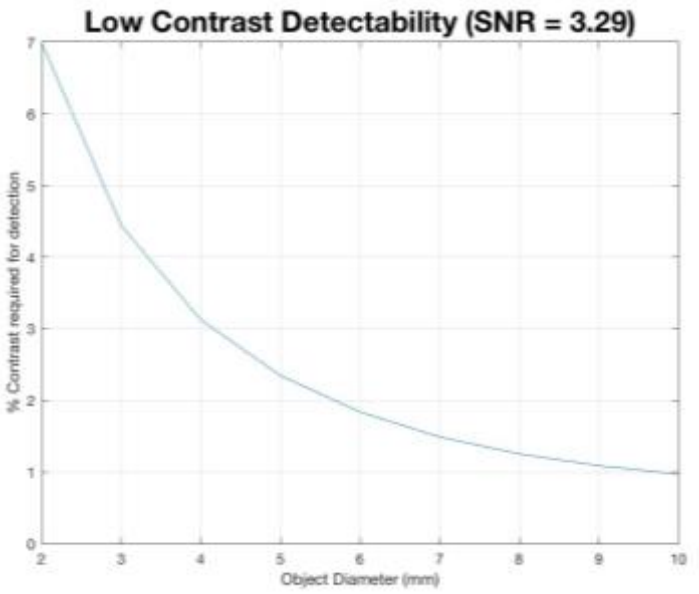

Figure 7. LCD Panel Low Contrast Detectability Plot. The LCD panel produces a plot of the low contrast detectability of the imported images. This plot was produced using two example images and does not accurately represent typical data.

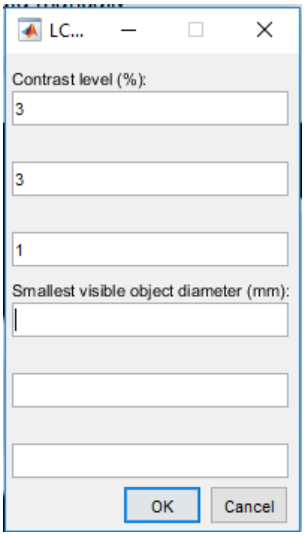

Figure 6. LCD Panel Manual Input. Manual user input is also accepted via a pop-up dialog box.

CT Number

This portion of the code allows the user to browse for a CT image and draw a circular ROI over a portion of the image. The program automatically computes the CT number of the pixels inside the ROI, displays it to the user, and saves it to the program. This panel also features the DICOM tag extraction function. See the Artifacts panel for the CT Number panel layout.

CT Uniformity

The CT Uniformity panel functions similar to the Artifacts, Noise, and CT Number panels by allowing the user to browse for a DICOM image and draw a circular ROI around a region of the image. The program computes the standard deviation of the pixels inside each ROI, displays these to the user, and saves them to the program. See the Artifacts panel for the CT Uniformity layout.

Monitor

The monitor panel (Figure 8) is used to assess the quality of the screen's display. First, a test pattern must be selected from the drop down menu. Options include, ‘Society of Motion Picture and Television Engineers (SMPTE) Pattern’, ‘All Black Image’, ‘All White Image’, or ‘Other’. If ‘Other’ is selected, a new field will appear for the user to specify the test pattern name. A series of questions about the test pattern image must then be answered with a Yes/No/NA selection and optional comment. For example, using the SMPTE test pattern pictured in Figure 9, the user must assess whether the 0%-100% patches of grayscale (outlined in blue in Figure), are all distinguishable. Additionally, the 0/5% and 95/100% patches (outlined in red in Figure 9) must each contain a discernible square within the patch. Lastly, to assess screen luminance, the user must adjust the screen brightness to its minimum level, enter the screen luminance at five locations (the center of the screen and the four corners), and average the recorded values. Screen luminance is measured by computer monitor colorimeter. This process is then repeated with the screen brightness at its maximum level. The program uses the average minimum and maximum luminance values to compute the nonuniformity of the display brightness with Equation 1.

