University of Iowa

Iowa Research Online

Spring 2016

Development and verification of medical image analysis tools within the 3D slicer environment

Jessica LeeAnn Forbes
University of Iowa

Follow this and additional works at: https://ir.uiowa.edu/etd

Part of the Biomedical Engineering and Bioengineering Commons

This thesis is available at Iowa Research Online: https://ir.uiowa.edu/etd/3085

Recommended Citation
Forbes, Jessica LeeAnn. "Development and verification of medical image analysis tools within the 3D slicer environment." MS (Master of Science) thesis, University of Iowa, 2016.
https://doi.org/10.17077/etd.cqznydy9

DEVELOPMENT AND VERFICATION OF

MEDICAL IMAGE ANALYSIS TOOLS WITHIN

THE 3D SLICER ENVIRONMENT

by

Jessica LeeAnn Forbes

A thesis submitted in partial fulfillment
of the requirements for the Master of Science
degree in Biomedical Engineering in the
Graduate College of
The University of Iowa

May 2016

Thesis Supervisor: Associate Professor Hans J. Johnson

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

_____

MASTER'S THESIS

_____

This is to certify that the Master's thesis of

Jessica LeeAnn Forbes

has been approved by the Examining Committee for
the thesis requirement for the Master of Science degree
in Biomedical Engineering at the May 2016 graduation.

Thesis Committee: _____
                  Hans J. Johnson, Thesis Supervisor

                  _____
                  Vincent A. Magnotta

                  _____
                  Edwin L. Dove

                  _____
                  Jatin Vaidya

To my family for always inspiring me to follow my dreams.

To Ben for your encouragement and friendship over all these years.

ACKNOWLEDGEMENTS

ABSTRACT

Rapid development of domain specialized medical imaging tools is essential for deploying medical imaging technologies to advance clinical research and clinical practice. This work describes the development process, deployment method, and evaluation of modules constructed within the 3D Slicer environment. These tools address critical problems encountered in four different clinical domains: quality control review of large repositories of medical images, rule-based automated label map cleaning, quantification of calcification in the heart using low-dose radiation scanning, and waist circumference measurement from abdominal scans. Each of these modules enables and accelerates clinical research by incorporating medical imaging technologies that minimize manual human effort. They are distributed within the multi-platform 3D Slicer Extension Manager environment for use in the computational environment most convenient to the clinician scientist.

PUBLIC ABSTRACT

Rapidly creating specialized medical imaging tools is essential for distributing medical imaging tools to advance clinical research and clinical practice. This work describes the development process, deployment method, and evaluation of modules constructed within the free image viewing software called 3D Slicer. These tools address critical problems encountered in four different clinical areas: quality assessment of large numbers of medical images, automatic corrections of three-dimensional computer representations of the brain, quantification of calcification in the heart using low-dose radiation scanning, and waist circumference measurement from abdominal scans. Each of these applications enables and accelerates clinical research by including medical imaging technologies that minimize manual human effort.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ix

x

xi

CHAPTER 1: INTRODUCTION

Rapid development of domain specialized medical imaging tools is essential for deploying medical imaging technologies to advance clinical research and clinical practice. To accelerate medical image analysis and review in several fields, four semi-automated graphical user interfaces (GUI) were developed within the open-source image viewing software 3D Slicer. This work describes the development process, deployment method, and evaluation of these modules. These tools address critical problems encountered in four different clinical domains: quality control review of large repositories of medical images, rule-based automated label map cleaning, quantification of calcification in the heart using low-dose radiation scanning, and waist circumference measurement from abdominal scans.

The first tool described is the ImageEval 3D Slicer module for assessing the quality of large repositories of anatomical magnetic resonance imaging (MRI) scans. The module requests a list of unevaluated scans from the specified database server, automatically loads a scan for review, populates a questionnaire based on an input configuration file, and pushes the review results to the database server. This module is a redesign of the stand-alone, quality control software tool called BRAINSImageEval to a Python Slicer module with the same functionality.

The next module described in this work, LabelAtlasEditor, was produced as a label atlas correction tool. LabelAtlasEditor incorporates several 3D Slicer widgets into one consistent interface and provides label-specific correction tools, allowing for rapid identification, navigation, and modification of the small, disconnected erroneous

1

labels within an atlas. This project offers an efficient process to drastically reduce the time necessary for manual revision in order to improve atlas label quality.

The third tool calculates the Cardiac Agatston Score (CAS) from cardiac computed tomography (CT) scans. This score is used to estimate the subject's chances of experiencing a cardiac event in the future and aids physicians in deciding on an appropriate course of action. Previous tools for calculating the CAS were developed specifically for CT scans taken at radiation levels of 120 kVp. Part of this work involved mapping the CAS scores to a lower radiation level of 80 kVp. The new tool calculates the scores for both 120 kVp and 80 kVp images to encourage research with lower levels of radiation exposure. This is a semi-automated process wherein the tool separates calcium plaques into one of four main arteries with minimal user interaction.

The final tool described, Waist Circumference, quickly and efficiently estimates patient waist circumference from large numbers of abdominal CT scans. It was designed to advance the use of this measurement as an indicator of obesity and aid in investigating the association between obesity and surgical outcomes. Results from the tool were compared to hand measurements obtained by clinicians using a measuring tape. The results indicated that the two methods were not statistically different.

3D Slicer Environment Description

This section describes the 3D Slicer environment and several useful features for developing medical image analysis tools. 3D Slicer is an extensively used and well-maintained open-source, multi-platform tool for image visualization and computing (Fedorov et al., 2012). Each application described in this work uses the built-in Python programming environment and the QT application framework for developing interfaces.

2

Images, called label maps, can easily be created and manipulated to segment an image into multiple objects within the packaged Editor widget. Three of the modules developed in this work utilize Editor to create and edit label maps. In these cases, a modified version of the Editor widget was included in the interface to provide specific tools useful for each application. Similarly, a modified version of the packaged Label Statistics widget was included in two modules to provide label summary statistics.

The QUIs utilize the packaged image processing toolkit SimpleITK (Lowekamp et al., 2013) for processing images. SimpleITK is a simplified layer built on top of the Insight Segmentation and Registration Toolkit (ITK) (Kitware, 2016). ITK is an open-source, cross-platform system that provides developers with an extensive suite of software tools for image analysis. Several tools described in this work utilize the *Label Shape Statistics* SimpleITK filter to identify objects in label maps and provide label summary statistics. This filter was used to retrieve the maximum intensity value and the number of voxels in cardiac calcium plaques. It was also used to calculate a perimeter of a waist object in an abdominal CT scan.

3D Slicer uses the Medical Reality Modeling Language (MRML) Library API to manage medical image data types including Volumes, Models, Transforms, and Fiducials (Slicer, 2015b). The data types are each stored in a specific MRML node and a collection of MRML nodes comprises the MRML Scene. The MRML Scene manages the nodes via add, delete, find, etc. and provides persistence of the MRML nodes by reading and writing to and from an Extensible Markup Language (XML) file. The MRML XML file was used in these applications to save and later to reconstruct the scene with the included references to the 3D datasets and scene properties.

3

3D Slicer also has a package distribution system called the Extension Manager that provides modules, including three of the modules described here, to users for easy installation. These features allow for easy creation and distribution of tools to the medical imaging community. 3D Slicer has a large and active development community that continually introduces advancements in medical imaging analysis to the application. The framework for each scripted module described in this work was created and published to the Extension Manager using the 3D Slicer Extension Wizard tool (Slicer, 2015a).

4

CHAPTER 2: QUALITY CONTROL OF LARGE MEDICAL IMAGING REPOSITORIES WITH IMAGE EVAL

Introduction

Visual inspection is essential to properly assess the usability of anatomical MRI for image processing. Poor quality images need to be identified and excluded from further processing. Previously, the anatomical images for two Huntingtons Disease studies, PREDICT-HD and TRACK, were reviewed in a stand-alone, quality control tool called BRAINSImageEval. This chapter contains a description of the redesign of this C++ tool as a Python 3D Slicer module called ImageEval with the same functionality.

Rewriting the tool allowed for a dramatic reduction in code complexity due to the availability of the built-in QT functionality and user-interface available in 3D Slicer. A large part of BRAINSImageEval code was dedicated to creating QT buttons, designing QT windows, and interacting with the database management system XNAT. The Python tool is more streamlined, more flexible, and easier to edit than the C++ version. It was developed with the potential to build an imaging analysis suite that could be consistent and reuse Python components and interfaces. As such, the tool will also allow for the reuse of components of this module for other quality control (QC) modules. This document contains images and descriptions of the user interface for both the stand-alone BRAINSImageEval and 3D Slicer ImageEval. It also displays use case diagrams to explain the interaction of the different users of the software.

Methods

This section describes the user characteristics, communication interfaces, and the inputs employed in the Python module. The ImageEval 3D Slicer module is used to

easily assess the quality of T1, T2, and PD MRI scans. The module requests a list of all scans from a database server, creates a list of unevaluated scans, automatically loads a scan for review in the 3D Slicer module, populates a questionnaire, and saves the review results to a database server.

User Characteristics

The users of the QC 3D Slicer module are trained Reviewers that interact only with the 3D Slicer interface to review scans. The reliability of the image quality assessment will be maintained by requiring that the reviewers complete the QC tutorial developed by Kathy Jones and Jacquie Marietta (Jones and Marietta, 2012). This tutorial pictorially describes example assessments at various levels of quality. The user must also have access to the database server used to query and save image information. A Database Administrator will interact with the database to update scan information.

Communications Interfaces

XNAT is the database management system used to query and store image review information. XNAT is an open source imaging informatics platform that enables management, data storage, and organization of medical images. The XNAT server requires a user account and password to view and access data. Upon opening the module, the pop-up box displayed in Figure 1 will request a username and password. The reviewers are required to have an XNAT account and permission to push reviews to XNAT.

Figure 1: The pop-up box requesting the user's username and password to access the requested database.

Inputs

The reviewer must specify on the terminal command line which of the two Huntingtons Disease datasets, PREDICT-HD or TRACK, to review. To do this, the variable "IMAGEEVALGRANT" variable must be set to "TRACK" or "PREDICT" before opening 3D Slicer in the terminal. This is accomplished by typing one of the two following command lines:

- $ export IMAGEEVALGRANT="TRACK"

- $ export IMAGEEVALGRANT="PREDICT"

Input Configuration File

Given that there are two datasets with images and QC reports that are stored in separate locations, an input configuration file must be provided to specify paths. The configuration file should contain the path to the input configuration XML file, the "basePath" location of the images, the database containing the images, and an optional input path to a file containing the specific session and series pairs to review. The variable "IMAGEEVALGRANT" (set to either "TRACK" or "PREDICT") is used to open either

7

the input configuration file in the current directory named

TRACKImageEvalConfigurationFile.csv or PREDICTImageEvalConfigurationFile.csv.

The paths listed in this file, seen in Table 1, can be changed to specify an alternative

questionnaire or image location. When specific images need to be reviewed, the

reviewSessionListPath should point to a simple comma separated values (CSV) file with

each row containing a session and series number separated by a comma.

Table 1: Description of the input configuration file.

ID	Example
imageEvalQuestionnaireFilePath	{PATH}/ImageEvalQuestionnaire.xml
basePath	{DIR/FOR/IMAGES}
dataBase	*https://xnat.hdni.org* OR *https://www.predict-hd.net*
reviewSessionListPath	{PATH}/reviewSessionList.csv

Input Questionnaire XML

The input questionnaire XML file used to create the questionnaire in the stand-

alone BRAINSImageEval tool was reused in the 3D Slicer module. This file can be

found in the Github source code file *ImageEvalQuestionnaire.xml* (Williams, 2012) and

is displayed in Appendix A of this document. It includes yes/no questions pertaining to

whether the image contains normal variants, lesions, full brain coverage, misalignment,

wrap around artifacts, ghosting, inhomogeneity, susceptibility, flow artifacts, and

truncation artifacts. It also includes a range score of 0-10 (0=bad, 10=good) for the

signal-to-noise ratio (SNR), contrast-to-noise ratio (CNR), and the overall quality

assessment. The questionnaire displayed in Figure 2 was populated based on this input

questionnaire XML.

8

Figure 2: Display of anatomical image QA questionnaire within the 3D Slicer module.

Input Scan Information

Communication between the application and XNAT is facilitated through the use of representational state transfer (REST) to both query for unevaluated scans and send QC results to XNAT. The REST uniform resource locator (URL) below is used to

9

retrieve QC information from XNAT to identify unevaluated scans. There are two

XNAT databases identified with their HostURL in Table 2. This REST command

retrieves scan information including the subject, session, date of scan, review status, and

quality. A condensed example of scan information retrieved utilizing this URL can be

seen in Appendix B.

HostURL + "/xnat/REST/custom/scans?type=(T1|T2|PD|PDT2)-(15|30)&format=xml"

**Table 2: Grant identifier and the HostURL for the
XNAT server.**

Grant Identifier	*HostURL*
PREDICT	*https://www.predict-hd.net*
TRACK	*https://xnat.hdni.org*

The functions put, pull, and delete within the Python library *requests* are used to

put, get, and delete data for a specified REST URL. The *delete* function is called to

delete a previous review before the *put* function is called to send the completed review to

the database. Since multiple users can review scans from the same database at the same

time, a random unevaluated scan is selected from the list of unevaluated scans with the

Python library *random*.

Results

Software Interfaces

As illustrated in Figure 3 the previous stand-alone interface displayed the axial,

sagittal, and coronal views of a scan in three separate panels. The user scrolled through

slices of the image via a scroll bar above each panel. He/she could change the contrast of

the image by holding the left mouse button down and moving over the image. A
questionnaire was populated in the left panel for the reviewer to complete. Once the user
completed the questionnaire, he/she selected the "Finish Evaluation" button. That would
save the results to XNAT, clear the current scene, and automatically load the next image
for review. The user did not have any contact with the database and was not required to
manually load images to review.
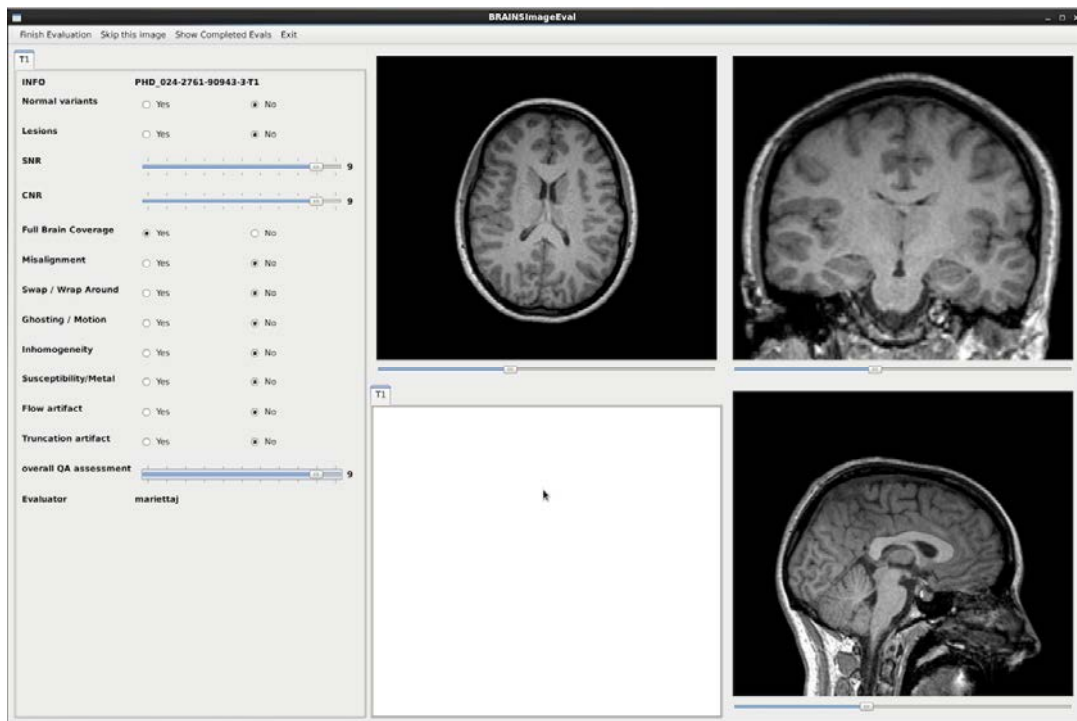
Figure 3: The user interface for the stand-alone version of BRAINSImageEval.

As pictured in Figure 4, the 3D Slicer interface also displays the axial, sagittal,
and coronal view of a scan in three separate panels. The user can scroll through slices of
the image via a scroll bar in the red, yellow, and green bars above each panel. The user
can change the contrast of the image with the use of a range scroll bar in the left panel
under "Display" or by holding the left mouse button down and moving over the image.

11

There is a questionnaire populated in the left panel for the reviewer to complete. Once the user has completed the questionnaire, the "Next" button saves the results to a database, clears the current Slicer scene, and automatically loads the next image for review. The user does not have any contact with the database and is not required to load images to review.
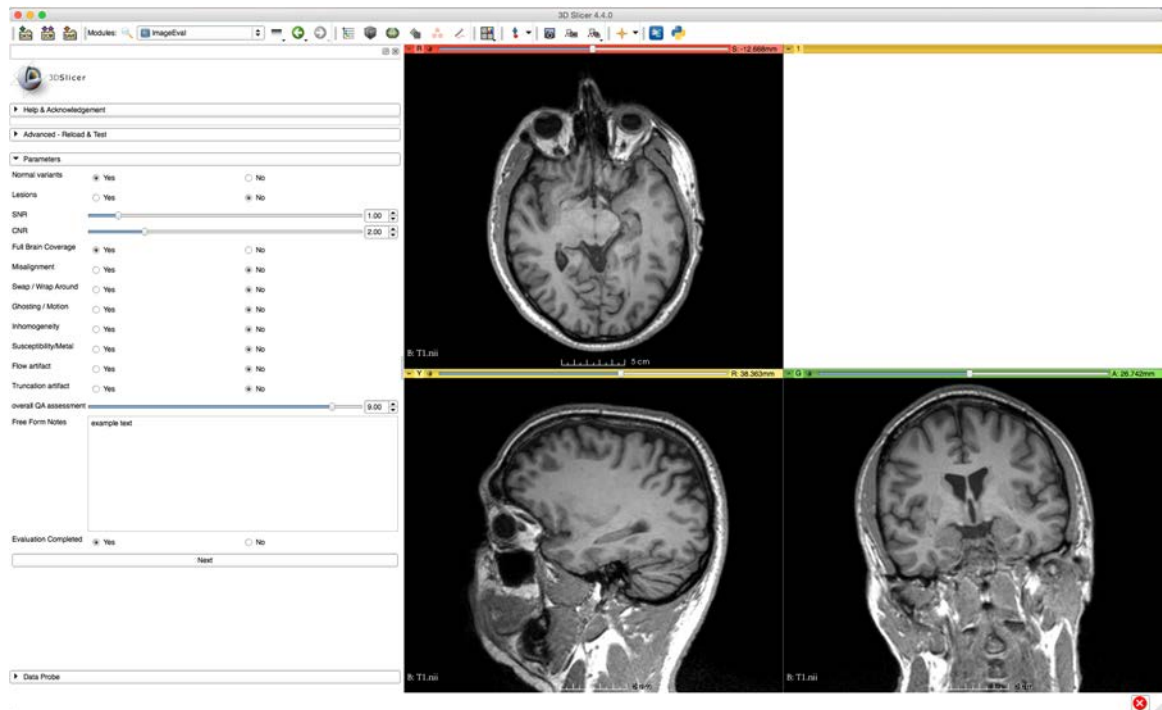
Figure 4: The interface for the Slicer module version of ImageEval.