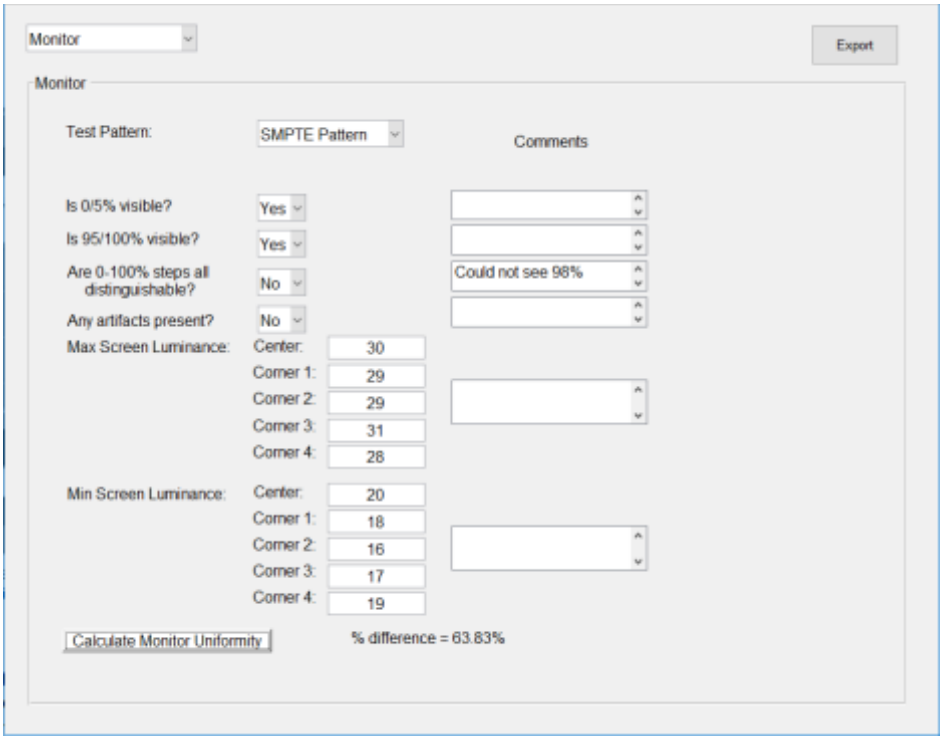

Figure 8. Monitor Panel. User enters information regarding the visibility of certain patterns and the max and min screen luminance of the monitor.

$$\% \, Difference \; = \; 200 \; * \; (L_{max} \; - \; L_{min}) \; / \; (L_{max} \; + \; L_{min})$$

Equation 1. Nonuniformity of display brightness. L_{max} and L_{min} represent the average maximum and minimum screen luminance, respectively.

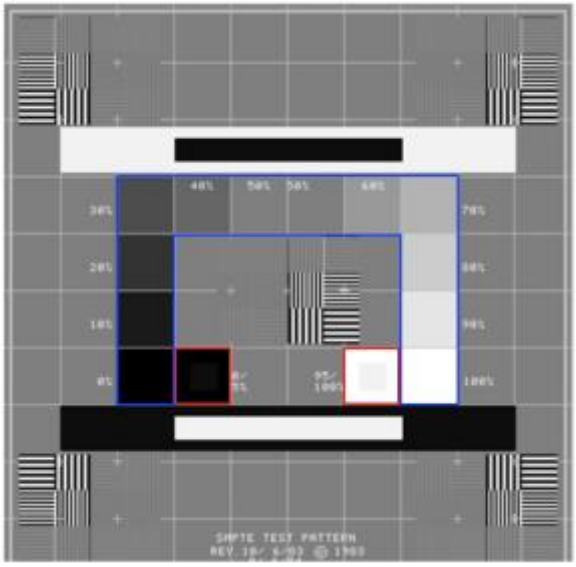

Figure 9. SMPTE Test Pattern. Widely used testing pattern for the evaluation of display systems for medical diagnostic imaging. Most important pattern components for CT QA include the 0-100% patches outlined in blue in addition to the 0/5% and 95/100% patches outlined in red.

Beam Width

The beam width panel allows the user to calculate the width of the X-ray beams. This is done by taking a picture of radiochromic filmstrips that were darkened by the X-ray beams (Figure 10.1). The width of the darkened portion of each strip is calculated by first finding the pixel to millimeter calibration factor. To find this, the user draws a line over a known distance on the ruler in the image (Figure 10.1) and enters this distance. The user then chooses how many beam widths to calculate. After the user draws a line over the darkened region of the film, an intensity profile will be displayed (Figure 10.2). The user selects the maximum and average minimum intensity from this plot. The program calculates the distance between the full width half maximum points on each side of the intensity profile using the calibration factor. Finally, the resulting beam width is displayed to the user and saved to the program (Figure 11).

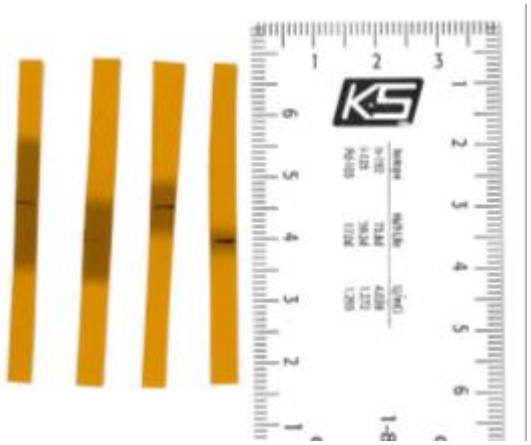

Figure 10.1. Beam Width Measurements. The width of the darkened portion of each strip represents the beam width. The user calibrates the pixels by marking a known distance on the ruler.

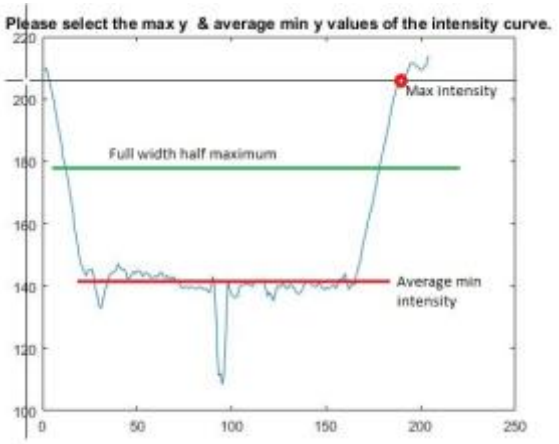

Figure 10.2. Beam Width Intensity Profile. The intensity profile of the beam width of the first strip. This prompts the user to select the max intensity and min intensity.

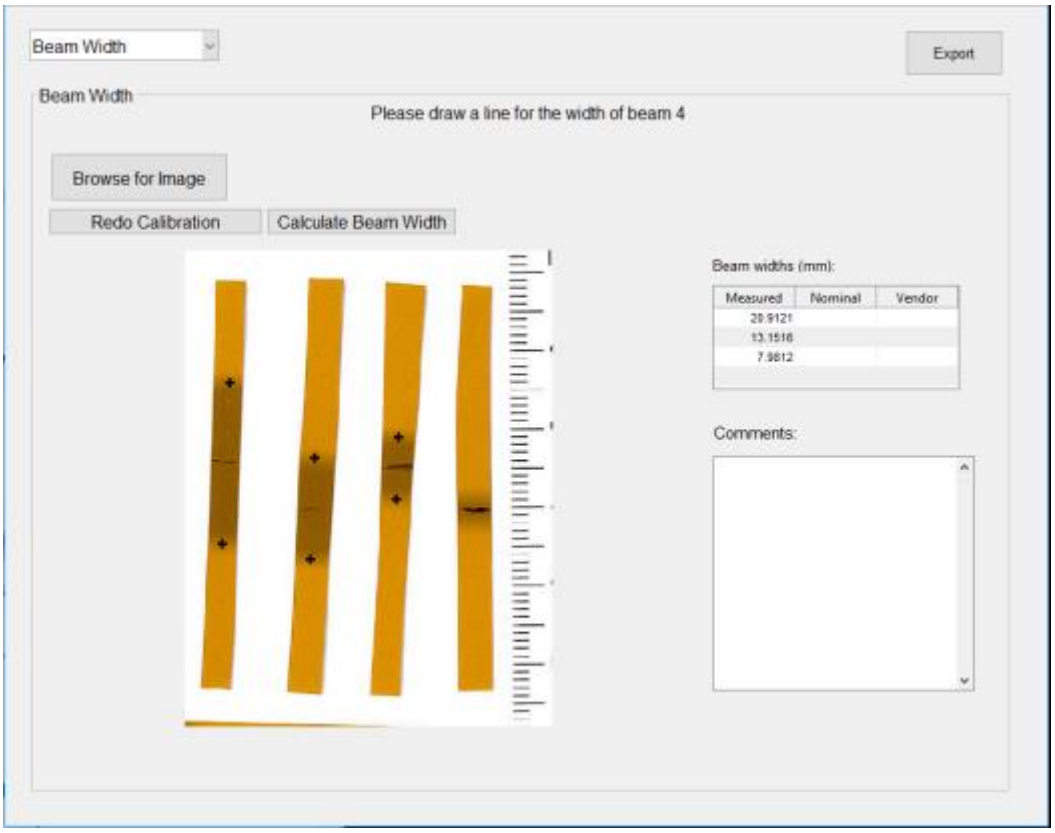

Figure 11. Beam Width Panel. The width of the darkened regions of the film is calculated and displayed in the table on the right. The user can also manually enter beam width data in the table.