System Environment
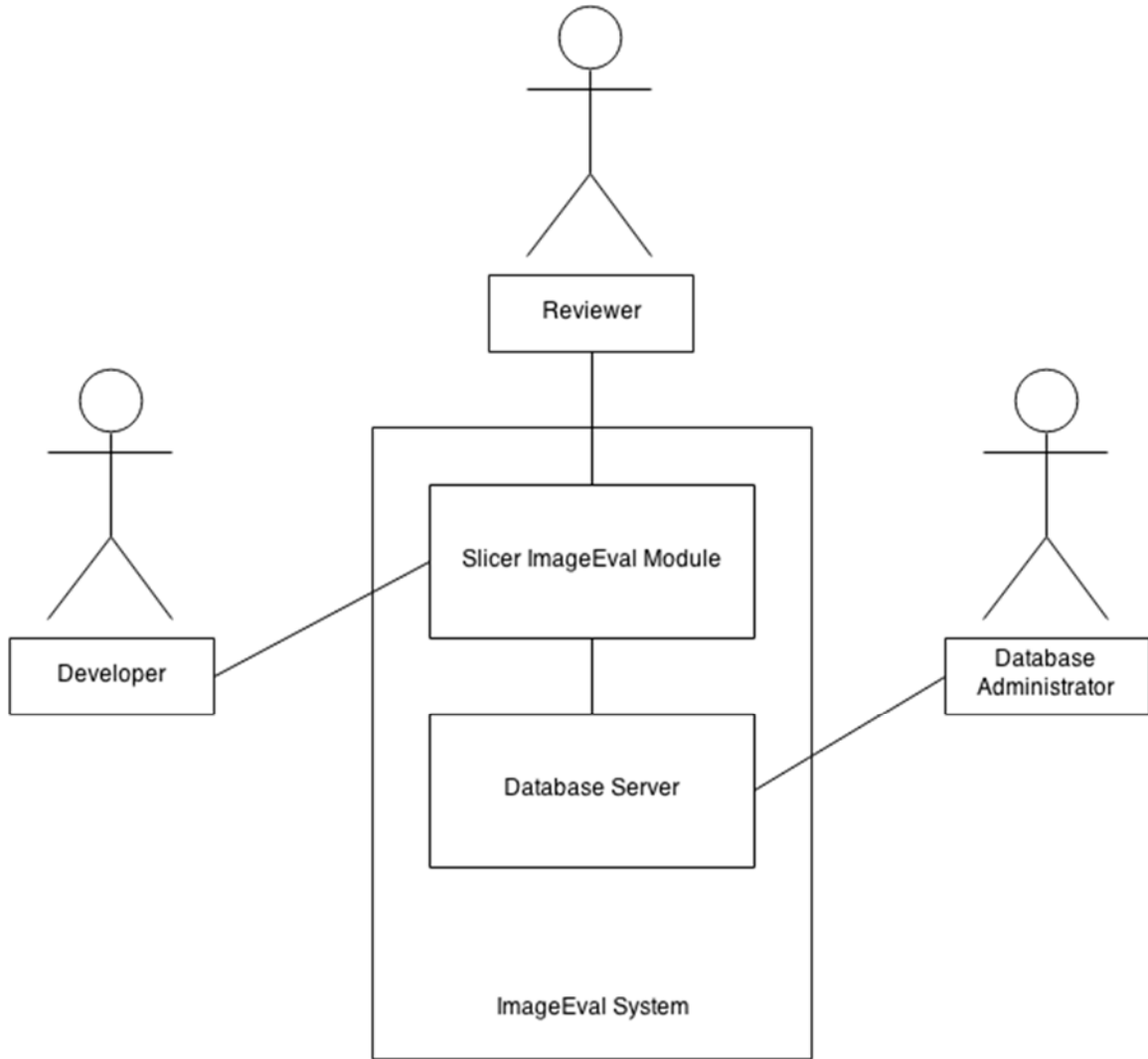
Figure 5: System Environment.

The ImageEval System has three active actors. The Developer and the Reviewer access the ImageEval module through 3D Slicer. Any communication between 3D Slicer and the Database Server is done through a REST URL. The Database Administrator will be able to make changes to the scan information via communication with the Database Server.

13
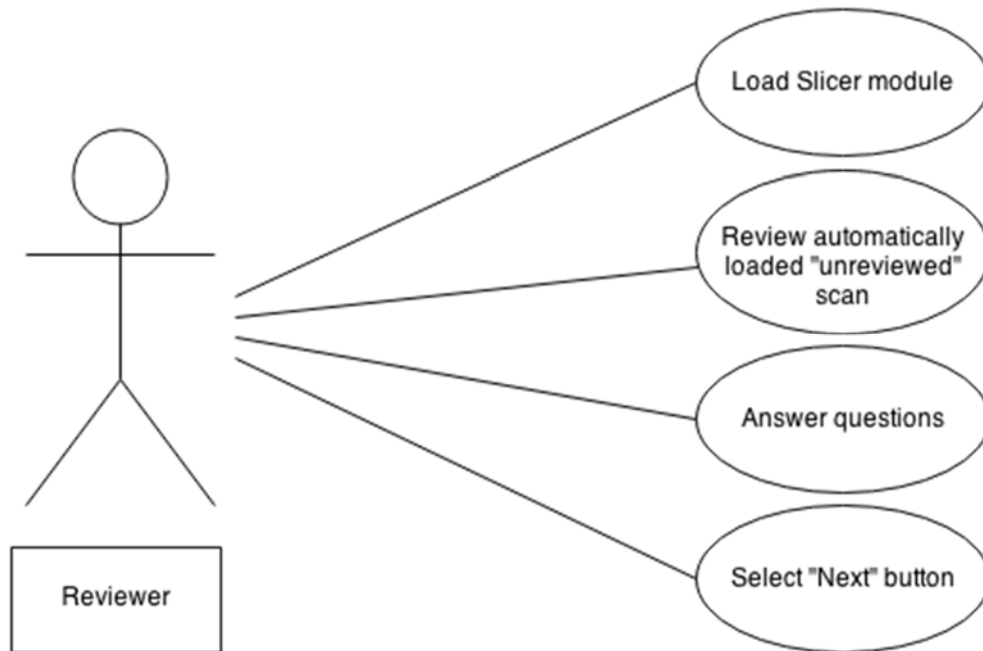
Figure 6: Reviewer Use Case.

Before this use case can be initiated, the Reviewer must open 3D Slicer and the ImageEval module file path must be added to "Additional modules path" in the "Modules" tab of "Settings." The Reviewer then opens the ImageEval module in 3D Slicer where a scan is automatically loaded. Then the Reviewer scrolls through the scan to answer quality assessment questions. Once the review is complete, he/she selects the "Next" button to save the results and load the next scan for review.

14

Figure 7: Database Administrator Use Case.

The Database Administrator interacts with the Database in order to upload scan information, update information as needed, and download scan data. The Database Administrator does not have direct contact with the 3D Slicer module, the Reviewer, or the Software Developer.

Figure 8: Software Developer Use Case.

The Software Developer will interact with the Slicer module if an update is
needed. He/she may also change the input configuration XML file used to generate the
module questionnaire. The Software Developer will not have direct contact with the
Database, the Reviewer, or the Database Administrator.

As seen in Figure 9, there are six objects utilized in the 3D Slicer Module. The "SlicerInstance" is the main object that initiates the cascade of events necessary to review images. Upon opening the module, the "SlicerInstance" initiates the creation of the QC questionnaire panel. This is based on an input template XML file that is parsed by the "XMLSession" object.
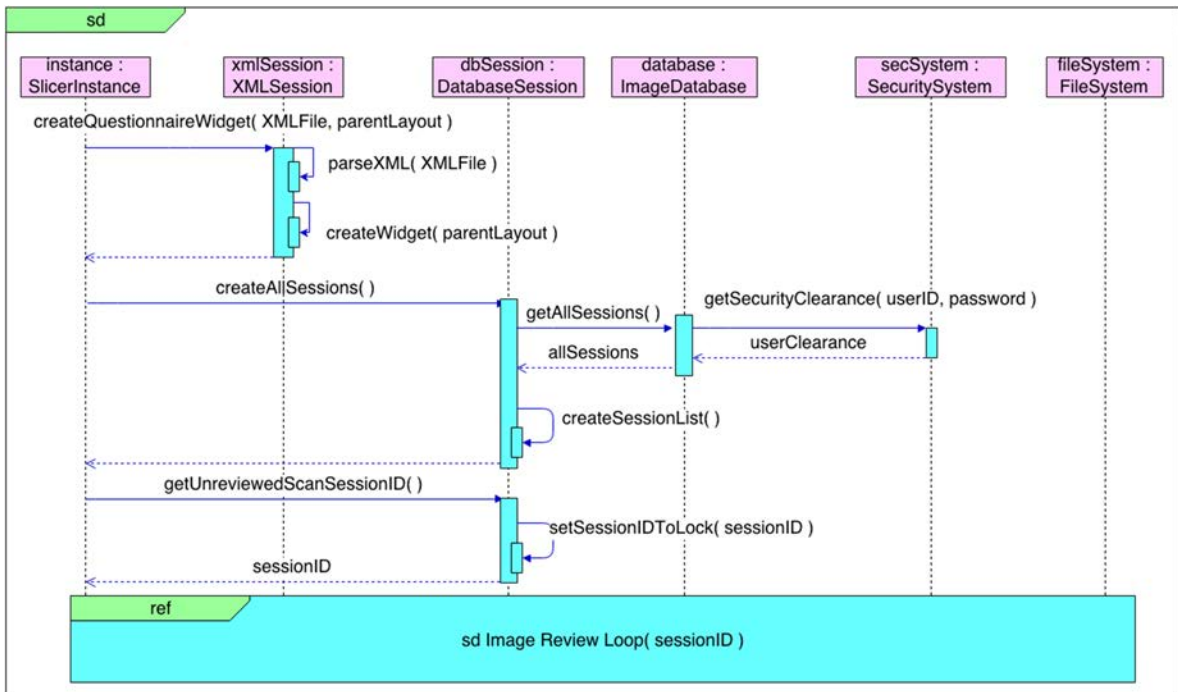
Figure 9: Sequence Diagram for the ImageEval Slicer Module.

Once the questionnaire is generated in the 3D Slicer display panel, a request from the "ImageDatabase" to the "DatabaseSession" object retrieves review information for all sessions. Before the information is returned, user clearance must be obtained from the "Security System" object. Subsequently, a list is created in "DatabaseSession" to store the review information for all scans. This list is then queried in the next call

17

getUnreviewedScanSessionID to find an unreviewed scan. Once an unreviewed scan is

identified, the sessionID is put through the "Image Review Loop" displayed in Figure 10.
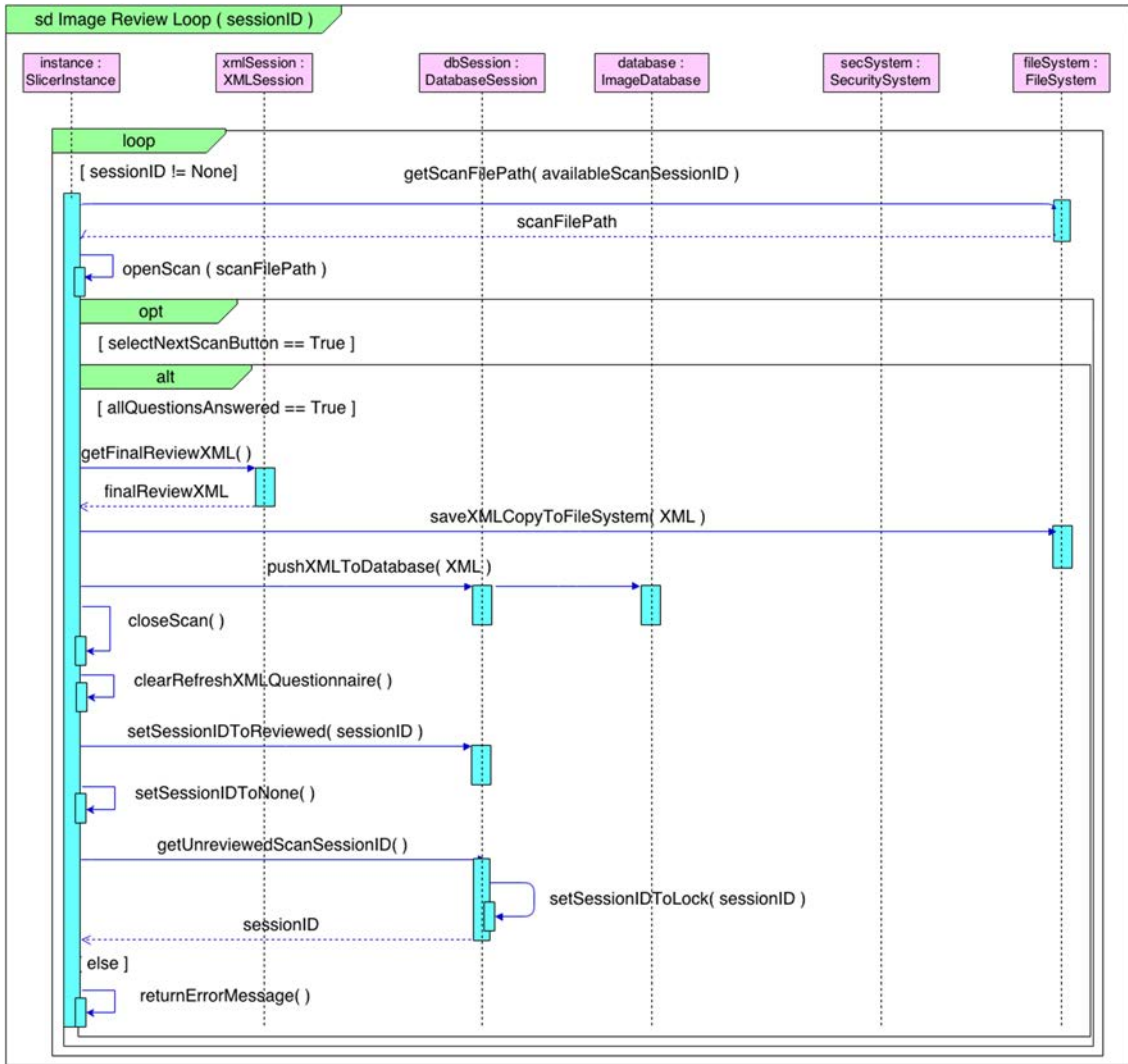
Figure 10: Sequence Diagram of the Image Review Loop.

Figure 10 describes the inner "Image Review Loop" introduced in Figure 9.

Before the loop is initiated, a check is performed to ensure the *sessionID* variable has

been set to the session ID of an unevaluated scan. Then, the first step in the loop is to

determine the scan file path from the "FileSystem." The scan is then opened in the

"SlicerInstance."

18

Once the "Next" button is selected by the user, the "SlicerInstance" checks if all QC questions are answered. If all questions are not answered an error message is returned specifying the questions that need to be addressed. Otherwise, the "SlicerInstance" will request the final review XML from the "XMLSession." Then the final review XML is sent to the "ImageDatabase" via the "DatabaseSession."

In preparation for the next review instance, the current scan is closed and the XML questionnaire panel is refreshed. The sessionID variable is set to "reviewed" in the "DatabaseSession" and the sessionID variable is set to "None" in the "SlicerInstance." The final step is to query the "DatabaseSession" for the next unreviewed scan. If there is another scan to review, sessionID will be set to this new scan's session ID. If all scans have been reviewed, the sessionID variable will remain set to "None" and the "SlicerInstance" will break out of this loop to end the review process.

Data Flow Diagram

In the data flow diagram illustrated in Figure 11, there are two external entities called the Database Manager and the Reviewer. The Database Manager can update scan information located in the "Review Database" data store. The Reviewer can review a scan and later update the scan information in the "Review Database" data store with the finished scan review results. Process 2 initiates Process 3 to create the "All Sessions" data store with information from the "Review Database" data store. The "All Sessions" data store is used by Process 4 to create the "Current Session Information" data store. Then the "Current Session Information" data store is utilized to return information about the current session to Process 4 and Process 5.

19
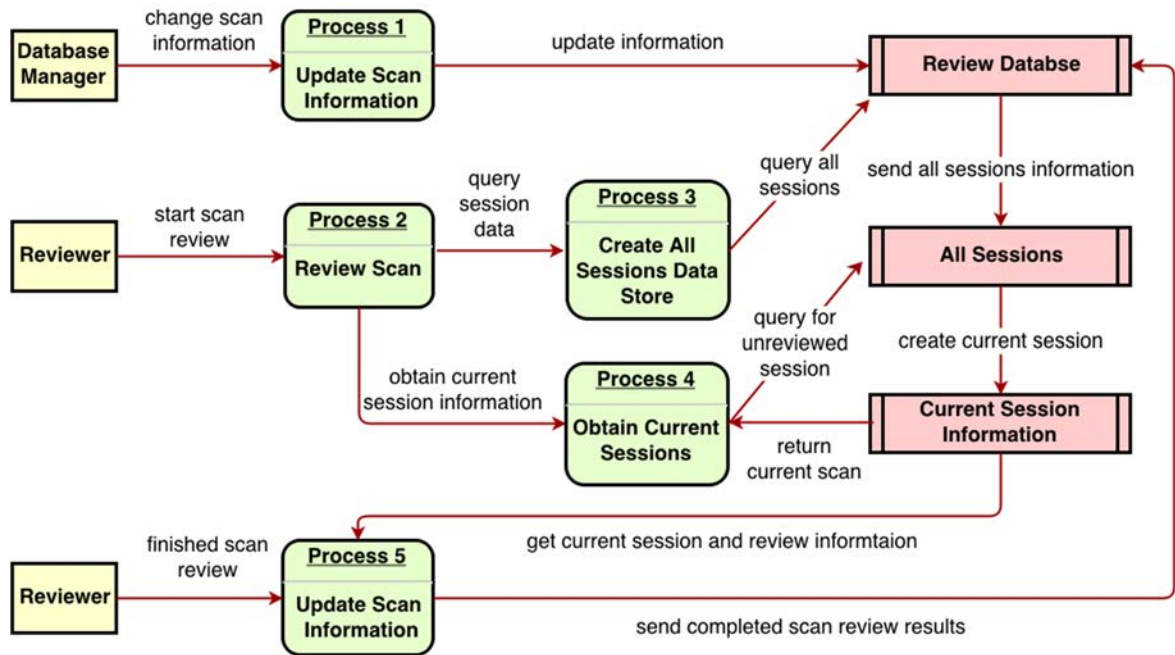
Figure 11: Data Flow Diagram for the ImageEval Slicer module.

Discussion

The 3D Slicer module has the same functionality as the stand-alone

BRAINSImageEval and serves as its replacement. The ImageEval module is compatible

with several platforms, including Windows, Linux, and Mac Os X, whereas the

BRAINSImageEval was designed for use only on Mac Os X and Linux. This allows for

greater versatility. The program, rewritten in Python, is more streamlined and more

flexible than the C++ version. It will also allow for the reuse of components of this

module for other QC modules in an imaging analysis suite. The ImageEval module is

stored in a public Github repository and is available to the reviewers (Forbes, 2015). It

has been used to review several hundred T1 and T2 images in the PREDICT-HD and

TRACK cohorts.

20

CHAPTER 3: AN OPEN-SOURCE LABEL ATLAS CORRECTION TOOL AND DEMONSTRATION ON HUNTINGTONS DISEASE WHOLE-BRAIN MRI ATLASES

Abstract

The creation of high-quality medical imaging reference atlas datasets with consistent dense anatomical labels is a challenging task. Reference atlases have many uses in medical image applications and are essential components of atlas-based segmentation tools commonly used for producing personalized anatomical measurements for individual subjects. The process of manual identification of anatomical regions by experts is regarded as a gold standard. However, manual identification is usually impractical due to its labor-intensive nature. Further, as the number of regions of interest increases, manually created atlases often contain many small, inconsistently labeled or disconnected regions that need to be identified and/or corrected.

This project proposes an efficient process to drastically reduce the time necessary for manual revision with the goal of improving atlas label quality. LabelAtlasEditor, a SimpleITK-based open-source label atlas correction tool is introduced. This tool is distributed within the image visualization software 3D Slicer and incorporates several 3D Slicer widgets into one consistent interface. It provides label-specific correction tools, allowing for rapid identification, navigation, and modification of the small, disconnected erroneous labels within an atlas. The performance of LabelAtlasEditor is demonstrated through its application to the task of improving a set of 20 Huntingtons Disease-specific multi-modal brain atlases. After the correction of atlas inconsistencies and small, disconnected regions, the number of unidentified voxels for each dataset was reduced on

21

average by 68.48%. Additionally, the advantages and limitations of automatic atlas correction are presented.

Introduction

The study of human brain anatomy is important in clinical studies of normal brains as well as in studies of neurodegenerative disorders such as Huntingtons Disease (HD), Alzheimer's disease, and Parkinson's disease. A precise assessment of the volumetric characteristics of brain structures may provide a non-invasive means to monitor the treatment effects of clinical intervention. During the last decade, many studies have collected series of imaging data to better understand the brain. These studies of structural brain magnetic resonance imaging (MRI) have provided important understanding of healthy development (Herting et al., 2014; Sullivan et al., 2011; Treit et al., 2013), normal aging (Mungas et al., 2005; Resnick et al., 2003; Risacher et al., 2010; Scahill et al., 2003; Tang et al., 2001), and disease progression (Ahdidan et al., 2011; Li et al., 2015; Tabrizi et al., 2012; Takahashi et al., 2012; Weiner et al., 2012).

Atlas-based segmentation is a commonly used approach (Cabezas et al., 2011) that identifies regions of interests (ROI) by propagating atlas labeling to a target image. More recently, multi-atlas labeling approaches, instead of single-atlas labeling, have gained popularity for their superiority in segmentation quality (Kim et al., 2015). Naturally, the performance of this atlas-based segmentation largely depends on how well the atlas structures are defined and the similarity between the atlas and the research population.

Currently, there are limited atlas labels available in the field (Mori et al., 2005; Tzourio-Mazoyer et al., 2002). While manual identification of brain structures from MRI

22

is considered the gold standard, the creation of a population or research-specific atlas is limited by its labor-intensive and time-consuming nature. Furthermore, if one chooses to use multi-atlas labeling approaches, which show increasing evidence of improved segmentation quality (Kim et al., 2015), manual approaches for label atlas creation are almost always impractical for ongoing research. The choice of atlases for brain research is therefore often limited to what is already available, despite that there is a danger of bias when using unrepresentative atlases for the study of interest.

There has been a concern about the accuracy and consistency of manual traces for brain MRI structures in the neuroimaging community. Via a collection of open-source tools called Open Atlas (Lorensen, 2015), it has been recognized that manually identified brain structures can be inaccurate due to the large numbers of small, disconnected regions (islands) that are biologically invalid. These inaccuracies of manually identified structures are mainly a result of the current limitations of the tracing environment, wherein experts segment convoluted three-dimensional structures within two-dimensional planes. These label atlas errors may have been underestimated previously since they are difficult to recognize in a two-dimensional display.