Protocol Review

Protocol review is required to ensure the diagnostic image quality is achieved with the lowest dose of radiation possible. Examples of protocols that require annual review include pediatric head, pediatric abdomen, adult head, adult abdomen, high-resolution chest, and brain perfusion [11]. Protocol parameters are programmed into the machine and subject to adjustment whenever under review. This panel is designed to document these changes by allowing the user to enter the protocol name, flag it as 'Okay', 'Needs Attention', or 'Changed during Survey', and indicate the changes implemented and/or needed in the corresponding comment section (Figure 12). This can be performed for an unlimited number of protocols by adding more rows to the table.

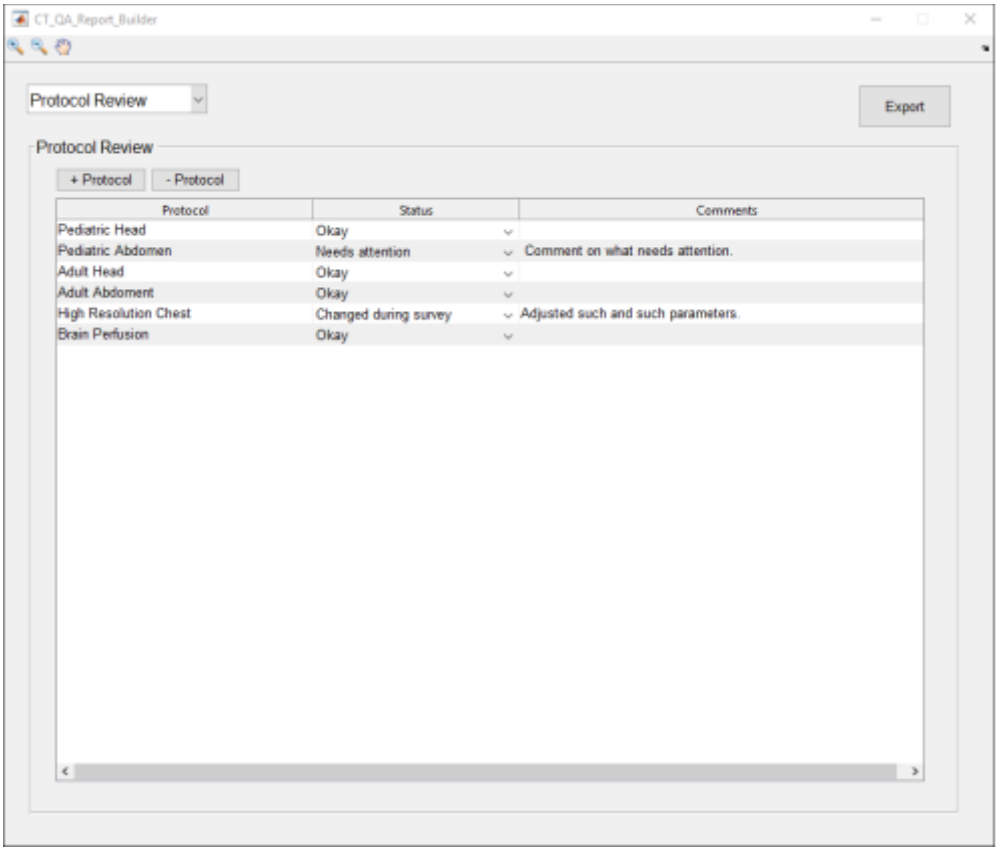

Figure 12. Protocol Review Panel. The user enters the protocol name, selects status from drop-down menu, and optionally can provide comments.

Gantry Tilt

This portion of the code computes the gantry tilt by calculating the angles between five lines that were created on radiochromic film by the X-ray beam. The program prompts the user to draw a line parallel to the top of the paper to find zero degrees. The program calculates the 90˚ angle based on the user-selected 0˚ angle. The user then draws a line over each beam to calculate the angle relative to the 90˚ line (Figure 13). The user can also entry gantry tilt data manually. Lastly, the program displays the angles to the user, calculates the percent difference between the calculated and nominal angle values, and saves the data to the program.

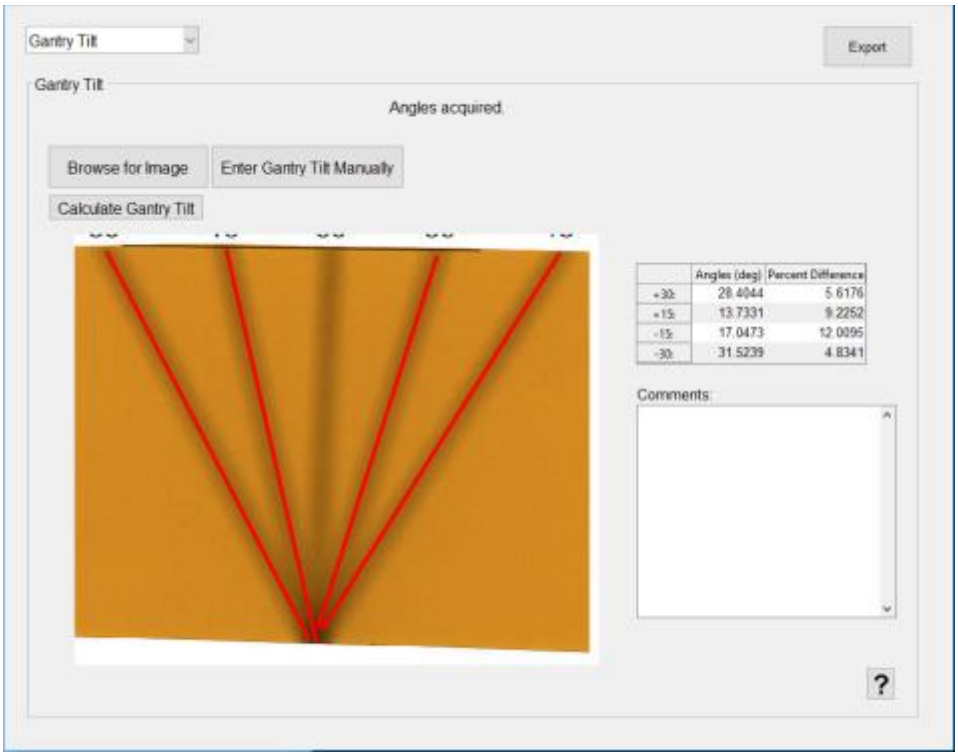

Figure 13. Gantry Tilt Panel. The user snaps a line over the top of the piece of radiochromic film. This represents the 0˚ line. Then the user snaps a line over each beam to calculate the angle between the beam and the 90˚ line. The angles are displayed on the right and the percent difference between the calculated angle and nominal angle is also displayed.

Dose

Dose is evaluated by measuring the radiation exposure with a dosimeter probe. The probe is placed in or on the phantom and a scan is performed. The exposure is then measured by the dosimeter and entered into the program by the user. This process is performed three times with the probe in the center of the phantom and three more times with the probe on the phantom's surface. The center and surface values are averaged separately and used to calculate the CT Dose Index ($CTDI_{100}$) from the averaged

exposure values using Equation 2 and user entered beam width. Both $CTDI_{100,\ center}$ and $CTDI_{100,\ surface}$ are used in the program's calculation of the weighted CTDI ($CTDI_w$) with Equation 3. Lastly, the user enters the $CTDI_w$ displayed on the CT scanner's screen so the percent difference between the scanner and dosimeter dose measurements can be determined (Figure 14).

Note: If the user already has the $CTDI_w$ measurement from the dosimeter, therefore not needing the intermediate calculations, this value can be entered directly.

$$CTDI_{100} = (Average\ Dose * 100)\ /\ Beamwidth$$

Equation 2. $CTDI_{100}$ calculation. The average dose is multiplied by the CTDI subscript, in this case 100, and divided by the beam width to determine CTDI.

$$CTDI_w = \frac{1}{3}CTDI_{center} + \frac{2}{3}CTDI_{surface}$$

Equation 3. $CTDI_w$ calculation. The center and surface CTDI values are weighted differently and summed to determine the weighted CTDI.