This chapter proposes a prototype application for creating and/or improving a set of MRI label atlases by incorporating prior information as well as spatial heuristics with highly automated procedures. The primary goal of this project is to provide an efficient procedure that reduces the editing time to a few hours while providing valid segmentation results. This proposed application, LabelAtlasEditor, is freely distributed with 3D Slicer, and the proposed approach is expected to require less time for manual intervention in creating and/or improving MRI label atlases.

23

The label atlas correction approach is demonstrated by using a dataset of 20 subjects spanning a range of Huntingtons Disease progression. These subjects were selected from the PREDICT-HD dataset that contains pre-symptomatic gene-positive subjects collected during a 10-year time period (Paulsen et al., 2006). This article walks through the steps used for identifying the regions of interest with a semi-automated tool, details the method used for automatically removing small disconnected regions of voxels, and demonstrates a technique for cleaning the regions of interest with prior probability maps.

Methods

This section first describes the dataset used for this study. This is then followed by an illustration of the MRI pre-processing method applied. The section is concluded with a description of the set of developed label atlas correction applications and the proposed procedure for developing a reliable set of label atlases.

Dataset

The HD label atlas candidates were carefully selected from the PREDICT-HD database. The 20 MRI subjects selected consisted of 10 males and 10 females between the ages of 28.4 and 68.1 years. These subjects span the HD disease index of CAG repeat lengths: control, low, medium, and high. Scans were collected from three types of scanners: GE, Siemens, and Phillips. All datasets are multi-modal with most containing multiple T1-weighted (T1-w) and T2-weighted (T2-w) images from repeated data sessions. Demographic variables (gender, age, and CAG repeat length) are reported in Table 3 and the imaging acquisition parameters for the scans are reported in Table 4.

Table 3: HD Atlas set demographics

Manufacturer	Gender	CAP	Age	Summary	
GE	F	Med	28.4		
GE	F	Med	46.5	Manufacturer	
GE	F	High	56.4	GE	7
GE	F	High	58.0	Philips	6
GE	M	Low	31.3	Siemens	7
GE	M	Cont	36.0	Total	20
GE	M	High	40.5		
Phillips	F	Med	36.0		
Phillips	F	High	44.7	Gender	
Phillips	F	Cont	47.3	Male	10
Philips	F	High	59.2	Female	10
Phillips	M	Cont	43.6	Total	20
Phillips	M	High	46.8		
SIEMENS	F	Low	36.4		
SIEMENS	F	Med	39.9	CAP	
SIEMENS	M	High	41.6	Control	4
SIEMENS	M	Med	41.9	Low	2
SIEMENS	M	High	51.6	Med	5
SIEMENS	M	High	55.6	High	9
SIEMENS	M	Cont	68.1		

Table 4 HD Atlas MRI Imaging parameter summary (the magnetic field strength for all scans is 3T). Mod: Modality, Manufact: Manufacturer, TR: Repetition Time, TE: Echo Time, TI: Inversion Time, Acq. Matrix: Acquisition Matrix.

Mod	Manufact	TR (ms)	TE (ms)	TI (ms)	Thickness (mm)	Acq. Matrix
T1	GE	6.524-7.816	2.796-3.004	450	1.0	0, 256, 256, 0
	Philips	7.313-7.7	3.271-3.501		0.63, 1.1	0, 256, 256, 0 0, 220, 218, 0
	SIEMENS	2300	2.67-2.98	900	0.7, 0.75, 1.1	0, 256, 256, 0 0, 320, 320, 0
T2	GE	3000-15000	39.6-100.128		1.1, 1.4, 1.8	0, 256, 256, 0 0, 288, 288, 0
	Philips	2500	181.29-185.97		1.1	220, 0, 0, 218
	SIEMENS	4800	354-433		0.7, 1.4	0, 256, 248, 0 0, 256, 250, 0 0, 256, 254, 0

Preprocessing

All repeated scans in one MRI session were processed together, i.e., repeated T1-w and T2-w MRI using the BRAINSTools suite (BRAINSia, 2015; Pierson et al., 2011). Experts rated the quality of each MRI to determine its suitability for further processing and then ordered the scans from high to low quality. MRI scans were then preprocessed using tools from the BRAINSTools suite. The preprocessing of MRIs consists of an AC–PC spatial alignment (Ghayoor et al., 2013; Lu, 2010), co-registration between T1-w and T2-w images, and multimodal bias-field correction (Kim and Johnson, 2013).

26

To create an initial set of label atlases for these 20 HD atlas candidates, label atlases were first automatically segmented with the ANTs joint fusion algorithm (Wang and Yushkevich, 2013; Wang et al., 2012). The joint fusion algorithm was applied on the HD MRI atlas candidates by using 20 T1-w MRI with whole brain label atlases already available from Neuromorphometrics Inc. Note that white matter sub-parcellation was further incorporated with the inclusion of FreeSurfer automatic segmentations. After undergoing ANTs joint fusion and sub-parcellation, the data were ready for the label atlas correction procedure.

LabelAtlasEditor

To visualize and efficiently edit label atlases, a user interface was developed within the infrastructure of the imaging software 3D Slicer. The open-source image processing toolkit SimpleITK (Lowekamp et al., 2013) was used for processing the label atlases. This section describes the custom user interfaces (widgets) that were developed for correcting these brain structures. The *Label Merge* widget allows a user to utilize a mask or a posterior probability map while merging the voxels of different labels in order to ensure that a voxel meets a user-defined minimum probability for a specific type, e.g., white matter or cerebrospinal fluid. The *Label Suggestion* widget provides a list of candidate labels for a questionable group of voxels based on the neighborhood information attained from intensity images. The *Automatic Dust Cleanup* widget automatically merges large amounts of small, disconnected regions to the most similar bordering label via the process employed in the *Label Suggestion* widget.

The 3D Slicer widgets Editor and Markups were included within this all-in-one module to expedite the manual cleaning process when required. Assuming that one label

27

represents a single biological structure or densely packed structures, small isolated regions should be examined for validity. The open-source software OpenAtlas can be used for providing an excellent three-dimensional (3D) visualization of the disconnected ROIs. This tool identifies errors that are difficult to visually recognize in two-dimensional slices, by placing a fiducial point on each disconnected region. Within the Markups widget, the user selects a target region from the fiducial points and then can view and easily modify the region by using Editor or one of the custom widgets developed.

Label Merge widget

The *Label Merge* widget is useful for reassigning incorrectly defined islands surrounding well-defined structures. It automatically corrects voxels from a source label to a target label at locations where the two are connected. Initially, a combined mask of the source and the target labels is created. Then, islands of connected voxels are identified. As shown in Figure 12 the user can choose to reassign the source voxels contained within the largest island of connected voxels or within all islands of connected voxels. A mask or a probability label map image for a specific type can be used to restrict the merging of voxels that do not meet the user-defined minimum value.

In Figure 12, "input_label" voxels from the source label "999" (unsegmented) are merged with the target label "24" (cerebrospinal fluid-CSF) if they are within the largest connected island and have a corresponding value in "csf_probility_map" that is larger than the user-defined minimum value of "0.10". The resulting label map is saved to "output_label".

28

Figure 12: Merge widget graphical user interface within LabelAtlasEditor. The input parameters include: (1) *Input Label Map Volume* **specifying the label image, (2)** *Target Label* **specifying the identification number of the target label to which new voxels are reassigned, (3)** *Suspicious Label* **specifying the identification number of the source label from which voxels are reassigned, (4) a check box to either reassign the source voxels contained within the largest island of connected voxels or within all islands of connected voxels, (5)** *Posterior Volume* **providing optional apriority information via a mask or a probability label map image for a specific type, (6)** *Minimum Threshold* **used for restricting the merging of voxels that do not meet this value in the** *Posterior Volume* **image, and (7)** *Output Label Map Volume* **specifying the output label image.**

Label Suggestion widget

The *Label Suggestion* widget suggests candidate labels for a questionable region. Even for experts, it is often difficult to visually assign a label for an ambiguous voxel such as the one illustrated in Figure 13. The *Label Suggestion* widget provides a quantitative measure for each neighboring label to remove ambiguity for small regions, such as those identified by OpenAtlas. As shown in Figure 14 the label candidate list is

29

ordered by a similarity criterion computed for each label on the target region's border. The similarity metric used in this study is a distance measurement of the mean intensity per region per modality given by

$$d = \sqrt{\sum_{For\ all\ image\ i} (T_i - B_i)^2} \quad ,$$

where T_i is the mean intensity value of the target region for image modality i and B_i is the mean intensity value of a border label region for image modality i. The border label with the smallest similarity metric has the closest average intensity to the target region. Figure 14 displays an example use-case in which a fiducial point was placed at the suspicious voxel (pink) displayed in Figure 13. The label "2129" (green) has been selected as the most similar adjacent region for the ambiguous region. The voxels within the ambiguous region are reassigned to the selected label in the *Output Label Map* via the *Relabel* button.
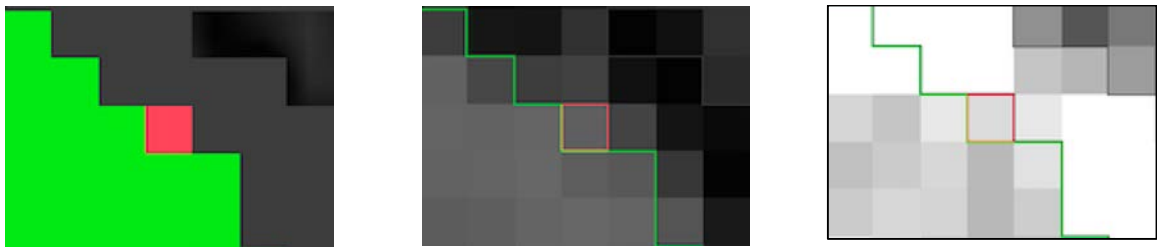
Figure 13: (Left) Example of an undefined pixel (shown in orange) at a border between two regions of interest illustrated by the green and dark gray labels. (Middle) Illustration of the T1-w pixel intensity values of the undefined pixel and the bordering labels. (Right) Illustration of the T2-w pixel intensity values of the undefined pixel and the bordering labels.

30

Figure 14: Label Suggest widget's graphical user interface within LabelAtlasEditor. The input parameters include: (1) *Input Intensity Volume 1* **specifying the T1-w image, (2)** *Input Intensity Volume 2* **specifying the T2-w image, (3)** *Input Label Map Volume***, and (4)** *Output Label Map Volume***.**

Automatic Dust Cleanup widget

The *Automatic Dust Cleanup* widget was developed to further expedite the correction process and reduce manual interaction. This widget utilizes an automated process that reassigns small, disconnected islands of voxels, called *dust*, by using the underlying process described in the *Label Suggestion* widget. As illustrated in Figure 15, the user may define a list of labels to include or exclude from the correction. The algorithm will review all labels if either list was omitted. It proceeds through the label list by correcting the label image one label at a time. Within each label *l*, the island size *s* is increased from 1 to *S*, the user-defined maximum island voxel count. All islands of size *s* are named sequentially from s_1 to s_N and are reviewed individually. In the

31

example illustrated in Figure 15, islands consisting of "5" or fewer voxels in the labels "999" and "15000" will be reassigned to the most similar adjacent label. Islands will be built using face connectivity and islands in densely packed groups will be reassigned given that the mask is not dilated.

Since it is possible for a label to consist of several densely packed islands of voxels, the proposed algorithm accounts for the proximity of a questionable island to the other bodies within that label. The goal is to correct islands in each label that are spatially isolated and therefore, more likely to be incorrectly identified. To identify isolated islands larger than one voxel, the binary mask of label l is dilated with a kernel radius calculated from the current island size. Dilation is not performed on islands containing only one voxel so that they remain isolated and are subsequently removed. The kernel radius, r, was set to the radius of a sphere with volume s, the current island voxel count. That is,

$$r = \left\lceil \sqrt[3]{\frac{s}{\frac{4}{3} * \pi}} \right\rceil .$$

Islands are built from the resulting binary mask for label l on the basis of the user-specified option for either four-neighbor (face) connectivity or eight-neighbor (face + edge + vertex full) connectivity. If the island size s is greater than one, the resulting image identifying islands is then masked by the original non-dilated mask for label l in order to create a label map based on the dilated label mask connections. Each island of size s is then merged with the neighboring label containing the most similar average pixel intensity values from the provided intensity images, as described in the *Label Suggestion*

32

widget. The user can choose to change the island label even if it is most similar to its own label. This cleaning process can also be performed on the command line with the same input parameters.

Figure 15: Automatic Dust Cleanup widget's graphical user interface within LabelAtlasEditor. The input parameters include: (1) *Input Intensity Volume 1* **providing the T1-w image, (2)** *Input Intensity Volume 2* **providing the T2-w image, (3) Input Label Map Volume, (4)** *Output Label Map Volume* **, (5) the** *Maximum Island Voxel Count* **specifying the largest island size to correct, (6)** *List of Labels to Review* **identifying labels to correct, (7)** *List of Labels to Exclude* **identifying labels to exclude from correction, (8) a check box to dilate or not dilate islands when determining connectivity, (9) a check box to build islands with full or face connectivity, and (10) a check box to reassign an island label even if it is most similar to its own label.**

This section demonstrates the application of LabelAtlasEditor and describes the process used to clean atlases in the PREDICT-HD dataset. The tools utilized in this study are available in the 3D Slicer Extensions Manager.

Stage 1: Addition of new regions using Editor

New regions of interest were identified and added to label maps by using the semi-automated *Editor* tool within 3D Slicer. The new regions of interest included the optic chiasm, Dura, and pineal gland. Obvious issues such as islands located outside of the brain region were then identified and corrected with *Editor*.

Stage 2: Automatic cleanup of unidentified dust using the Automatic Dust Cleanup widget

One of the goals of the atlas cleanup process is to reduce the number of unidentified voxels. To assign appropriate labels to the dust in the "unidentified" label, the Automatic Dust Cleanup widget was used to change the islands of six voxels or less to the most similar bordering label. It was noted that islands greater than six voxels often needed to be split into more than one label and were therefore excluded from this process. In order to break the unidentified voxels into smaller groups, the four-neighbor (instead of the eight-neighbor) connectivity option was used when building islands. To remove all unidentified dust particles, the option to force label reassignment was selected. This was performed without dilation in order to maximize the number of islands cleaned. Since ideally one would like to remove all islands of suspicious pixels, the mask was not

dilated. This allowed the widget to change small islands that were spatially close to other unidentified islands.

Stage 3: Semi-automated cleanup with the Merge widget and Editor tools

In this stage, larger unidentified islands were reassigned to the correct label with the *Editor* tools. Further, some label atlases contained excess venous blood beyond the brain region. To correct this, the *Merge* widget was used with a venous blood mask based on intensity ranges from both the T1-w and the T2-w images. This process efficiently reassigned those voxels to the background label.

Stage 4: Reassign isolated dust particles for most labels using the Automatic Dust Cleanup widget

The Automatic Dust Cleanup widget was used to change the islands of five voxels or less to the most similar bordering label. In this stage, the goal was to automatically reassign the isolated dust particles observed in many brain structures. Labels with viable isolated islands, such as the cerebrospinal fluid (CSF), were excluded from this stage. The options to build islands with eight-neighbor connectivity and to force label reassignment were selected. The process was performed with dilation to limit automatic label reassignment of viable island clusters.

Results

The proposed atlas generation approach successfully identified large numbers of undefined and isolated islands and the reassigned them to proper structures based on available prior information and morphometric operations. The dust removal process successfully decreased the plethora of small, isolated islands scattered throughout most

images. As illustrated in Figure 16 the overall brain structure definitions were maintained. The top left zoomed-in views in A and B, display the reassignment of unsegmented voxels (pink) to the *superior frontal* label (teal). The right zoomed-in views in A and B, display the reassignment of unsegmented voxels (pink) to the *brainstem* label (blue). The bottom left zoomed-in views in A and B, display the reassignment of unsegmented voxels (pink) to a new, manually identified label for the *optic chiasm* (yellow).

Figure 17 shows the progress of our approach for undefined regions. It can be seen for the three datasets that a range of unidentified voxels can occur at the initial stage. Many of the voxels occur on borders between two labels due to a partial volume effect. Figure 17 illustrates the removal of most unidentified voxels as the procedure progresses.

Figure 18 demonstrates the results of the automatic assignment of small, disconnected islands of voxels for three labels in the atlas. In the top image, it can be seen that many voxels along the border of the brainstem (green) are added to that label. These voxels were likely previously unidentified due to partial volume effects. The middle and bottom image illustrate the removal of several disconnected islands displayed in red. Those islands were automatically reassigned to the most similar neighbor label.