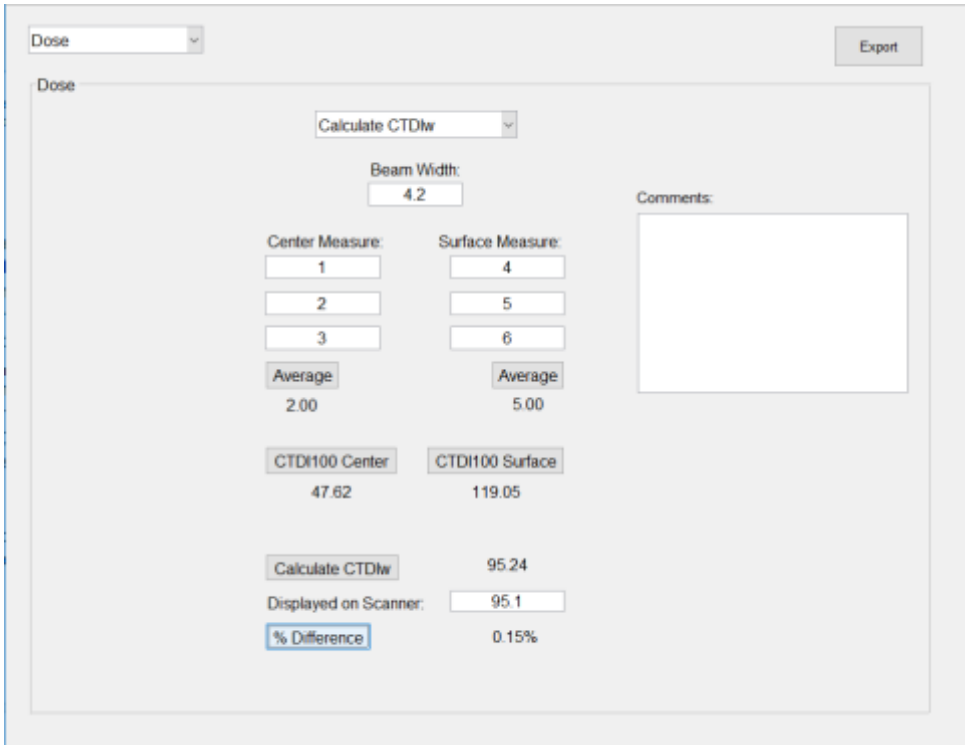

Figure 14. Dose Panel. The user enters center and surface measurements from the dosimeter. The program then calculates the $CTDI_{100,\ center}$ and $CTDI_{100,\ surface}$ as well as the percent difference between the calculated and observed CTDIw.

Couch

The couch panel of the program accepts user input for the nominal and measured start and end positions of the couch as the couch is moved in both the Z and Y directions (Figure 15). The program calculates the total nominal and measured distance moved and calculates the difference between the two. It also takes into account whether the nominal positions were inferior or superior and performs necessary calculations. The input and results are contained within a table on the panel. Additionally, the panel has tables for the couch drift, couch levelness, couch movement, table twisting, and vertical change in the Z-direction.

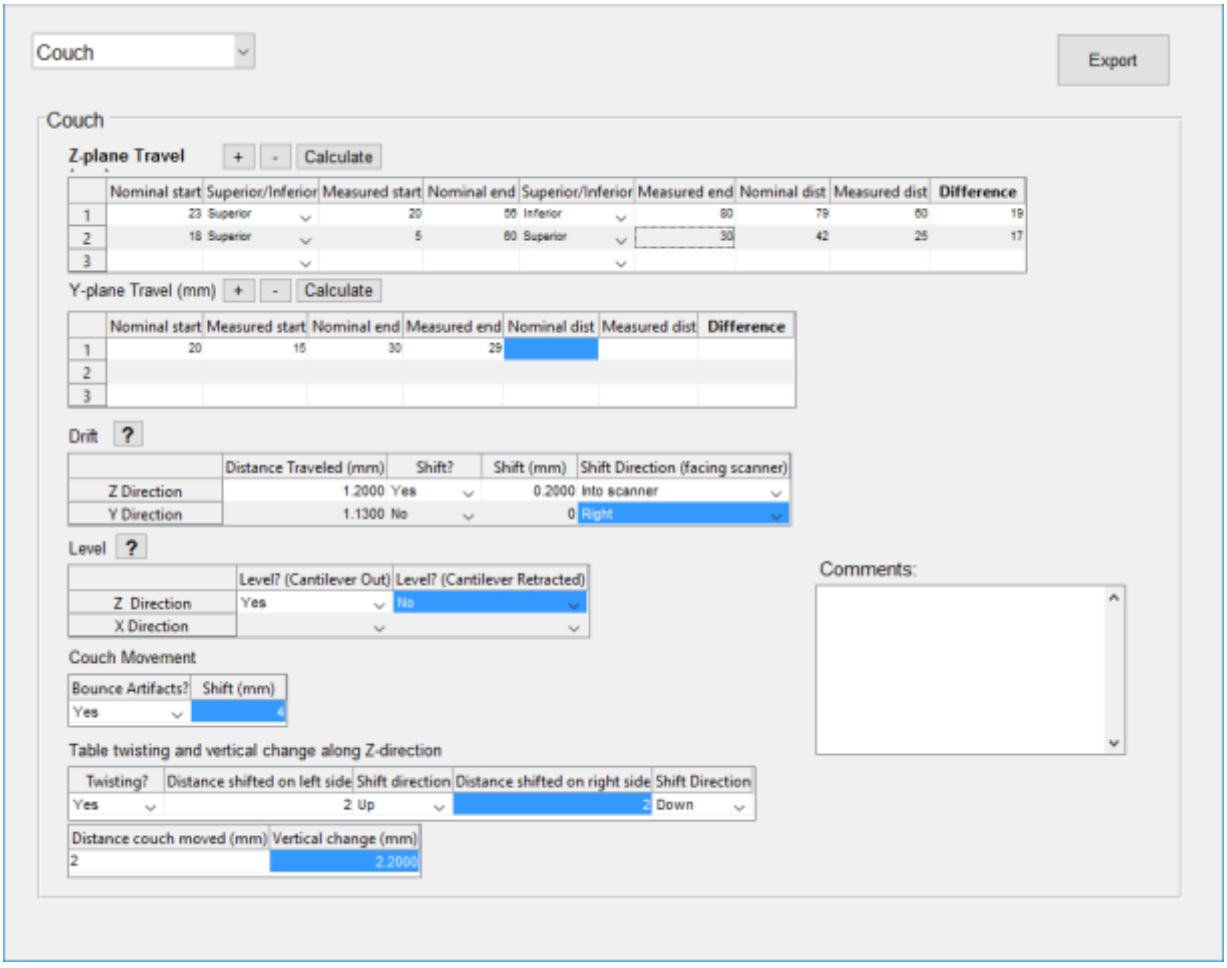

Figure 15. Couch Panel. The user enters information regarding couch movement in the y and z direction as well as drift, levelness, bounce artifacts, and twisting.

Lasers

The lasers panel accepts user input regarding the alignment of the lasers (Figure 16). The user will perform a tissue test (Figure 17) to determine if the lasers are coincident. If the lasers are not coincident, the difference in millimeters is entered into the program. The lasers are also tested for “wobble”, levelness, and orthogonality (Figure 18). These parameters are entered into the program and stored for the final report. The question mark icons on the panel, when clicked, bring up pop-up menus that include images of the tests and explanations of how to perform them, which are useful for different medical physicists to know the exact methods used for a given testing protocol.

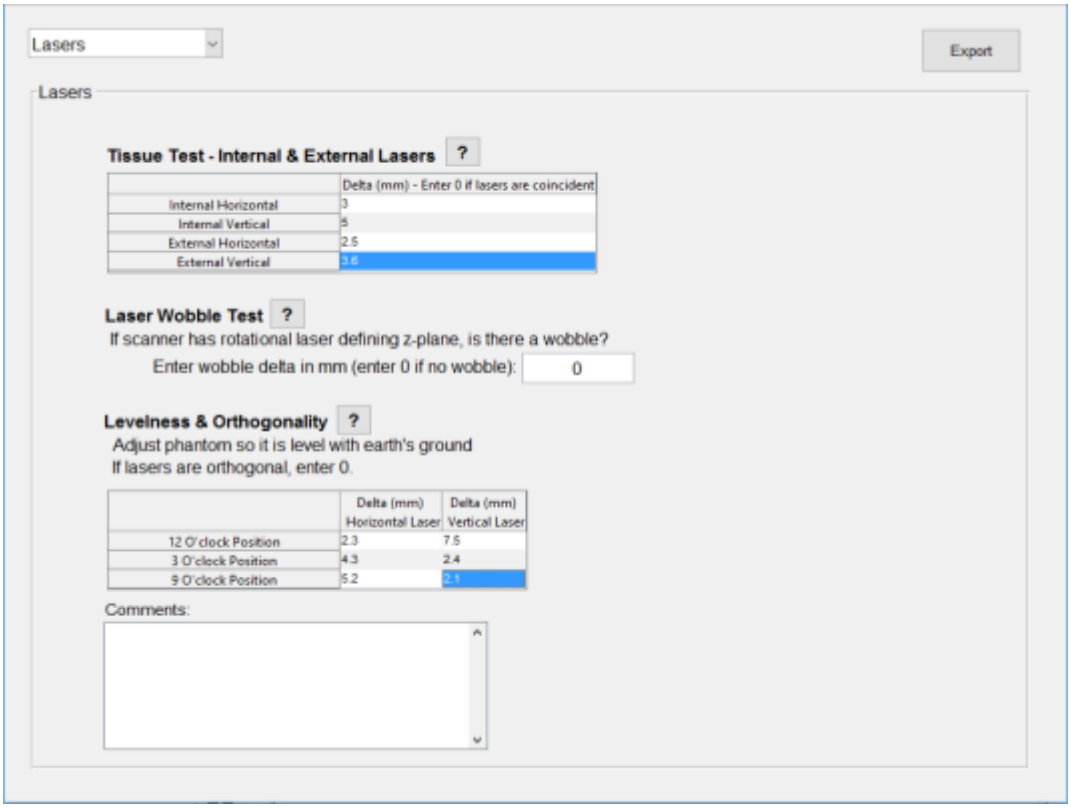

Figure 16. Laser Panel. The user enters measurements regarding laser coincidence, wobble, levelness, and orthogonality.