36
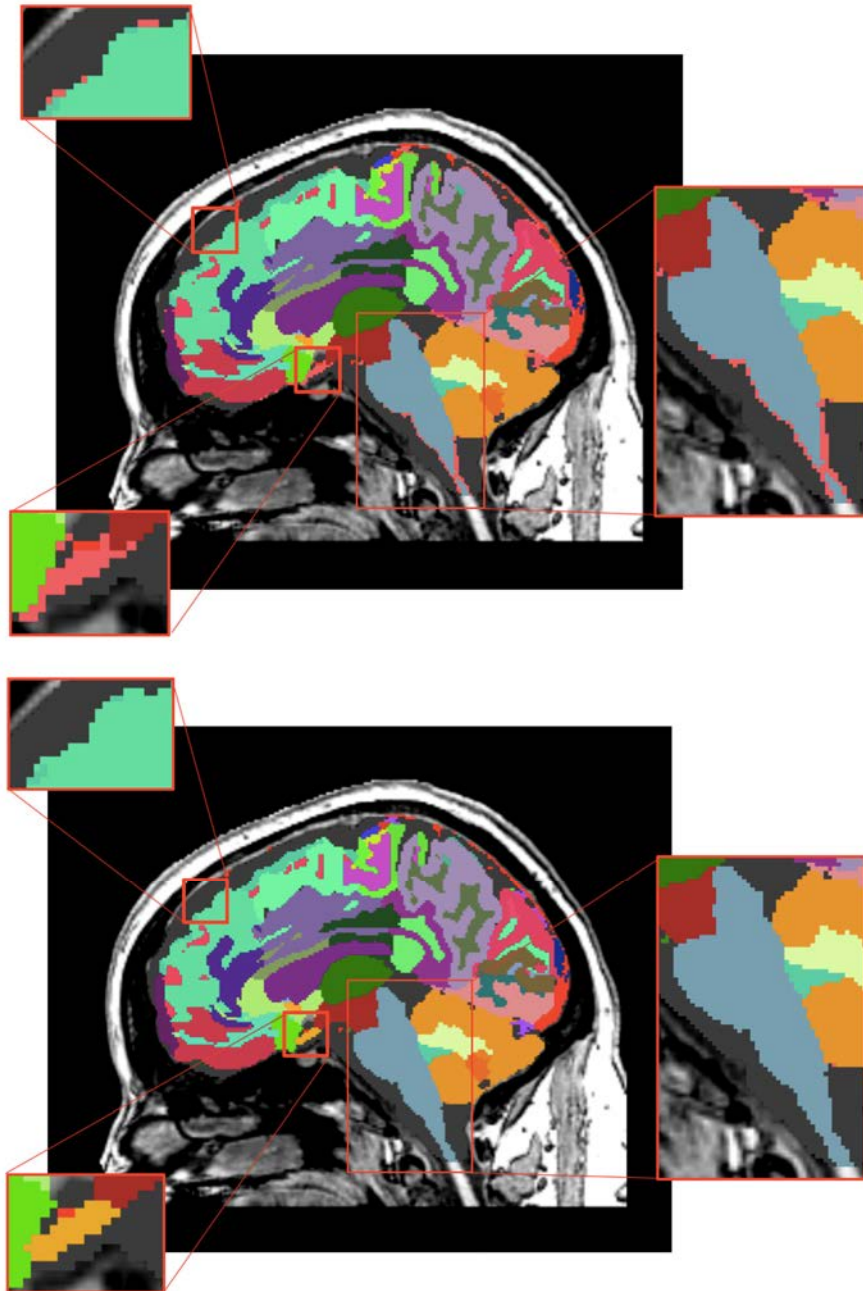
Figure 16: Before (A) and after (B) representation of the whole brain atlas and zoomed-in views of atlas corrections. (A) Thousands of undefined voxels occur at label borders (pink) because of partial volume effects. (B) The automatic dust removal process assigned previously undefined voxels and isolated islands.
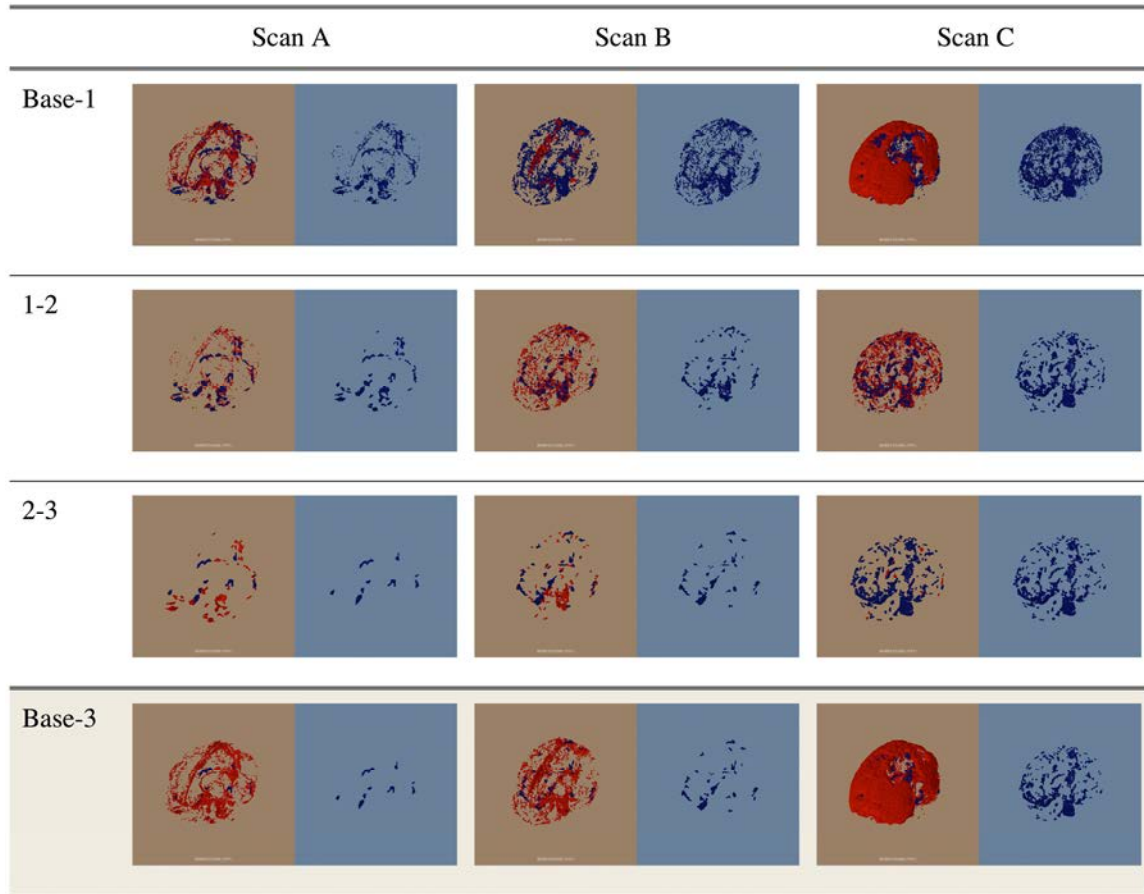
Figure 17: The change in the unidentified voxels through the first three correction stages (Base-1, 1-2, and 2-3) and the change from the base to Stage 3 (Base-3). Scans A, B, and C are examples of datasets containing low, average, and high amounts of unidentified voxels. For each image, the red voxels in the left half of the image (brown) indicate the voxels removed in this stage. The voxels shown in blue in the right half of the image (blue) are the remaining unidentified voxels at each stage.
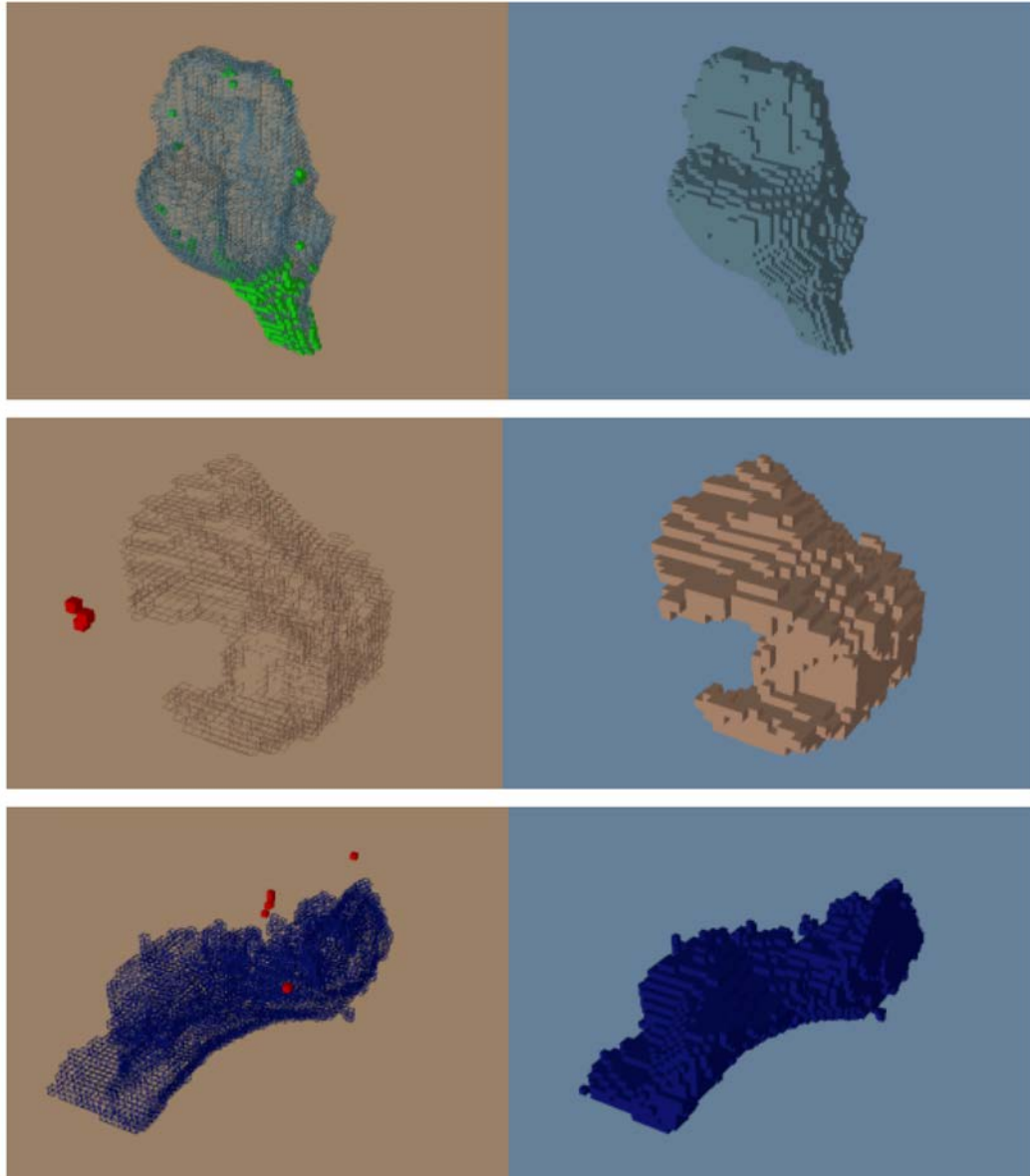
Figure 18: Before (left) and after (right) the automatic dust cleaning process. Voxels that are added (green) and removed (red) are color-coded using OpenAtlas. (Top) Example of voxels spatially adjacent to the *brainstem* (green) reassigned to the *brainstem* label. (Middle) Example of isolated voxels (red) removed from part of the *inferior frontal gyrus*. (Bottom) Example of isolated voxels (red) removed from the *lingual* label.

The average number of undefined voxels was reduced at each stage of the correction procedure as seen in Figure 19. The median counts of undefined voxels in the stages Base, 1, 2, and 3 were 16364, 10608, 5639, and 4305 respectively. There is one large outlier in the Base stage with 175,715 undefined voxels. This number was reduced to 13,498 after Stage 1, near the median count of 10,608 for label atlases in Stage 1. The majority of voxels for this outlier were merged with the background label because they were outside of the brain region. The average percent reduction in undefined voxels for each dataset from the initial atlas (Base) to the final cleaning stage (Stage 4) was 68.49% with a standard deviation of 14%. The maximum percent reduction in undefined voxels was 95.67% (this reduction occurred for the outlier) and the minimum percent reduction was 46.76%.
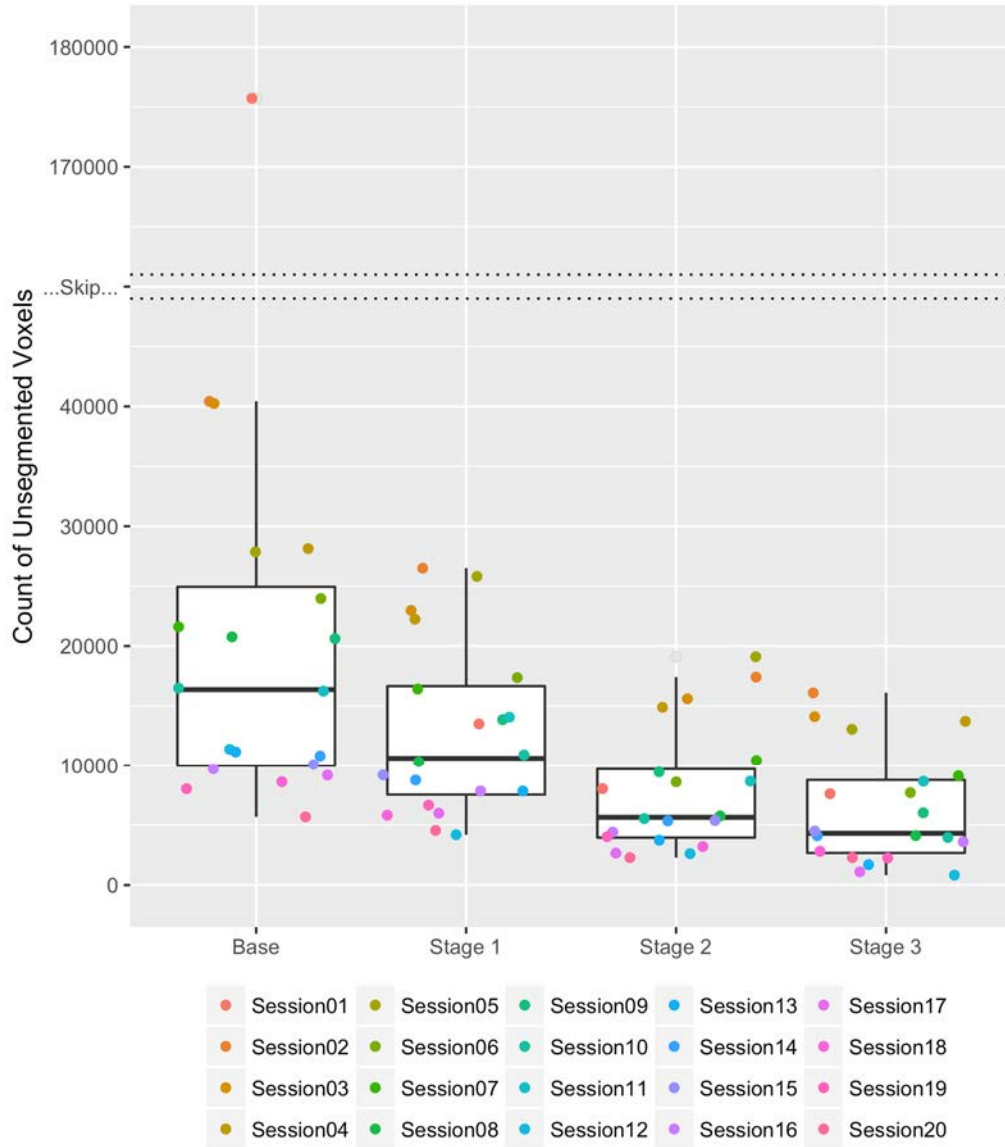
Figure 19: Boxplots of the count of undefined voxels remaining during each stage of correction for the 20 HD datasets. Each of the 20 datasets is represented by a consistently colored dot in each stage of correction.

41

Analysis of unidentified voxel reassignment (Base – Stage 1)

In the Stage 1 cleaning process, the Editor tool was used to manually and semi-automatically reassign unidentified voxels to new or existing labels. In addition, large segmentation failures were addressed to remove non-brain tissues (i.e. reassign voxels to the Background label) or other obvious failures. The volume in mm^3 of unidentified voxels reassigned to another label and the percentage increase of the label's volume are listed in Table 5. The number of unidentified voxels remaining at the end of Stage 1 ranged from 7.7% to 92.6%, and was 68.8% on average.

Four new labels (Dura, Nerve, Optic Chiasm, and the Pineal Gland) were manually added to the label atlases. First the Editor ChangeIslandEffect tool was used to assign unsegmented voxels connected in three-dimensions (to a user selected seed point) to the new label. Then an expert reviewed and cleaned each of the new label segmentations. On average 5502.9 mm^3, 654.3 mm^3, 461.0 mm^3, and 43.2 mm^3 were reassigned to these labels, respectively.

In addition, an average of 6108.8 mm^3 of the unidentified label's volume was reassigned to the Background label to exclude non-brain tissues incorrectly included due to segmentation failures. This resulted in an average 0.08% increase in the Background label's volume. The maximum volume of unidentified voxels reassigned to the Background label (122,017 mm^3) occurred for the outlier dataset containing a large number of unidentified voxels outside the brain region.

Analysis of unidentified voxel reassignment (Stage 1 – Stage 2)

In the Stage 2 cleaning process, the Automatic Dust Cleanup tool was used to automatically reassign unidentified small islands of voxels to the most similar adjacent

region. This method was only run on islands containing six voxels or less, and larger islands were left for manual editing. The unidentified voxels were reassigned throughout the brain and not limited to a particular region. Many of the voxel reassignments occurred at the border of two or more tissue types and most likely resulted from partial volume effects. The twenty datasets contained between 1,102 and 6,202 unsegmented islands with an average of 3,416 islands. The Automatic Dust Cleanup tool reassigned between 93.2% and 95.9% of these islands in each dataset. These small, reassigned islands accounted for an average of 41.0% of the unsegmented voxels and greatly reduced the number of unidentified islands requiring manual revision.

The volume in mm^3 of unidentified voxels reassigned to another label and the percentage increase of the label's volume are listed in Table 6. The percentage of unidentified voxels present at the end of Stage 1 that remained at the end of Stage 2 ranged from 44.4% to 74.0% and was 59.0% on average (7854.3 mm^3). The Gray Matter and the CSF labels were reassigned an average of 2671.9 mm^3 and 1114.5 mm^3 of the unidentified voxels (ranging from 880 mm^3 to 4592 mm^3 and from 110 mm^3 to 3626 mm^3, respectively). Venous Blood, Dura, and the Brainstem were reassigned the next highest averages of unidentified label volumes at 416.0 mm^3, 220.6 mm^3, and 152.7 mm^3, respectively. Other labels were reassigned less than 1.3% of the unidentified voxels each on average.

Analysis of unidentified voxel reassignment (Stage 2 – Stage 3)

In the Stage 3 cleaning process, the Merge and Editor tools were used by a reviewer to reassign larger islands of unidentified voxels to the correct regions. The volume in mm^3 of unidentified voxels reassigned to another label and the percentage

increase of the label's volume are listed in Table 7. On average, 23.7% of unidentified

voxels present at the end of Stage 2 were reassigned during this stage. The Background

label volume was increased by 432.8 mm^3 on average, which accounted for a 0.01%

average increase in volume. The Gray Matter, Brainstem, Nerve, Cerebrospinal Fluid,

and the Cerebellum Cortex were assigned average volumes from the unidentified label of

381.2 mm^3, 297.9 mm^3, and 107.4 mm^3, respectively. An additional 92.8 mm^3 in voxels

were manually added to the Nerve label, increasing the label's volume by an average of

22.7%. Other labels were reassigned on average less than 0.9% of the unidentified

voxels each.

Analysis of unidentified voxel reassignment (Base – Final Stage)

In the overall cleaning process, the Editor and Merge tools were used to manually

and semi-automatically reassign unidentified voxels to new or existing labels and the

Automatic Dust Cleanup widget was used to automatically reassign small islands. In

addition, large segmentation failures were addressed to remove non-brain tissues (i.e.

reassign voxels to the Background label) or other obvious failures. The volume in mm^3

of unidentified voxels reassigned to another label and the percentage increase of the

label's volume are listed in Table 8. The percentage of unidentified voxels remaining at

the end of the final cleaning stage ranged from 4.3% to 53.2% and was 31.5% on

average.

The Background label was reassigned the largest volume of the unidentified label

with an average of 6663.8 mm^3. This occurred due to the reassignment of unidentified

voxels located outside the brain region. The maximum volume reassignment (122,451

mm^3) of unidentified voxels reassigned to the background occurred for the outlier dataset

44

containing a large number of voxels outside the brain region. In addition, four new labels (Dura, Nerve, Optic Chiasm, and the Pineal Gland) were manually added to the label atlases. On average, these accounted for the reassignment of 5597.9 mm^3, 779.2 mm^3, 471.1 mm^3, and 49.7 mm^3 of the initial unidentified label's volume. Gray Matters labels and Cerebrospinal Fluid (CSF) received 3051.2 mm^3 and 1486.1 mm^3 of the initial unidentified label's volume.

Table 5: Stage 1 (1st Manual cleaning): Volumetric distribution in mm³ of previously unidentified (Suspicious) voxels and the percent increase in volume for each label. The average, standard deviation, minimum, and maximum for both measures are reported. Note that the "Suspicious" row represents remaining unidentified voxels and the Dura, Nerve, Optic Chiasm, and Pineal Gland are new labels.

Label	Avg. Vol. (mm³)	St. Dev. Vol. (mm³)	Min. Vol. (mm³)	Max. Vol. (mm³)	Avg. (%)	St. Dev. (%)	Min. (%)	Max. (%)
Suspicious	12735.0	7073.4	4188	26495	68.78	19.83	7.68	92.64
Background	6108.8	27282.0	0	122017	0.08	0.38	0.00	1.69
Dura	5502.9	8679.7	0	37901	100.00	0.00	100.00	100.00
Nerve	654.3	480.3	244	2396	100.00	0.00	100.00	100.00
Optic Chiasm	461.0	313.3	147	1658	100.00	0.00	100.00	100.00
CSF	271.5	1196.9	0	5356	0.19	0.86	0.00	3.83
Brainstem	71.3	184.0	0	684	0.44	1.18	0.00	4.66
Pineal Gland	43.2	32.5	0	124	100.00	0.00	100.00	100.00
Gray Matter	2.7	8.4	0	31	0.00	0.00	0.00	0.01

Table 6: Stage 2 (Automated run on unidentified): Volumetric distribution in mm^3 of previously unidentified (Suspicious) voxels and the percent increase in volume for each label. The average, standard deviation, minimum, and maximum for both measures are reported. Note that the "Suspicious" row represents remaining unidentified voxels.

Label	Avg. Vol. (mm^3)	St. Dev. Vol. (mm^3)	Min. Vol. (mm^3)	Max. Vol. (mm^3)	Avg. (%)	St. Dev. (%)	Min. (%)	Max. (%)
Suspicious	7854.3	5185.9	2282	19111	58.97	7.79	44.40	74.04
Gray Matter	2671.9	1030.3	880	4592	0.48	0.17	0.15	0.81
CSF	1114.5	814.9	110	3626	0.76	0.42	0.08	1.51
Venous Blood	416.0	461.8	24	1984	1.64	1.68	0.09	6.13
Dura	220.6	176.4	0	771	12.42	21.13	0.66	100.00
Brainstem	152.7	64.0	71	299	0.83	0.40	0.37	1.82
Cerebellum Cortex	134.4	64.9	50	316	0.13	0.06	0.04	0.30
Background	45.9	61.3	0	221	0.00	0.00	0.00	0.00
Nerve	32.0	38.5	3	147	4.46	3.63	0.71	12.55
Inferior Frontal Gyrus	27.1	18.9	2	65	0.61	0.44	0.04	1.35
Cerebellar Vermal Lobules	19.2	17.6	1	88	0.19	0.17	0.01	0.84
Right Subcortical Gray Matter	10.2	9.2	0	39	0.08	0.07	0.00	0.33
Optic Chiasm	9.6	7.7	1	28	1.70	1.78	0.12	7.91
Pineal Gland	6.5	9.7	0	30	25.78	35.92	0.00	100.00
Ventral DC	5.8	5.8	0	17	0.06	0.07	0.00	0.20
Cerebellum White Matter	5.3	5.1	0	19	0.02	0.02	0.00	0.07
Left Subcortical Gray Matter	5.2	5.6	1	20	0.04	0.05	0.01	0.19
Right Thalamus Proper	2.5	4.4	0	18	0.03	0.06	0.00	0.26
Left Thalamus Proper	1.4	2.0	0	7	0.02	0.03	0.00	0.09
White Matter	0.7	1.1	0	4	0.00	0.00	0.00	0.00

Table 7: Stage 3 (2nd Manual cleaning): Volumetric distribution in mm^3 of previously unidentified (Suspicious) voxels and the percent increase in volume for each label. The average, standard deviation, minimum, and maximum for both measures are reported. Note that the "Suspicious" row represents remaining unidentified voxels.

Label	Avg. Vol. (mm³)	St. Dev. Vol. (mm³)	Min. Vol. (mm³)	Max. Vol. (mm³)	Avg. (%)	St. Dev. (%)	Min. (%)	Max. (%)
Suspicious	6363.7	4718.3	828	16095	76.29	19.88	31.71	100.00
Background	432.8	424.1	0	1602	0.01	0.01	0.00	0.03
Gray Matter	381.2	1154.1	0	5231	0.06	0.18	0.00	0.80
Brainstem	297.9	369.9	0	1250	1.63	2.02	0.00	6.49
CSF	107.4	149.4	0	446	0.08	0.11	0.00	0.34
Nerve	92.8	179.7	0	788	22.74	53.45	0.00	241.72
Cerebellum Cortex	72.9	189.6	0	835	0.07	0.18	0.00	0.74
Dura	58.7	164.4	0	728	6.03	22.18	0.00	100.00
Venous Blood	42.2	73.6	0	239	0.19	0.36	0.00	1.37
Inferior Frontal Gyrus	1.7	6.1	0	27	0.05	0.18	0.00	0.82
Cerebellar Vermal Lobules	0.9	2.8	0	11	0.01	0.03	0.00	0.14
Cerebellum White Matter	0.8	3.1	0	14	0.00	0.01	0.00	0.06
Optic Chiasm	0.5	2.2	0	10	0.10	0.43	0.00	1.93
White Matter	0.5	2.2	0	10	0.00	0.00	0.00	0.00
Ventral DC	0.4	1.8	0	8	0.00	0.02	0.00	0.08

Table 8: Final Cumulative change: Volumetric distribution in mm³ of previously unidentified (Suspicious) voxels and the percent increase in volume for each label after all cleaning stages. The average, standard deviation, minimum, and maximum for both measures are reported. Note that the "Suspicious" row represents remaining unidentified voxels.

Label	Avg. Vol. (mm³)	St. Dev. Vol. (mm³)	Min. Vol. (mm³)	Max. Vol. (mm³)	Avg. (%)	St. Dev. (%)	Min. (%)	Max. (%)
Background	6663.8	27257.5	1	122451	0.09	0.38	0.00	1.70
Suspicious	6361.7	4719.4	828	16094	31.51	13.97	4.33	53.24
Dura	5597.9	8591.7	0	38156	100.00	0.00	100.00	100.00
Gray Matter	3051.2	1857.3	1227	9809	0.54	0.29	0.21	1.52
CSF	1486.1	1216.5	180	5369	1.04	0.79	0.22	3.84
Nerve	779.2	500.6	362	2543	100.00	0.00	100.00	100.00
Venous Blood	563.0	875.5	24	4036	2.79	6.25	0.12	28.95
Brainstem	522.3	389.6	76	1498	2.92	2.26	0.42	7.36
Optic Chiasm	471.1	312.8	151	1669	100.00	0.00	100.00	100.00
Cerebellum Cortex	206.1	177.2	84	885	0.20	0.16	0.09	0.79
Pineal Gland	49.7	37.9	0	134	100.00	0.00	100.00	100.00
Inferior Frontal Gyrus	28.8	20.4	2	66	0.66	0.52	0.04	2.02
Cerebellar Vermal Lobules	20.1	17.6	1	88	0.20	0.17	0.01	0.84
Right Subcortical Gray Matter	10.2	9.2	0	39	0.08	0.07	0.00	0.33
Ventral DC	6.2	5.9	0	17	0.07	0.07	0.00	0.20
Cerebellum White Matter	6.1	5.5	0	19	0.02	0.02	0.00	0.07
Left Subcortical Gray Matter	5.2	5.6	1	20	0.04	0.05	0.01	0.19
Right Thalamus Proper	2.5	4.4	0	18	0.03	0.06	0.00	0.26
Left Thalamus Proper	1.4	2.0	0	7	0.02	0.03	0.00	0.09
White Matter	1.2	2.4	0	10	0.00	0.00	0.00	0.00

Atlas Correction and Processing Time

The semi-automatic reassignment of unidentified voxels to new labels and the correction of atlas issues in Stage 1 took between 30 minutes and 1.5 hours depending on the quality of the initial atlas segmentations. On a quad-core Intel Xeon machine, the automatic cleaning process for the undefined voxels in Stage 2 took an average of 20.2 minutes to complete with a standard deviation of 7.7 minutes. Then, the semi-automatic reassignment of larger unidentified islands to the correct label and further correction of atlas inconsistencies took between 30 minutes and 1.5 hours depending on the quality of the atlas segmentation. Finally, the automatic cleaning process for most labels in Stage 4 took an average of 12.2 minutes to complete with a standard deviation of 1.3 minutes.

The total time for the cleaning process (including all stages) ranged from 1.5 to 4 hours per dataset. Considering there are thousands of islands in a dataset and manual reassignment of each island takes approximately 45 seconds, the automatic cleaning process substantially improved editing time. The manual and semi-automatic revision stages are optional, but recommended if the datasets will be used as template atlases. No effort was made to optimize computational time for the Python implementation of the automatic cleaning tool.

Discussion

This work described the creation of a multi-atlas dataset for the entire brain region that spans a range of HD progression and MRI scanners. These atlases are expected to improve the future atlas-based MRI analysis on HD. The tool was demonstrated to decrease the editing time for the 20 considered atlases by providing automatic and semi-automatic reassignment of islands. After the correction of atlas inconsistencies and

50

small, disconnected regions, the number of unidentified voxels for each dataset was reduced on average by 68.48%. Additionally, new regions of interest were efficiently identified and included in the atlases. The Background label was reassigned the most unidentified voxel volume on average at 6663.8 mm^3. This occurred due to the reassignment of non-brain tissue previously assigned to the unidentified label due to segmentation failures.

The *Automatic Dust Clean Up* widget was designed for use in correcting small, disconnected islands and may not be accurate when applied to larger islands. As the islands increase in size, the probability of the voxels belonging to more than one label increases and it is more appropriate to examine the islands voxel by voxel. To reduce the number of unidentified islands requiring manual revision, this method was only run on islands containing six voxels or less. Larger islands were left for manual editing. Many of the voxel reassignments occurred at the border of two or more tissue types and most likely resulted from partial volume effects. The twenty datasets contained between 1,102 and 6,202 unsegmented islands with an average of 3,416 islands. The Automatic Dust Cleanup tool reassigned between 93.2% and 95.9% of these islands in each dataset. These small, reassigned islands accounted for an average of 41.0% of the unsegmented voxels remaining after the first manual cleaning process.

The LabelAtlasEditor offers an efficient, user-friendly method for editing label segmentations. The relative ease of atlas manipulation simplifies the correction of atlases and the tool is flexible enough to be used for anatomical regions other than the brain. LabelAtlasEditor has broad potential applications and is available for free download in the Extension Manager and for use as part of 3D Slicer.

51

CHAPTER 4: ULTRA-LOW RADIATION CORONARY CALCIUM SCANNING WITH AN ENERGY-MODIFIED AGATSTON SCORE

Introduction

Cardiovascular disease (CVD) is the leading cause of death globally according to the World Health Organization (WHO, 2015). In the United States, 640 thousand people die of heart disease every year, representing one in every four deaths (CDC, 2015). Of the men who die suddenly of coronary heart disease, half of them have no previous symptoms (CDC, 2015). It is therefore important to identify asymptomatic patients at risk of coronary events.

There are several methods used to identify patients with a high risk of developing CVD. One method uses the Framingham risk score (FRS) which estimates the risk of a cardiac event over a 10-year period (NIH, 2016). The risk scores are divided into three levels: low, intermediate, and high. These levels correspond to the percent likelihood of a cardiac event and are defined as less than 10%, 10% to 20%, and greater than 20%, respectively. Since this is a population-based model, it does not account for an individual's actual burden of atherosclerotic disease (Budoff and Gul, 2008).

The presence of coronary calcium is an indicator of atherosclerosis and its level corresponds to the degree of atherosclerotic disease (Budoff and Gul, 2008). Coronary calcium can be measured via cardiac computerized tomography (CT) scans. Coronary artery calcium (CAC) screening may detect individuals who have coronary atherosclerosis and would benefit from more aggressive therapy (Budoff and Gul, 2008). Studies show that individuals categorized by the FRS in the intermediate risk level may benefit the most by further risk stratification via CAC screening (Budoff and Gul, 2008).

52

The Cardiac Agatston Score (CAS) is a measure used to quantify CAC. The goal of this project is to allow the use of ultra low radiation dose CT for measuring CAS.

Prior to this work, it was hypothesized that the CAS measured with a low energy protocol (80 kVp) would be comparable to that obtained by the standard protocol (120 kVp). The aim was to develop and validate the low dose, low energy protocol for CAC screening using an Energy-Modified Agatston score. In this work, the design, development, and verification of a new semi-automated tool to calculate the CAS for both 80 kVp and 120 kVp images is described.

Methods

This section starts by describing the method used to calculate the CAS on standard 120 kVp scans and the co-registration performed in order to calculate the CAS scores on 80 kVp scans. It then illustrates the Cardiac Agatston Scoring application and the proposed procedure for measuring a set of CT images. It is followed by a description of the study dataset. Finally, two statistical analyses are presented, the first comparing the 120 kVp CAS measurements from the 3D Slicer module and a Siemens application, and the second comparing the 3D Slicer module measurements of the 80 kVp and 120 kVp scans.

Agatston Score Calculation

The pixels in the CT images are set to the Hounsfield unit (HU) scale where the radiodensity of distilled water is 0 HU and the radiodensity of air is -1000 HU. Calcium-rich structures, such as calcium plaques or bone, have an HU value greater than or equal to 130 in the standard 120 kVp scans. CT imaging can be performed at various energy

(kVp) or current (mAs) levels. Energy levels range from 80-140 kVp and changes in the energy level affect the HU density of the image. Previous Agatston scoring has been performed at 120 kVp.

To calculate the Agatston score, spots of coronary calcium are selected and each spot is scored based on its area and the maximum density of the pixels, known as the Peak CT. A value called the *x-factor* is used to categorize the intensity levels of each spot based on the Peak CT. An *x-factor* for each spot is set to the value of 1, 2, 3, or 4 based on the ranges shown in Table 9. The area of the spot is then multiplied by the *x-factor* to calculate the spot's score. The sum of all spot scores is the Agatston score or "calcium score".

Table 9: X-factor categories for the standard
CAS method at 120 KEV.

Categories for X-factor	Lower at 120 kV (Standard)	Upper at 120 kV (Standard)
1	130	199
2	200	299
3	300	399
4	400	

Figure 20 provides an example score calculated for two calcium spots on separate axial slices. The first spot (left) has an area of 15 mm^2 and a Peak CT of 450. Referencing the ranges in Table 9, it can be seen that this spot's *x-factor* is 4. Therefore, the score for the first spot is 60 (4 x 15). Similarly, the area for the second spot (right) is 8 mm^2 and this is multiplied by the *x-factor* 2 because the Peak CT is 290. The resulting score for the second spot is 16 (2 x 8).

54
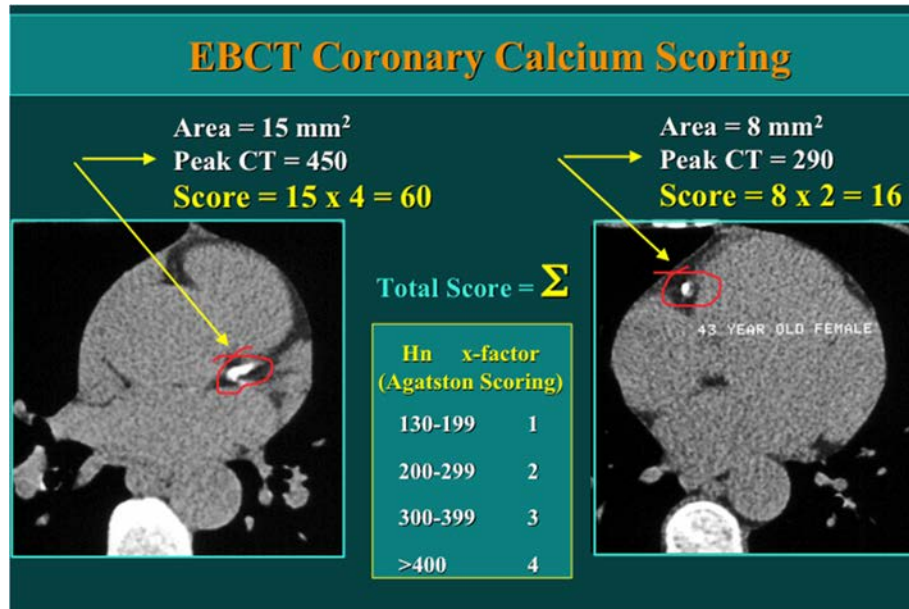
Figure 20: Example Coronary Calcium Scoring of two calcium spots in separate axial slices.

As the score increases, the risk of coronary heart disease also increases. Risk is divided into four levels based on the CAS: (1) CAS < 10 means that little to no calcium was found, (2) 10 <= CAS < 100 indicates a moderate amount of calcification, (3) 100 <= CAS < 400 signifies an increased level of calcification, and (4) CAS > 400 implies extensive calcification. A CAS over 1,000 is associated with a 20% chance of suffering a myocardial infarction or cardiac death within a year (Hoffmann et al., 2003).

Voxel to Voxel 3D Co-registration

Lowering X-ray energy results in a substantial reduction in radiation exposure, but also alters the measured HU of the CT image. Lower energy generally leads to higher scoring, and so the scoring categories need to be modified when using lower energy. To calculate *x-factor* levels for the 80 kVp CT scans, a 3D co-registration was

performed on a pair of scans of the same cadaver heart, one at 120 kVp and the other at 80 kVp. The voxel to voxel based correlation graph is shown in Figure 21.
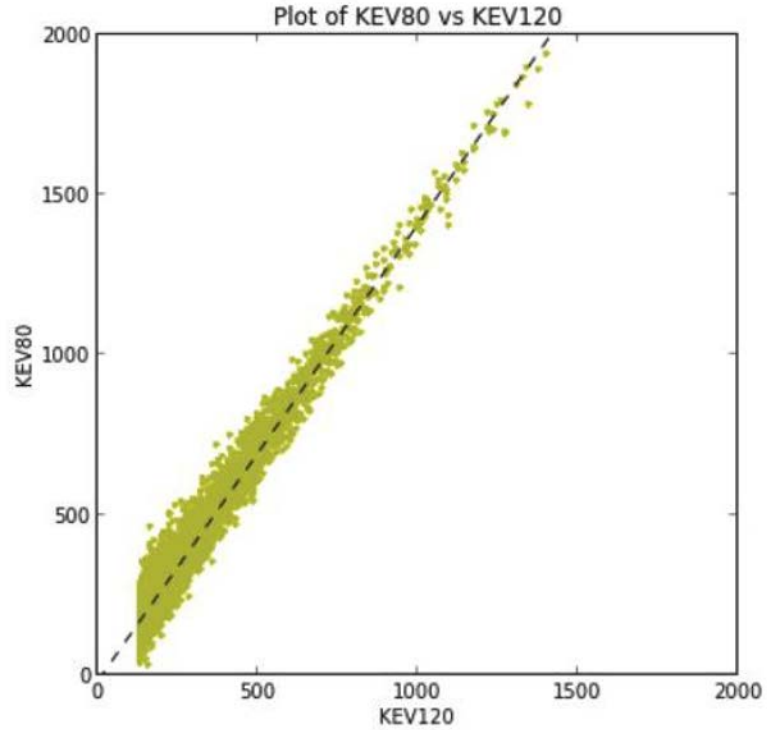
Figure 21: Voxel to voxel based correlation graph for a 3D co-registration of a cadaver heart image at both 80 and 120 kVp.

The linear relationship between the 120 kVp and 80 kVp HU obtained from the 3D co-registration is represented by

$$HU_{80} = 1.4256 * HU_{120} - 18.355$$

where HU_{80} is the projected HU for a voxel at 80 kVp based on HU_{120}, the HU of the same voxel at 120 kVp. Using this equation, the new category limits for the *x-factor* were calculated as listed in Table 10.

Table 10: X-factor categories for the standard and new methods at 120 and 80 kVp, respectively.

Categories for X-factor	Lower at 120 kV (Standard)	Upper at 120 kV (Standard)	New lower limit at 80 kV	New upper limit at 80 kV
1	130	199	167	265
2	200	299	266	407
3	300	399	408	550
4	400		551	

Results

Cardiac Agatston Scoring 3D Slicer Module

The Cardiac Agatston Scoring application, illustrated in Figure 22, provides the user with an efficient, semi-automated method to calculate the CAS. The interface consists of several sections: (1) the Help and Acknowledgements section providing usage instructions, (2) the Reload and Test section, (3) the *Input Parameters* section for the required input cardiac CT scan, and (4) the *Imaging Viewing* area.
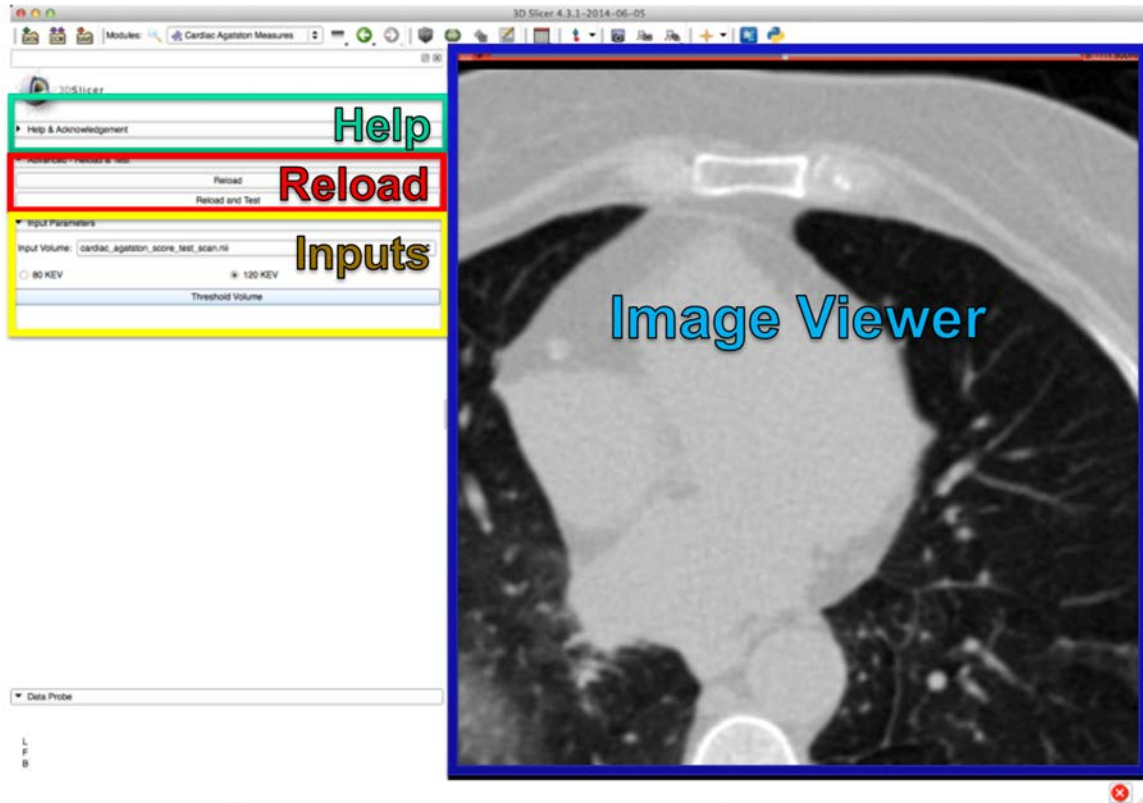
Figure 22: Display of the Cardiac Agatston Scoring application before a threshold has been applied to the Input Image and the Editor module has been populated to the panel.

First the user must set a cardiac CT scan to the *Input Volume* and select the radial button to use the *x-factor* levels for either 80 kVp or 120 kVp. Once those values are set, the *Threshold Volume* button can be selected. This button creates a binary label image via the SimpleITK Threshold filter which sets voxels less than the threshold value to 0 and voxels greater than or equal to the threshold to 1. The threshold is set to the minimum HU value in the lowest *x-factor* level: 130 HU for 120 kVp and 167 HU for 80 kVp.

After the label image is created, a custom lookup table containing seven labels is set for the label image. Label 0 identifies background pixels, label 1 signifies the *Default*

58

pixels with HU values greater than the threshold level, labels 2-5 identify the four

arteries, and label 6 is used to display the Total Agatston score. Table 11 lists the red

green blue alpha (RGBA) color specifications for the labels. The label image is then

pushed to 3D Slicer using the *sitkUtils* package (Kitware, 2015).

Table 11: Custom Lookup Table.