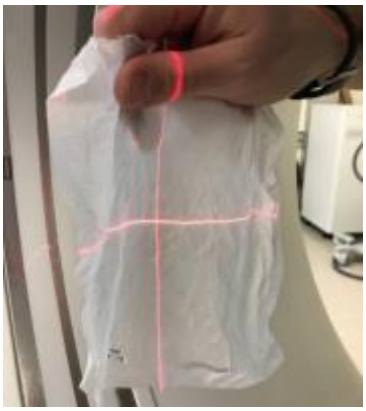

Figure 17. Tissue Test. A test performed to check the coincidence of the lasers. If the lasers are not coincident, the difference in millimeters is recorded.

Figure 18. Laser Levelness and Orthogonality Test. A phantom is used to assess the lasers by checking if they align with markers at 3, 9, and 12 o'clock positions.

LaTeX

LaTeX is a document preparation system that creates richly formatted PDF files. When the user decides to export their results from the report builder program, MATLAB writes all of the user-inputted information and test results into a text file that is properly formatted for LaTeX. This process involves taking in all of the variables from the various panels and placing them into the text file using proper LaTeX syntax so that when the text file is sent through LaTeX, the PDF can be created correctly. If a certain image is desired to be included in the report, the MATLAB code will find the full link to where that image is stored in the user's computer and will print the link in correct syntax to the text file. The benefit of using this system to create the final report is that it will always be formatted in the same way. This creates a standardized format for the report that will help service engineers working on the CT machines. They will always know where to look for the results of a given test, and they will be more familiar with the full set of testing processes given the same procedures will be performed for each test and documented in a consistent manner.

4. Discussion

The CT QA Automated Report Builder's main purpose is to standardize the reporting process for CT quality assurance and increase the efficiency of QA reporting. The program has successfully accomplished this by compiling all data calculations and image processing software into a single program capable of exporting a LaTeX formatted PDF document. The software allows the user to access and update all DICOM header information from an uploaded image as well as all image data. Pixel-to-

distance calculation, ROI evaluation, ROI distance-to-isocenter, and image angle functions were all created to provide a mathematical standard for image data calculations. These algorithms also reduce error between previous manual methods of data acquisition and expedite the computation process. The source code can be easily accessed, reviewed, and revised in an open-source format in order to increase the program's functionality. Additionally, the package will contain a fully compiled executable that can be used on all platforms.

The multiple test-panel layout allows the user to effortlessly transfer from one test to another while saving all information to the program. Each panel allows for the user to provide additional comments to further describe the testing process conducted. The single-GUI layout was designed for global variable accessibility and inter-panel communication. All data is saved until it is exported into a formatted document. Finally, the document formatting increases efficiency and physicist-to-technician communication by providing both a simple "Pass/Fail" for relevant tests as well as raw data and calculations reported in the same document.

In order to improve and ensure the program's functionality, the program was tested among professionals and students to address the program's technical and interface-related issues. The program was distributed to Dr. Tim Szczykutowicz and two medical physicist residents in the Wisconsin Institute for Medical Research to provide feedback on the technical performance of the program. In addition, a medical imaging class at the University of Wisconsin - Madison provided feedback on the overall intuitiveness and usability of the program. This testing protocol facilitated modification via user feedback for a future open-source application. The program has since then been improved in response to the feedback.

5. Conclusion

After thorough analysis of the CT QA reporting process, it was determined that an intuitive and comprehensive GUI system was the most efficient way to standardize QA report building. However, since the software is intended to be open-source, it was necessary to make the final deliverable modular to facilitate a dynamic program that can be applied to all CT QA testing. Overall, the CT QA Automated Report Builder will provide standardized procedures for QA calculations and reporting, as well as consistently update testing protocols to fit developing procedures. Ultimately, using this automated report builder will decrease the time and effort put into quality assurance tests, improve communication between medical physicists and service technicians, and result in a robust and standardized process to ensure reliable and accurate CT performance.

References

[1] "CT Scan," Mayo Clinic [Online]. Available: http://www.mayoclinic.org/tests-procedures/ct-scan/basics/definition/prc-20014610

[2] G. T. Herman, "Fundamentals of Computerized Tomography: Image reconstruction from projection" Springer 2nd edition, 2009.

[3] S. Mutic *et al*., "Quality assurance for computed-tomography simulators and the computed tomography-simulation process," Medical Physics 30 (10). Oct. 2003.

[4] T.P. Szczykutowicz. "CT Scanner Annual Testing: East Clinic UWHC DHO," UW-Madison Dept. of Radiology. Madison, WI. July, 2016.