Label	Color		Name	Red	Green	Blue	Alpha
0	Clear		Background	0	0	0	0
1	Green		Default	95	212	45	255
2	Purple		Left_Main_(LM)	226	57	241	255
3	Yellow		Left_Arterial_Descending_(LAD)	248	242	60	255
4	Blue		Left_Circumflex_(LCX)	111	184	210	255
5	Red		Right_Coronary_Artery_(RCA)	216	23	49	255
6	Blue		Total	14	24	255	255

Custom Editor Widget

After the *Threshold Volume* button is selected, a local version of the *Editor*

widget, seen in Figure 23, is populated to the module to provide a suite of label editing

tools. To activate these tools, the two Editor inputs *Master Volume* and *Merge Volume*

are automatically set to the input cardiac CT scan and the thresholded label image,

respectively. The local Editor object has been modified to display only a few Editor tools

including: *Undo* to revert the most recent change, *Redo* to re-implement the most recent

change made by *Undo*, *Default* to deselect the current *Editor* tool, and *Paint* to change

voxels within the current 2D plane.
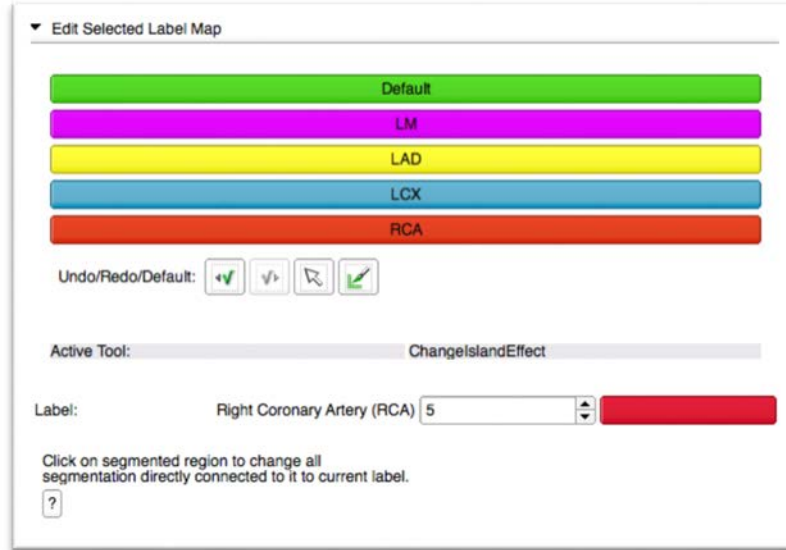
Figure 23: Display of the custom Editor widget with five new buttons to automatically set the Change Island Effect tool.

To minimize manual human effort in calcium plaque segmentation, five colorful custom buttons were added to the Editor widget as displayed in Figure 23. Selection of one of these buttons automatically assigns the *Change Island Effect* Editor tool to either the *Default* label or one of the four arteries. Then with the selection of a single seed point, all *Label Image* voxels connected to (in three-dimensions) the seed point and of the same label as the seed point are changed to an artery label as seen in Figure 24. This provides an efficient method to assign voxels quickly to one of the four arteries in three-dimensions. The island color is changed to the value specified in the *Lookup Table*, seen in Table 11. Shortcut keys for each of the five buttons were added to the list of *Editor* shortcut keys to make label identification more efficient. The shortcut keys are the integer keys 1, 2, 3, 4, and 5 for *Default, LM, LAD, LCX,* and *RCA*, respectively.

60
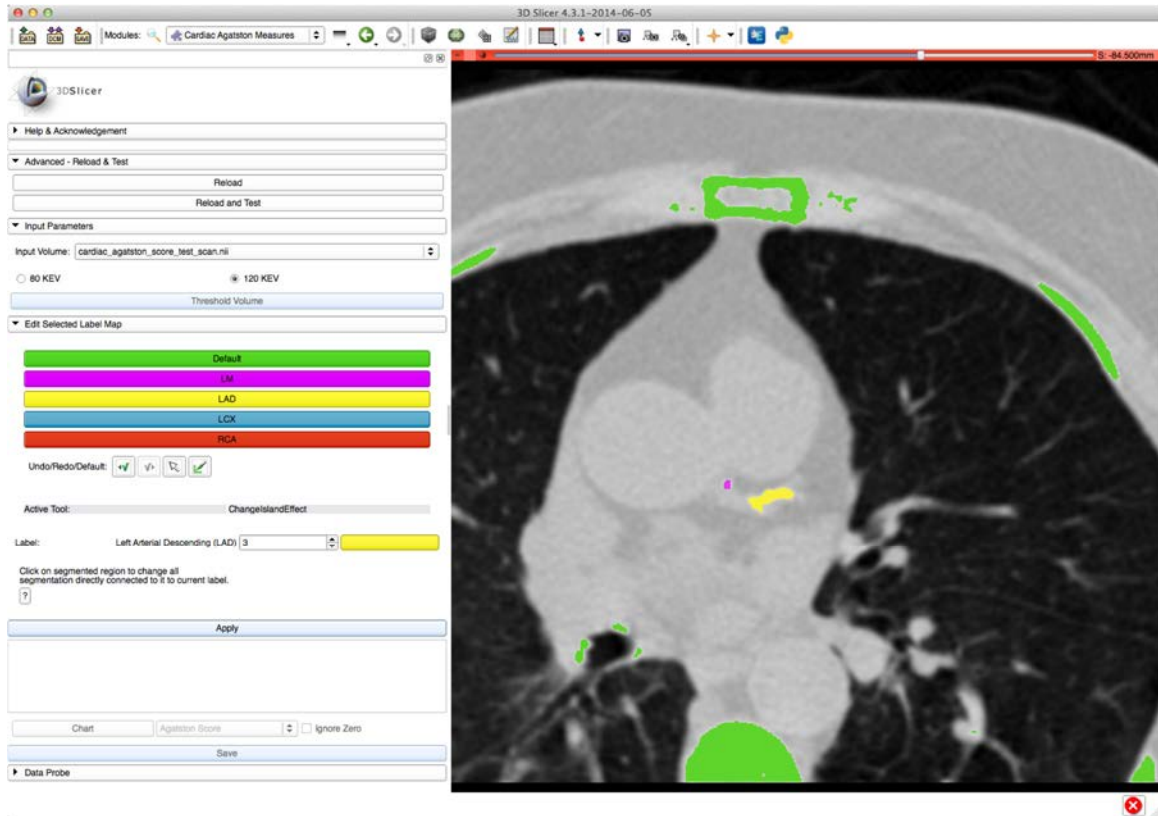
Figure 24: Display of the Cardiac Agatston Scoring application after a threshold has been applied to the input image and two calcium plaques have been identified: a small plaque in the LM (pink) and a large plaque in the LAD (yellow).

Cardiac Agatston Scoring Calculation

After the calcium plaques have been identified and assigned to arteries, selection of the *Apply* button calculates the CAS scores for each artery individually and also gives the total score. The button's text will change from "Apply" to "Working" to indicate that the algorithm is running. To calculate and display the CAS results, a custom *Label Statistics* object was created based on the built-in 3D Slicer widget (Pieper et al., 2014a). To process the input and label images with SimpleITK filters, these images were pulled from 3D Slicer and stored as SimpleITK image objects using the *sitkUtils* package.

61

A binary mask for each artery label was created using the *Binary Threshold* filter to isolate the pixels that identify that artery. Then a connected components image containing a new label identifier for each island is produced for each binary mask using the *Connected Components* and *Relabel Components* image filters.

Since the CAS for a calcium plaque is calculated in the axial plane, 2D images for each axial slice are created for both the input image and the connected components image. Then the Label Statistics image filter is used to return the pixel count and maximum HU value for each island in the axial plane. The area of each island is calculated by multiplying the pixel count by the product of the image spacing in the "x" and "y" coordinates. The maximum HU value for each island is used to determine the *x-factor* by using the ranges defined in Table 10. The score for each island is then simply the product of the area and the *x-factor*. These scores are used to calculate the CAS score for each artery and the total Agatston score.

Displaying Agatston Scores

The *Label Statistics* widget's table was adjusted to display the Agatston score and voxel count for each artery and all arteries combined. As seen in Figure 25, the label color, index, name, Agatston score, voxel count, volume in mm^3, volume in cc, minimum HU value, maximum HU value, mean HU value, and standard deviation of the HU values in each label are displayed. Also shown in Figure 25, is the option to create a chart based on the tabled statistics. Figure 26 illustrates the module after the CAS values are calculated and presented in a chart.

	Index	Label Name	Agatston Score	Count	Volume mm^3	Volume cc	Min	Max	Mean	StdDev
■	2	Left Main (LM)	4.5229	26	6.78436	0.00678436	137	276	201.538	44.7251
■	3	Left Arterial Descending (LAD)	88.3706	254	66.2779	0.0662779	131	654	290.76	121.42
■	5	Right Coronary Artery (RCA)	24.2671	104	27.1374	0.0271374	130	384	208.077	61.6902
■	6	Total	117.161	384	100.2	0.1002	130	654	262.326	111.709

Figure 25: Tabular display of the Agatston scores and label statistics for each artery and the combined arteries.
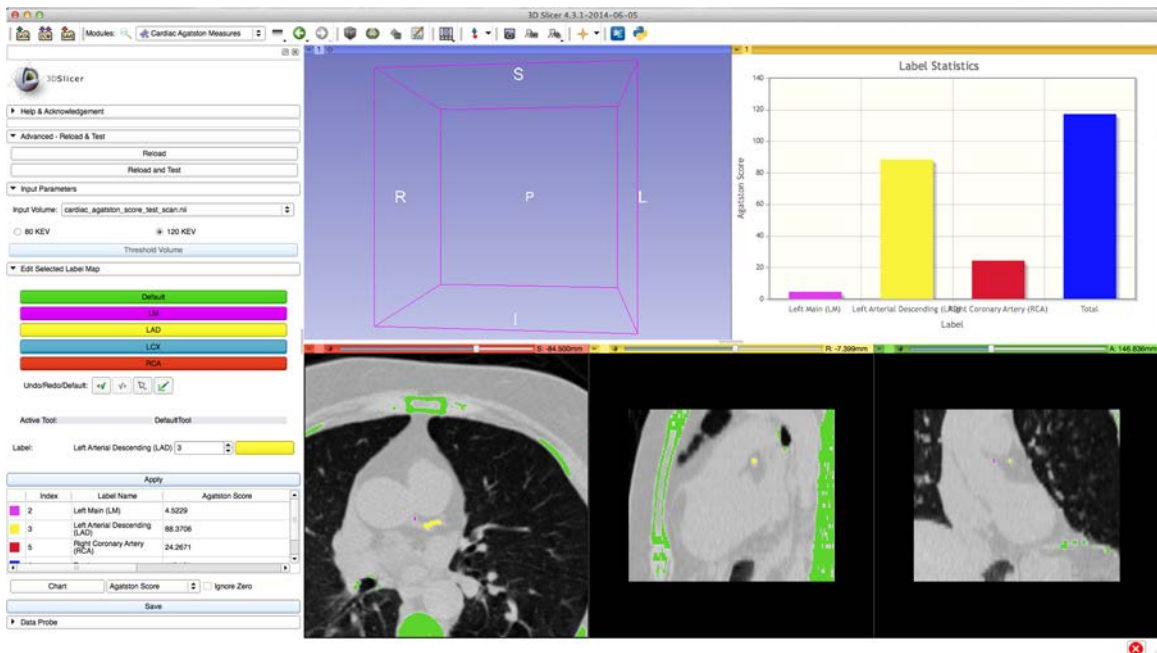
Figure 26: Display of Cardiac Agatston Scoring module with Agatston scores displayed in a table and chart for each artery and the combined arteries.

The Save button saves the information in the table to a comma separated formatted file using a function within the local *Label Statistics Logic* object. This file can be reviewed using a text editor or with the commonly used Excel application. This also saves the MRML scene, label and input images, and charts to a user specified folder with the 3D Slicer application logic. If the saved MRML scene is later opened with 3D Slicer,

the volumes are automatically loaded and positioned in the same view as when the

MRML scene was saved. This allows for quick and easy review of previous

measurements.

Slicer Extension Manager

The Cardiac Agatston Scoring module was submitted through the 3D Slicer

Extension Manager. As seen in Figure 27, the module can be installed and managed with

a single click. When the 3D Slicer application is restarted, the module is ready for use.
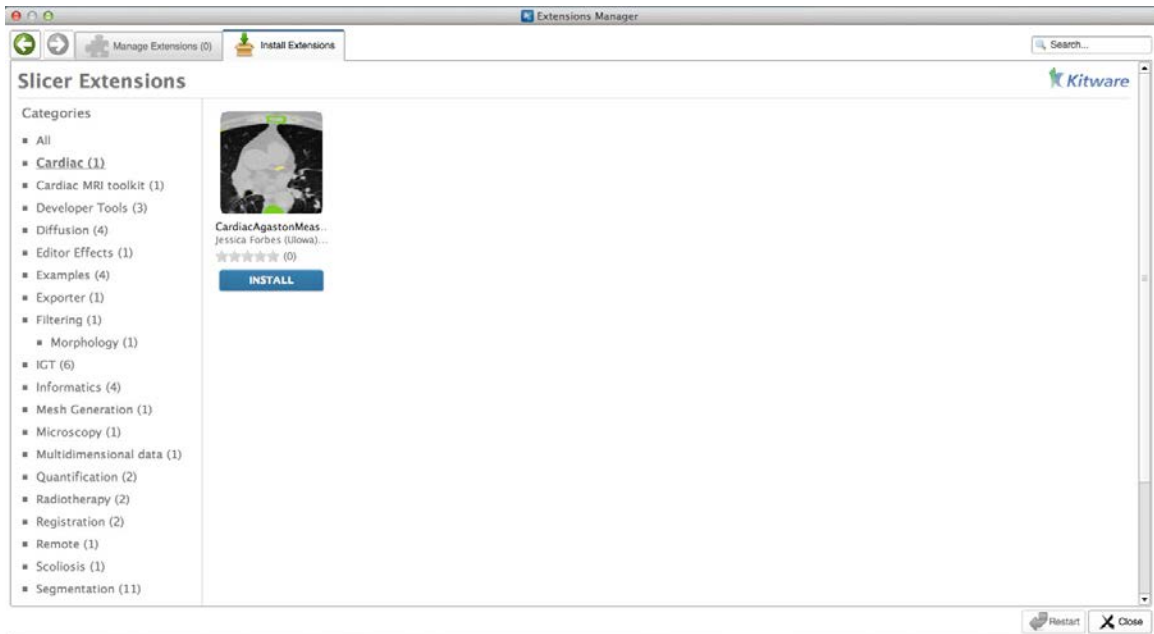
Figure 27: Display of the 3D Slicer Extension Manager used to install the Cardiac Agatston Scoring module.

Statistical Analysis

Cardiac CT scans at both 120 kVp and 80 kVp were obtained for a group of 16

subjects with written informed consent. The total CAS was calculated for the 120 kVp

64

scans with both the Siemens and 3D Slicer tools, and for the 80 kVp scans using the

modified Agatston scoring method in the 3D Slicer application.

Outlier Removal

It can be seen in Figure 28 that there is an extreme outlier in the dataset where the

80 kVp (CAS = 3070) was more than twice the value measured using 3D Slicer for the

120 kVp scan (CAS = 1305). The QQ plot in Figure 29 shows the outlier (top right) in

the comparison of 120 and 80 kVp scans measured using 3D Slicer.
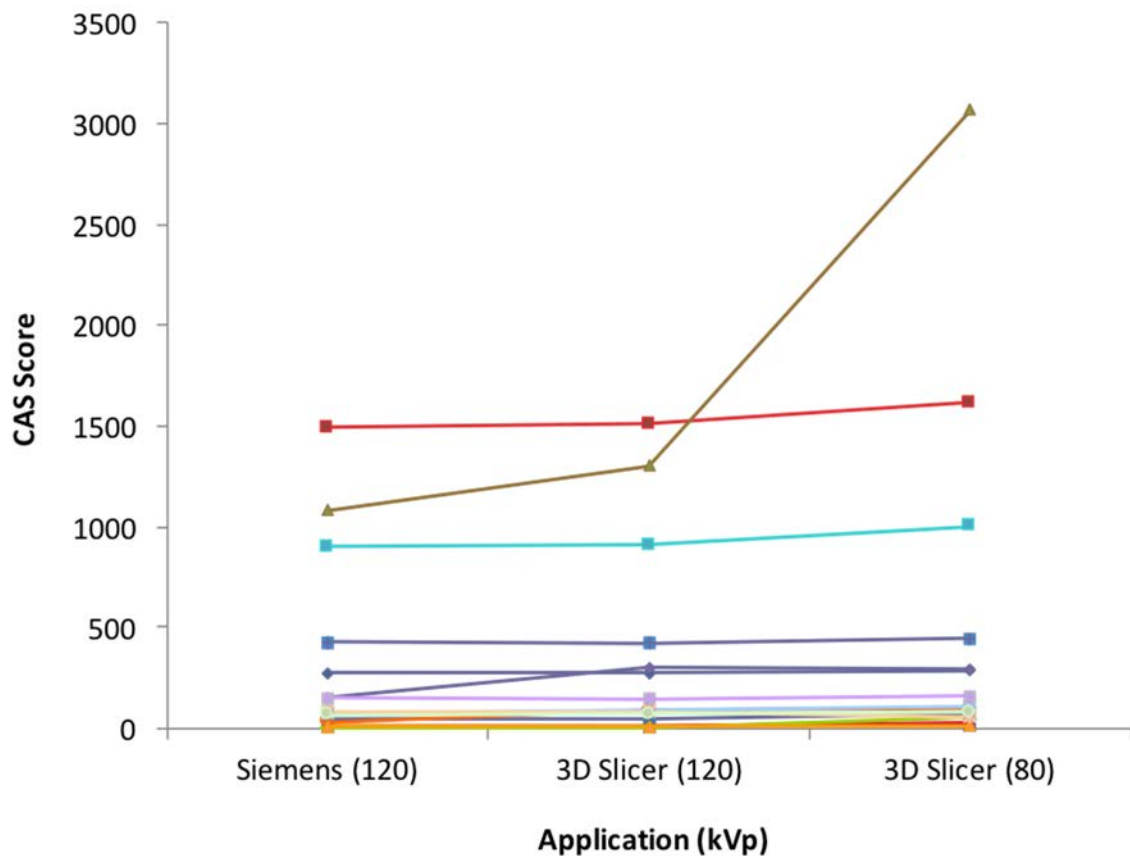
Figure 28: CAS scores for 16 subjects with the Siemens (120 kVp) and 3D Slicer (120 kVp and 80 kvP) applications.

Normal Q-Q Plot
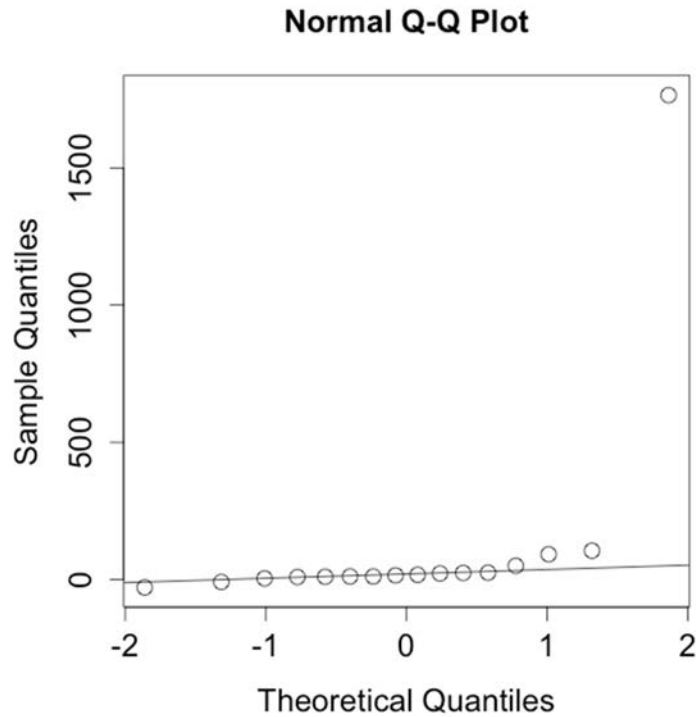
Figure 29: QQ plot for the differences in CAS scores between the 3D Slicer measurements of 120 kVp and 80 kVp scans (n=16).

ICC And T-Test Results

Table 12 lists the intraclass correlation coefficients (ICC) and the paired t-test comparisons of CAS values to assess the agreement between the tools. The results containing all 16 datasets (n=16) and results after the removal of an extreme outlier (n=15) are displayed. Since the CAS data exhibited right skewed distributions, the data were transformed using the natural logarithm before the paired t-tests were conducted to satisfy the assumptions of normality.

66

Table 12: ICC values and t-test results for comparisons of the Siemens tool for 120 kVp and the 3D Slicer tool for both 80 kVp and 120 kVp scans.

Scans (n)	Method 1 Tool	kVp	Method 2 Tool	kVp	ICC value	t-test (p-value)
16	Siemens	120	3D Slicer	120	0.958	0.0641
16	Siemens	120	3D Slicer	80	0.818	0.0237
16	3D Slicer	120	3D Slicer	80	0.929	0.0235
15	Siemens	120	3D Slicer	120	0.952	0.0784
15	Siemens	120	3D Slicer	80	0.791	0.0414
15	3D Slicer	120	3D Slicer	80	0.922	0.0461

The ICC values comparing the Siemens 120 kVp and 3D Slicer 120 kVp methods before outlier removal (Figure 30 top left) and after (Figure 30 bottom left) are highly correlated (ICC=0.958, 0.952). Similarly the ICC value comparing the CAS measurements for the 3D Slicer application at 120 kVp and 80 kVp before outlier removal (Figure 30 top right) were highly correlated (ICC=0.929), as well as highly correlated (ICC=0.922) after the removal of an outlier (Figure 30 bottom right). The ICC values comparing the Siemens 120 kVp and 3D Slicer 80 kVp methods before outlier removal and after were almost perfectly correlated (ICC=0.818, 0.791).

Paired t-test indicates that the Siemens 120 kVp and 3D Slicer 120 kVp methods are not statistically different at a significance level of 0.05 before and after the removal of the outlier (p-value=0.0641, 0.0784). The comparisons between the 120 kVp and 80 kVp scans indicate statistical differences at the significance level of 0.05 before outlier removal (p-value=0.0237, 0.0235) and after (p-value=0.0414, 0.0461).

67
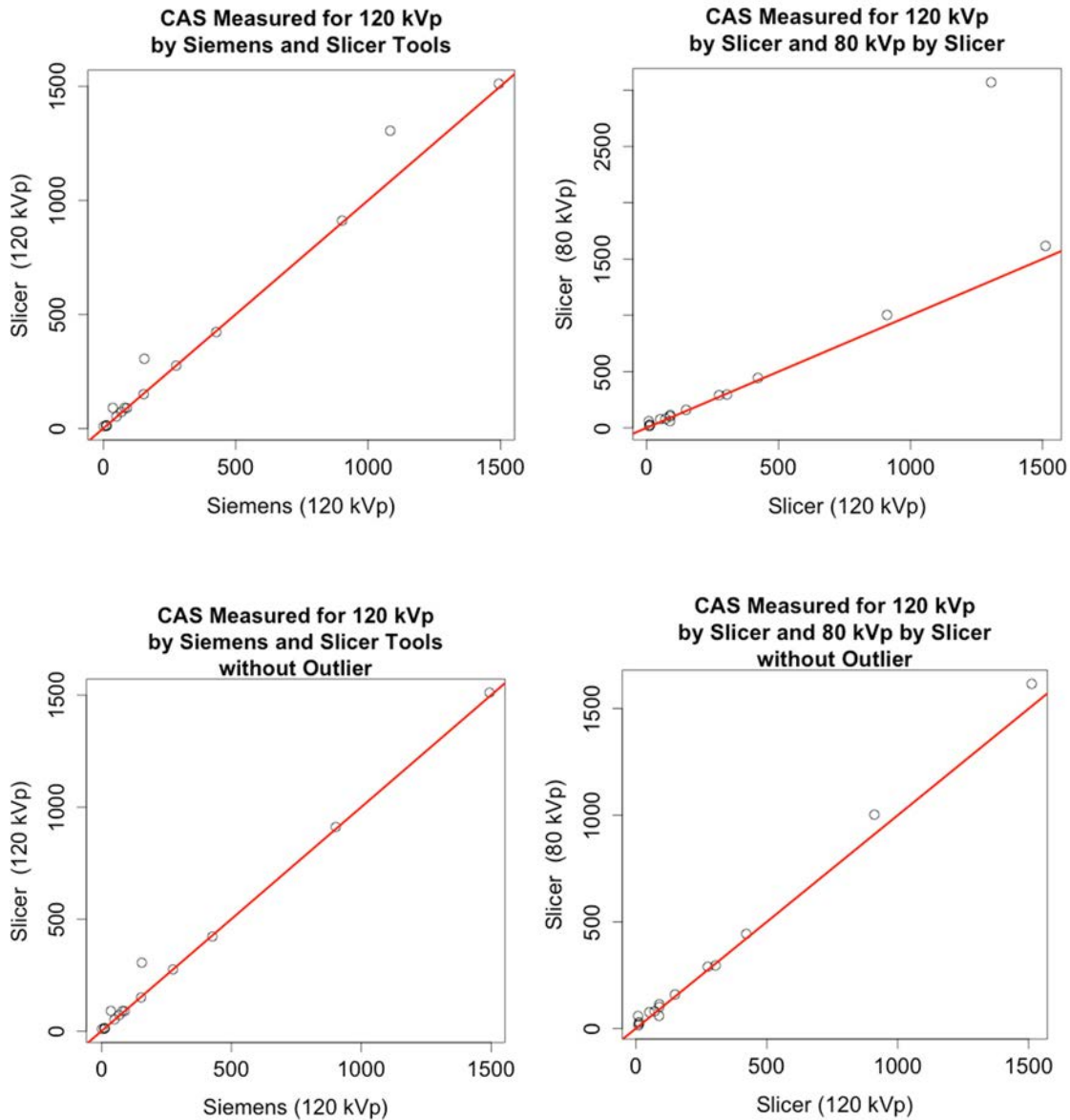
Figure 30: CAS comparisons between different methods. The red line shown in the figure depicts a line with an intercept of 0 and a slope of 1. If the methods gave exactly the same results, all data points would be on this line. (Top left) Siemens versus 3D Slicer at 120 kVp (n=16). (Top right) 3D Slicer measures for 80 kVp and 120 kVp (n=16). (Bottom left) Siemens versus 3D Slicer at 120 kVp (n=15). (Bottom right) 3D Slicer measures for 80 kVp and 120 kVp (n=15).

68

Discussion

The Cardiac Agatston Scoring application was developed to calculate a low dose, low energy protocol for CAC screening. An Energy-Modified Agatston scoring method was developed using energy calibration with 3D co-registration of low energy protocol (80 kVp) and the standard protocol (120 kVp). A statistical analysis of 16 cardiac scans indicated that the results from the Siemens 120 kVp values and the 3D Slicer 120 kVp values were very highly correlated before and after the removal of an outlier. Similarly, the 16 datasets comparing the Slicer 120 kVp and Slicer 80 kVp methods highly correlated before and after the removal of an outlier.

The paired t-test of the Siemens versus Slicer 120 kVp methods indicated that they were not statistically different before or after the removal of an extreme outlier. Although, the paired t-test of the Slicer 80 versus 120 kVp methods indicated that the methods were statistically different before and after the removal of an extreme outlier, the p-values after the removal of the outlier were near the significance level of 0.05 (p-value=0.0414, 0.0461). Even though the CAS for the outlier was much larger for the 80 kVp method than the 120 kVp methods, it was still classified correctly in the highest severity category since the CAS value was larger than 400.

Given that the Siemens and Slicer methods are highly correlated, this application was shown to provide an accurate CAS for 120 kVp scans. The Slicer 120 and 80 kVp scans are also highly correlated and show that accurate CAS at 80 kVp can be obtained through this tool. This efficient, semi-automated tool is freely available through the 3D Slicer Extension Manager to increase the efficiency and improve the availability of Agatston scoring with low radiation in clinical and research areas.

69

CHAPTER 5: WAIST CIRCUMFERENCE MEAUSUREMENTS FROM ABDOMINAL SCANS

Introduction

There is increasing interest in the effect of obesity on surgical outcomes in cancer patients. Obesity was identified as a major unrecognized risk factor for cancer by the American Society of Clinical Oncology (Ligibel et al., 2014). There remains a need to quantify the risk associated with obesity. Various methods exist for measuring obesity including waist circumference (WC), adipose measurement, and intra-abdominal adipose tissue. Currently there is not a computerized application which optimizes the use of these measurements in research and clinical practices (Ciudin et al., 2014; House et al., 2008; Snell-Bergeon et al., 2004). This work offers an application to optimize WC measurement.

Elevated body mass index (BMI=weight divided by height squared) has become synonymous with obesity (Arnold et al., 2015) and is available in clinical practice. However, BMI is an insensitive measure of abdominal obesity and interestingly a large study with self-measured WC found that WC was positively associated with mortality within all categories of BMI (Jacobs et al., 2010). Thus WC is potentially a better indicator of mortality.

WC is typically acquired by manually measuring the circumference of a subject's abdomen just above the navel. It can also be determined through the use of radiographic images such as Computed Tomography (CT) images. This is particularly attractive in patients undergoing surgery for cancer since preoperative abdominal imaging is standard for many cancers. Previously, there was not an optimized and efficient method to

70

measure the WC of patients through radiographic images. This was inhibiting the widespread use of this measurement.

This chapter describes the development of an efficient tool for rapidly and accurately measuring WC in CT scans. It was created within the 3D Slicer environment and is freely distributed by the Extension Manager. Results from the application were compared to hand measurements obtained by using a measuring tape. There was not a statistically significant difference between the measures by the application and the hand method. The two measures were also highly correlated with an ICC value of 0.99.

The goal of this tool is to increase the efficiency and improve the availability of WC in clinical and research areas. An advantage of the module over clinical methods is that large numbers of WCs can be quickly measured in succession by a single reviewer. This reduces inter-rater variability due to the differences in measuring technique between different clinicians. It also allows for generation of WC measurements for patients that were not hand measured in clinic. The images and measures resulting from the use of this tool are saved and are easily retrieved for future review or training purposes.

Methods

This section illustrates the Waist Circumference application and the proposed procedure for measuring a set of CT images. It also contains a description of a test dataset and the applied image pre-processing. Additionally, it contains the results of a statistical analysis that was conducted to compare the computer-derived CT based measurements to the clinical measurements obtained for the same subjects.

Waist Circumference 3D Slicer Module

The Waist Circumference application, illustrated in Figure 31, provides an efficient, semi-automated tool for rapidly measuring WC in large images sets. The interface consists of several sections: (1) *Input Parameters* for two required files, (2) a local version of the *Editor* widget – a suite of label editing tools packaged with 3D Slicer, (3) *Measurements* containing the tabular WC results, and (4) the *Imaging Viewing* area.
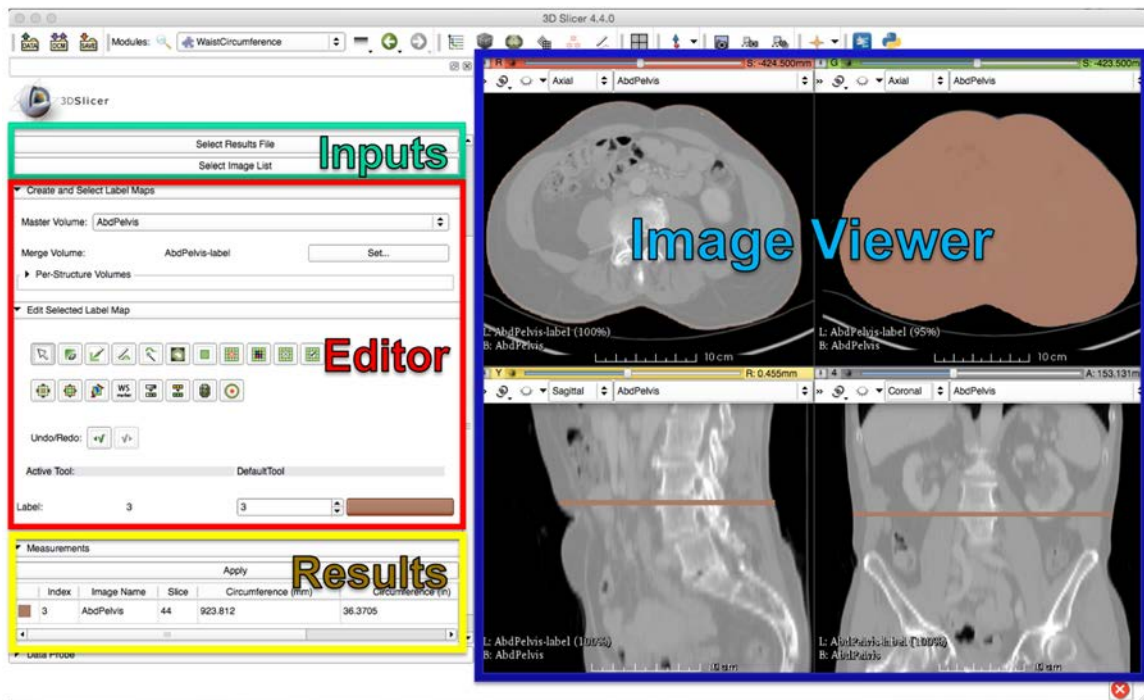
Figure 31: Interface for the Waist Circumference measurement tool.

There are two input files selected in the *Input Parameters* section that are required: (1) a *Results* file to contain the measured WC values for all scans and (2) an *Image List* file containing the absolute file path for each abdominal CT scan on separate rows. An Image List file is utilized to efficiently and automatically load images in succession and save results to a common *Results* file.

72

The *Results* file is formatted as a comma separated values (CSV) file. This file can be reviewed using a text editor or with the commonly used Excel application. If this file does not exist, the module creates a new file with an appropriate header row. The WC value(s) for each scan will then be appended to the end of this file.

Once the *Image List* file is selected, the first scan will automatically open. A label image of the same dimensions as the CT scan is automatically created via the Python interface for the *Editor* widget. To edit the newly created label image with the *Editor* widget, the input image is automatically set as the *Master Volume* and the label image is set as the *Merge Volume*.
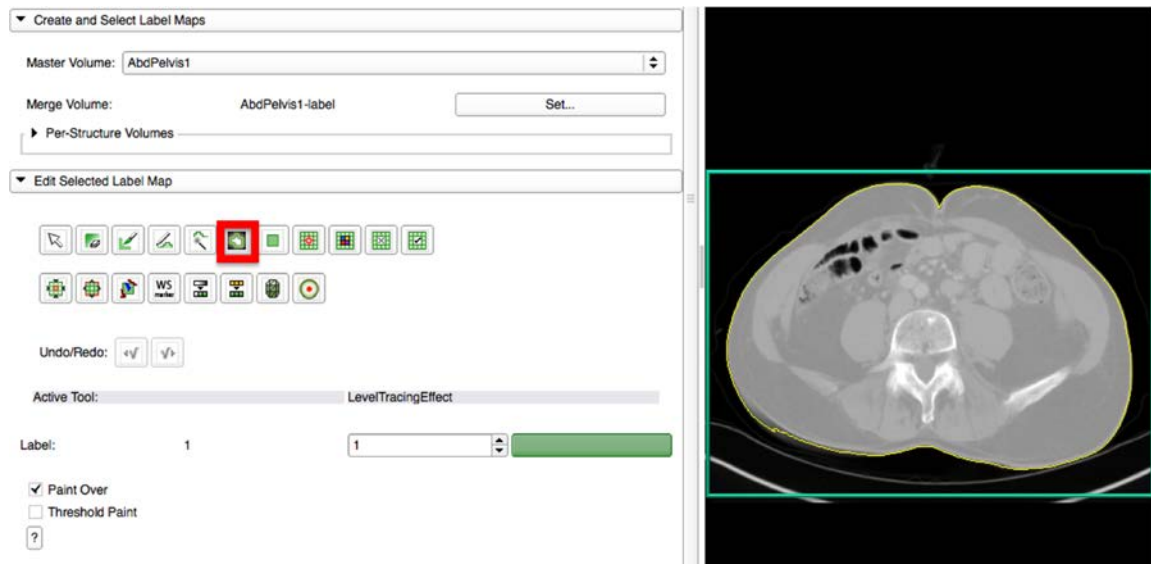
Figure 32: Display of waist circumference outline (in yellow)

Editing tools within the *Editor* module allow the user to create a WC label in the axial plane very quickly, often with a single mouse click. The *Level Tracing Editing* tool, outlined in red in Figure 32, allows the user to select a single point on the boundary of an

object and automatically create a border of the pixels with the same intensity value
(Pieper et al., 2014b). An example WC outline is shown in yellow in Figure 32.

In Figure 33, the two red arrows point to the top and bottom of the umbilicus.
The slice directly superior to the last axial slice in which indentation of the umbilicus is
visible (indicated by the blue line) was used for measuring the WC. This corresponds to
the region measured via the tape measure in clinic.
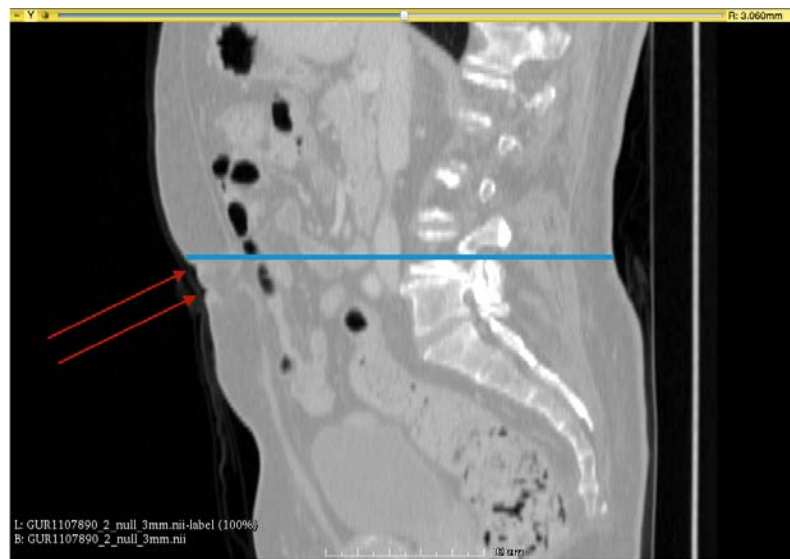
Figure 33: Sagittal display of an abdominal CT scan. The
blue line indicates the axial slice used to measure the WC.
The red arrows indicate the top and bottom of the navel.

The user can create labels on different axial slices as illustrated in Figure 34.
There the red and green objects represent different labels on separate axial slices. Those
measurements are displayed in the results table with the corresponding slice identifier and
label color as seen in Figure 35.

74
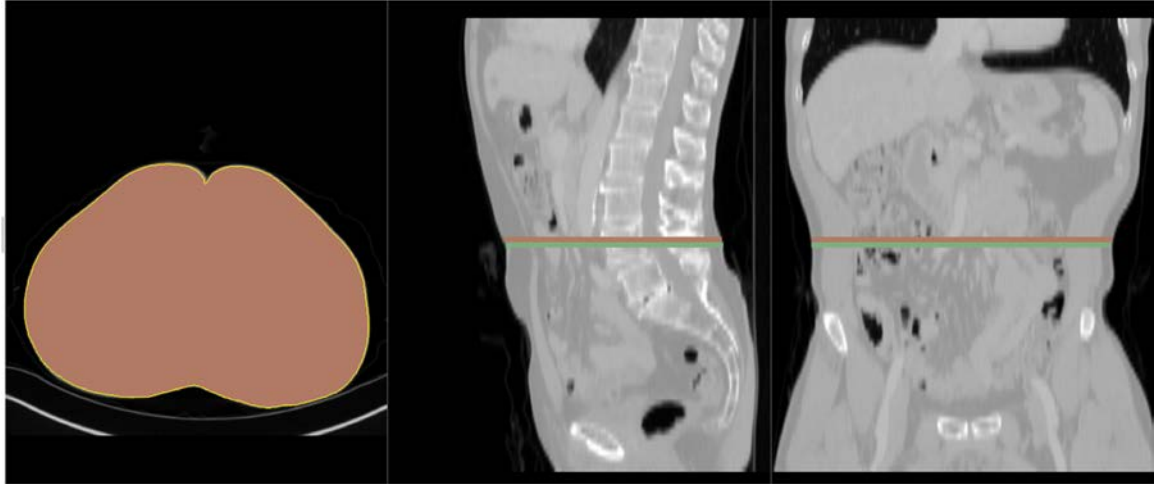
Figure 34: Axial, sagittal, and coronal views of an abdominal CT scan. The red and green lines in the sagittal and coronal views show the location of two labels created in the axial plane.

Selection of the "Apply" button calculates the perimeter of the labeled objects in each slice. The algorithm iterates through each axial slice and identifies each label with the SimpleITK image filter *LabelShapeStatisticsImageFilter*. If there is no label in the current axial slice the next slice is examined. The filter returns the perimeter in millimeters for each object in the two-dimensional label image. This information is then printed to the results table, and the circumference is displayed in millimeters and inches as seen in Figure 35.

The user selects "Save and Next" when satisfied with the label selections. This button performs several steps: (1) it appends the data in the results table to the output *Results* file, (2) it creates an *Output Folder* named after the *Master Volume*, (3) it saves the *Master* and *Merge* images to the *Output Folder*, (4) it saves a CSV file with this subject's WC measures from the results table, (5) it saves the MRML scene containing links to the volume files, (6) it closes the current MRML scene to close the current

75

volumes, and then (7) it opens the next scan in the input *Image List* file. If the saved

MRML scene is later opened with 3D Slicer, the volumes are automatically loaded and

positioned in the same view as when the MRML scene was saved. This allows for quick

and easy review or revision of previous measurements.

Figure 35: Illustration of printed waist circumference
measurements in millimeters and inches for two axial slices.

Dataset

The Waist Circumference tool was tested on a set of 14 preoperative cystectomy

CT scans from the Urology department at the University of Iowa Hospitals and Clinics

(UIHC). They are from established UIHC urology cohorts with renal cancer and bladder

cancer in a method with current IRB approval. These scans have matching WC data

measured in clinic with a tape measure by one of three urologists. In some cases, the

hand measure did not occur on the same day as the CT scan, but they were close enough

in time that minimal change in body shape was to be expected.

Initially the set consisted of 20 CT scans with matching WC values. Prior to

statistical analysis, 6 of 20 scans were excluded: 3 because part of the abdomen lies

outside of the field of view on the CT scan; 2 due to image quality issues which occurred

76

when the images were transferred to a CD by the Radiology department; and 1 due to an abdominal implant or foreign body that concealed the umbilicus.

Image Pre-processing

All original images were converted from the Digital Imaging and Communications in Medicine (DICOM) standard via tools provided by the *BRAINSTools* suite (BRAINSia, 2015). *ConvertBetweenFileFormats* was used to convert the abdominal CT images from DICOM format to Neuroimaging Informatics Technology Initiative (NifTI). The absolute file paths for the converted images were then stored in a file for use with the Waist Circumference module. The application uses this file to automatically read input images, name each scan, and save the results to a folder based on the scan's name. A simple text file is used as input because it is easy to manipulate and works on all platforms.

Statistical Analysis

The WC tool measurements were compared to the hand measurements of the same subjects. The Intraclass Correlation Coefficient (ICC) was calculated to measure the reliability of the two methods. Since one of three urologists measured the WC in clinic and were chosen by random, the single raters absolute ICC value was calculated. This ICC value is sensitive to differences in means between raters and is a measure of absolute agreement. A two-sided paired t-test was conducted to test if the mean difference of the 14 paired "hand-based" and "Slicer-based" WC measurements were statistically different than zero. The null hypothesis was that the mean of the differences was equal to zero and the alternative hypothesis was that the mean difference was not equal to zero. The statistical significance level was set to 0.05.

77

As seen in Figure 36, "hand-based" and "Slicer-based" methods gave very similar results. The red line shown in the figure depicts a line with an intercept of 0 and a slope of 1. If the methods gave exactly the same results, all data points would be on this line. The largest difference between measures occurred on the scan with the smallest waist as measured by hand.

Waist Circumference Measured by Hand Versus Slicer Tool
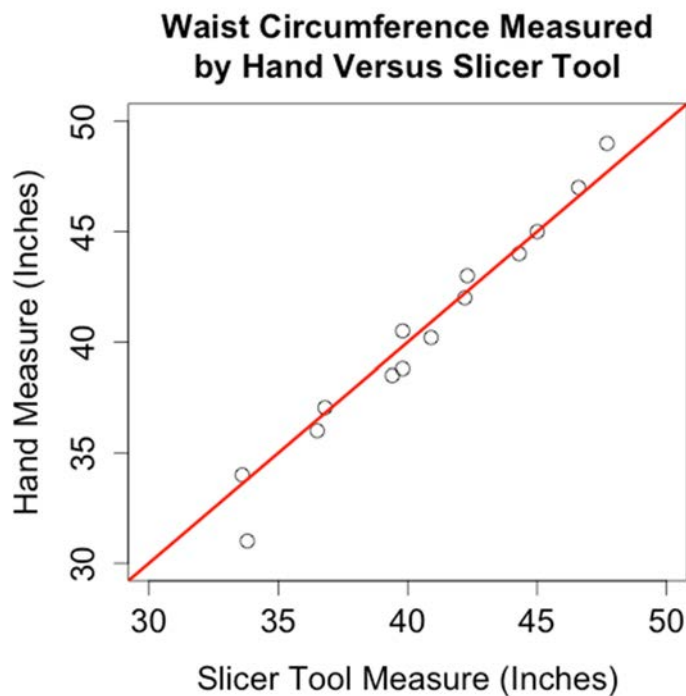
Figure 36: Displays the WC measures for the Hand and Slicer tool methods. The red line has an intercept of 0 and slope of 1.

As shown in the boxplot in Figure 37 the median difference value was 0.1 inches, the first quartile was -0.4 inches, the third quartile was 0.65 inches, and the maximum difference value was 2.8 inches. The single raters absolute ICC value of 0.99 (p-value=3.3e-11) indicates that the two measures are highly correlated.

78

A two-sided paired t-test was performed to determine if the means of the two methods were statistically the same. It revealed that the mean difference was not significantly different from zero. That is, the "hand-based" measures were not significantly different from the "Slicer-based" measures ($t = -0.70632$, $df = 13$, p-value = 0.4925).

Waist Circumference Differences Measured by Hand Versus Slicer Tool

Difference in Measures (Inches)
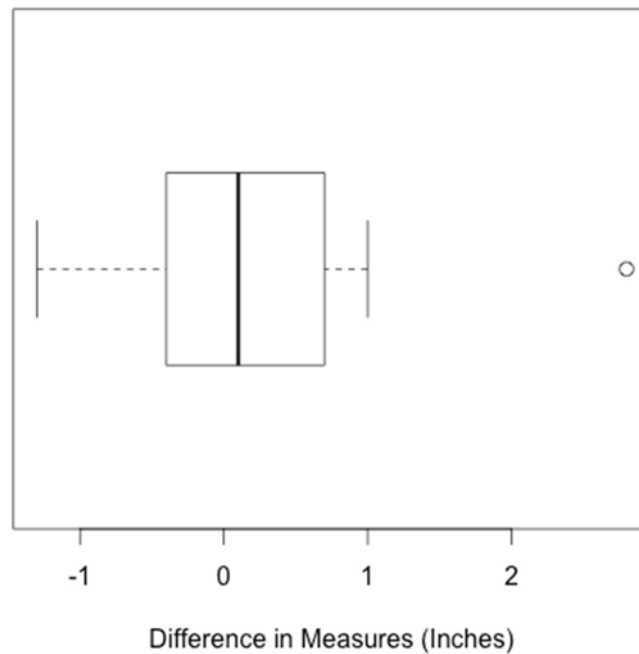
Figure 37: A boxplot of the difference in WC measures between the Hand and Slicer methods.

Discussion

Despite the proliferation of waist circumference as a measurement of obesity in the literature, the measurement of this factor was not previously optimized. The prior radiographic measurement tools were inefficient and not readily accessible in clinical practice. The 3D Slicer Waist Circumference tool was developed to increase efficiency

79

and improve the availability of WC in clinical and research areas. The application is open-source and freely distributed within the 3D Slicer Extension Manager.

One advantage of using the application compared to clinical measures is the potential reduction or elimination of inter-rater variability since an entire dataset could be measured very quickly with this tool and the previous WC measures could be reviewed to ensure consistency. Variability of hand measurements in clinic can arise due to multiple measurers and methods.

The application could also be used to stratify subjects into levels of obesity that span several inches as seen in Jacobs et al. In that study, they separated subjects into WC categories using 10-cm (3.94 inch) increments. They identified a clinically defined threshold for abdominal obesity of WC as $\geq$ 88 cm (~35 inches) in women and $\geq$ 102 cm (~ 40 inches) in men (Jacobs et al., 2010). Due to the large range within each cut-off level, the Waist Circumference tool could be used to assign subjects to specific levels with a high certainty.

The single raters absolute ICC value of 0.99 indicated that WC measures of the "Slicer-based" and the "hand-based" methods are highly correlated. Furthermore, the two-sided paired t-test indicates that the difference between the paired measurements by hand and the application is not statistically significant. These results show that the Waist Circumference tool can be used to measure the WC in an efficient way without compromising accuracy.

80

CHAPTER 6: DISCUSSION

Several modules were developed within the 3D Slicer environment to help optimize the review of images, perform rule-based automated label map cleaning, identify and quantify calcification in the heart, and measure waist circumference from abdominal scans. Each of these modules enables and accelerates clinical research by incorporating medical imaging technologies that minimize manual human effort. They are distributed within the multi-platform 3D Slicer Extension Manager environment that allows use in the computational environment most convenient to the clinician scientist.

The BRAINSImageEval stand-alone software was rewritten as a 3D Slicer module with the same functionality as the original tool. The program, rewritten in Python, is more streamlined and more flexible than the C++ version. The 3D Slicer module functions as a more efficient replacement for the stand-alone tool. It will also allow for the reuse of components of this module for other QC modules in an imaging analysis suite.

This work also described the creation of a multi-atlas dataset for the entire brain region that spans a range of HD progression and MRI scanners. These atlases are expected to improve the future atlas-based MRI analysis on HD. The tool was demonstrated to decrease the editing time for the 20 considered atlases by providing automatic and semi-automatic reassignment of islands. After the correction of atlas inconsistencies and small, disconnected regions, the number of unidentified voxels for each dataset was reduced on average by 68.48%. Additionally, new regions of interest were efficiently identified and included in the atlases.

The Cardiac Agatston Scoring application was developed to calculate the low dose, low energy protocol for CAC screening using a low energy protocol. An Energy-Modified Agatston scoring method was developed using energy calibration with 3D co-registration of low energy protocol (80 kVp) and the standard protocol (120 kVp). It provides a novel method for Agatston scoring with low radiation. A statistical analysis of a set of cardiac scans indicated that the Siemens and 3D Slicer CAS methods for 120 kVp scans are highly correlated and the 3D Slicer application provided an accurate CAS for 120 kVp scans. The Slicer 120 and 80 kVp scans are also highly correlated and show that accurate CAS at 80 kVp can be obtained with this tool.

The use of waist circumference as a measurement of obesity is rapidly increasing in the literature, but the measurement of this factor was not previously optimized. A new radiographic measurement tool was developed that is very efficient and readily accessible in clinical practice. In a statistical analysis of 16 WC measures of the "Slicer-based" and the "hand-based" methods, the single raters absolute ICC value of 0.99 indicated that the two methods are highly correlated. Furthermore, the two-sided paired t-test indicates that the difference between the paired measurements by hand and the application are not statistically different. These results show that the Waist Circumference tool can be used to estimate the WC in an efficient and reproducible way.

82

CHAPTER 7: FUTURE WORK

In a small percentage of WC subject images, an issue arose where the subject was too large to be contained within the field of view. In these cases, one or both sides of the subject are not present in the scan. Future work may include estimating the WC for the incomplete data via a semi-automated method where the user selects several points along the border of the incomplete image and the missing circumference piece is estimated from this information.

The WC tool may also be modified to measure sarcopenia, the loss of muscle mass, via the psoas muscle. Sarcopenia is used to assess the effect of frailty on adverse surgical outcomes and has been identified with increased complications in women after cystectomy (Makary et al., 2010; Smith et al., 2014).

APPENDIX A

```xml
<?xml version="1.0" encoding="UTF-8"?>

<phd:ImageReview ID="" project="" label="" >

<xnat:date></xnat:date>

<xnat:time></xnat:time>

<xnat:imageSession_ID></xnat:imageSession_ID>

<phd:series_number></phd:series_number>

<phd:formdescriptor>

<phd:field name="Normal variants" help="Does the image show normal variants?" value="" type="YesNo"/>

<phd:field name="Lesions" help="Does the image show lesions?" value="" type="YesNo"/>

<phd:field name="SNR" help="Overall SNR weighted images 0=bad 10=good" value="" type="Range"/>

<phd:field name="CNR" help="Overall CNR weighted images 0=bad 10=good" value="" type="Range"/>

<phd:field name="Full Brain Coverage" help="Is the whole brain visible in the image?" value="" type="YesNo"/>

<phd:field name="Misalignment" help="Does the image show misalignment?" value="" type="YesNo"/>

<phd:field name="Swap / Wrap Around" help="Does the image show swap / wrap around?" value="" type="YesNo"/>

<phd:field name="Ghosting / Motion" help="Are there motion artifacts in the image?" value="" type="YesNo"/>

<phd:field name="Inhomogeneity" help="Does the image show Inhomgeneity?" value="" type="YesNo"/>

<phd:field name="Susceptibility/Metal" help="Does the image show susceptibility?" value="" type="YesNo"/>

<phd:field name="Flow artifact" help="Does the image show flow artifact?" value="" type="YesNo"/>
```

84

```xml
<phd:field name="Truncation artifact" help="Does the image show truncation?" value=""
type="YesNo"/>

<phd:field name="overall QA assessment" help="0=bad 10=good" value=""
type="Range"/>

<phd:field name="Evaluator" help="Name of person evalating this scan" value=""
type="String"/>

<phd:field name="Image File" help="Name of the image file being evaluated" value=""
type="String"/>

<phd:field name="Free Form Notes" help="Mention anything unusual or significant
about the images here" value="" type="TextEditor"/>

<phd:field name="Evaluation Completed" help="Is the evaluation completed? (No
implies further evaluation needed)" value="" type="YesNo"/>

</phd:formdescriptor>

</phd:ImageReview>
```

APPENDIX B

```xml
<ResultSet totalRecords="5667" title="scans">

    <results>

        <columns>

            <column>project</column>

            <column>subject_id</column>

            <column>subject</column>

            <column>session_id</column>

            <column>session</column>

            <column>date</column>

            <column>time</column>

            <column>seriesnumber</column>

            <column>type</column>

            <column>quality</column>

            <column>reviewed</column>

            <column>status</column>

            <column>element_name</column>

            <column>type_desc</column>

            <column>insert_date</column>

            <column>activation_date</column>

            <column>last_modified</column>

        </columns>

    </results>

</ResultSet>
```

REFERENCES

Ahdidan, J., Hviid, L. B., Chakravarty, M. M., Ravnkilde, B., Rosenberg, R., Rodell, A., et al. (2011). Longitudinal MR study of brain structure and hippocampus volume in major depressive disorder. *Acta Psychiatr. Scand.* 123, 211–9. doi:10.1111/j.1600-0447.2010.01644.x.

Arnold, M., Pandeya, N., Byrnes, G., Renehan, A. G., Stevens, G. A., Ezzati, M., et al. (2015). Global burden of cancer attributable to high body-mass index in 2012: a population-based study. *Lancet. Oncol.* 16, 36–46. doi:10.1016/S1470-2045(14)71123-4.

Aucoin, N. (2014). Markups SlicerWiki. Available at: http://www.slicer.org/slicerWiki/index.php/Documentation/4.4/Modules/Markups [Accessed August 7, 2015].

BRAINSia (2015). BRAINSTools. Available at: https://github.com/BRAINSia/BRAINSTools.

Budoff, M. J., and Gul, K. M. (2008). Expert review on coronary calcium. *Vasc. Health Risk Manag.* 4, 315–24. Available at: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2496978&tool=pmcentrez&rendertype=abstract [Accessed February 25, 2016].

Cabezas, M., Oliver, A., Llado, X., Freixenet, J., and Cuadra, M. B. (2011). A review of atlas-based segmentation for magnetic resonance brain images. *Comput Methods Programs Biomed* 104, e158–177.

CDC (2015). Heart Disease Facts & Statistics. Available at: http://www.cdc.gov/heartdisease/facts.htm [Accessed February 24, 2016].

Ciudin, A., Salvador, R., Budoy, A., Ciudin, A., Spinu, C., Diaconu, M. G., et al. (2014). Measurement of waist circumference for retrospective studies - prospective validation of use of CT images to assess abdominal circumference. *Endocrinol. y Nutr. órgano la Soc. Española Endocrinol. y Nutr.* 61, 147–52. doi:10.1016/j.endonu.2013.10.004.

Fedorov, A., Beichel, R., Kalpathy-Cramer, J., Finet, J., Fillion-Robin, J.-C., Pujol, S., et al. (2012). 3D Slicer as an image computing platform for the Quantitative Imaging Network. *Magn. Reson. Imaging* 30, 1323–41. doi:10.1016/j.mri.2012.05.001.

Forbes, J. (2015). ImageEval. Available at: https://github.com/BRAINSia/SlicerQCExtensions/tree/sqc/SlicerQCExtensions/ImageEval.

87

Ghayoor, A., Vaidya, J. G., and Johnson, H. J. (2013). Development of a novel constellation based landmark detection algorithm. in *SPIE Medical Imaging*, eds. S. Ourselin and D. R. Haynor (International Society for Optics and Photonics), 86693F. doi:10.1117/12.2006471.

Herting, M. M., Gautam, P., Spielberg, J. M., Kan, E., Dahl, R. E., and Sowell, E. R. (2014). The role of testosterone and estradiol in brain volume changes across adolescence: a longitudinal structural MRI study. *Hum. Brain Mapp.* 35, 5633–45. doi:10.1002/hbm.22575.

Hoffmann, U., Brady, T. J., and Muller, J. (2003). Cardiology patient page. Use of new imaging techniques to screen for coronary artery disease. *Circulation* 108, e50–3. doi:10.1161/01.CIR.0000085363.88377.F2.

House, M. G., Fong, Y., Arnaoutakis, D. J., Sharma, R., Winston, C. B., Protic, M., et al. (2008). Preoperative predictors for complications after pancreaticoduodenectomy: impact of BMI and body fat distribution. *J. Gastrointest. Surg.* 12, 270–8. doi:10.1007/s11605-007-0421-7.

Jacobs, E. J., Newton, C. C., Wang, Y., Patel, A. V, McCullough, M. L., Campbell, P. T., et al. (2010). Waist circumference and all-cause mortality in a large US cohort. *Arch. Intern. Med.* 170, 1293–301. doi:10.1001/archinternmed.2010.201.

Jones, K., and Marietta, J. (2012). Quality Analysis of Raw MRI Scans Using BRAINSImageEval.

Kim, E. Y., and Johnson, H. J. (2013). Robust multi-site MR data processing: iterative optimization of bias correction, tissue classification, and registration. *Front. Neuroinform.* 7, 29. doi:10.3389/fninf.2013.00029.

Kim, R. E. Y., Lourens, S., Long, J. D., Paulsen, J. S., and Johnson, H. J. (2015). Preliminary analysis using multi-atlas labeling algorithms for tracing longitudinal change. *Front. Neurosci.* 9, 1–11. doi:10.3389/fnins.2015.00242.

Kitware (2015). sitkUtils Module — Slicer 4.3 documentation. Available at: http://mwoehlke-kitware.github.io/Slicer/Base/sitkUtils.html [Accessed March 3, 2016].

Kitware (2016). ITK - Segmentation & Registration Toolkit. Available at: http://www.itk.org/ [Accessed February 11, 2016].

Li, X.-W., Li, Q.-L., Li, S.-Y., and Li, D.-Y. (2015). Local manifold learning for multiatlas segmentation: application to hippocampal segmentation in healthy population and Alzheimer's disease. *CNS Neurosci. Ther.* 21, 826–36. doi:10.1111/cns.12415.

88

Ligibel, J. A., Alfano, C. M., Courneya, K. S., Demark-Wahnefried, W., Burger, R. A., Chlebowski, R. T., et al. (2014). American Society of Clinical Oncology position statement on obesity and cancer. *J. Clin. Oncol.* 32, 3568–74. doi:10.1200/JCO.2014.58.4680.

Lorensen, B. (2015). OpenAtlas - NAMIC 2015 Winter Project Week. Available at: http://www.na-mic.org/Wiki/index.php/2015_Winter_Project_Week:OpenAtlas [Accessed October 22, 2015].

Lowekamp, B. C., Chen, D. T., Ibáñez, L., and Blezek, D. (2013). The Design of SimpleITK. *Front. Neuroinform.* 7, 14. doi:10.3389/fninf.2013.00045.

Lu, W. (2010). A method for automated landmark constellation detection using evolutionary principal components and statistical shape models. *Theses Diss.* Available at: http://ir.uiowa.edu/etd/851 [Accessed November 11, 2015].

Makary, M. A., Segev, D. L., Pronovost, P. J., Syin, D., Bandeen-Roche, K., Patel, P., et al. (2010). Frailty as a predictor of surgical outcomes in older patients. *J. Am. Coll. Surg.* 210, 901–8. doi:10.1016/j.jamcollsurg.2010.01.028.

Mori, S., Wakana, S., Van Zijl, P. C. M., and Nagae-Poetscher, L. M. (2005). *MRI atlas of human white matter*. Am Soc Neuroradiology.

Mungas, D., Harvey, D., Reed, B. R., Jagust, W. J., DeCarli, C., Beckett, L., et al. (2005). Longitudinal volumetric MRI change and rate of cognitive decline. *Neurology* 65, 565–571. doi:10.1212/01.wnl.0000172913.88973.0d.

NIH (2016). 10-year CVD Risk Calculator (Risk Assessment Tool for Estimating Your 10-year Risk of Having a Heart Attack Version). Available at: http://cvdrisk.nhlbi.nih.gov/#cholesterol [Accessed February 25, 2016].

Paulsen, J. S., Magnotta, V. a, Mikos, A. E., Paulson, H. L., Penziner, E., Andreasen, N. C., et al. (2006). Brain structure in preclinical Huntington's disease. *Biol. Psychiatry* 59, 57–63. doi:10.1016/j.biopsych.2005.06.003.

Pieper, S., Isomics, Quintus, K., and BWH (2014a). LabelStatistics SlicerWiki. Available at: https://www.slicer.org/slicerWiki/index.php/Documentation/4.4/Modules/LabelStatistics [Accessed March 3, 2016].

Pieper, S., Plesniak, W., Kikinis, R., and Miller, J. (2014b). Editor SlicerWiki. Available at: http://www.slicer.org/slicerWiki/index.php/Documentation/4.4/Modules/Editor [Accessed August 7, 2015].

Pierson, R., Johnson, H., Harris, G., Keefe, H., Paulsen, J. S., Andreasen, N. C., et al. (2011). Fully automated analysis using BRAINS: AutoWorkup. *Neuroimage* 54, 328–36. doi:10.1016/j.neuroimage.2010.06.047.

Resnick, S. M., Pham, D. L., Kraut, M. A., Zonderman, A. B., and Davatzikos, C. (2003). Longitudinal magnetic resonance imaging studies of older adults: a shrinking brain. *J. Neurosci.* 23, 3295–301. Available at: http://www.ncbi.nlm.nih.gov/pubmed/12716936 [Accessed October 6, 2015].

Risacher, S. L., Shen, L., West, J. D., Kim, S., McDonald, B. C., Beckett, L. A., et al. (2010). Longitudinal MRI atrophy biomarkers: relationship to conversion in the ADNI cohort. *Neurobiol. Aging* 31, 1401–18. doi:10.1016/j.neurobiolaging.2010.04.029.

Scahill, R. I., Frost, C., Jenkins, R., Whitwell, J. L., Rossor, M. N., and Fox, N. C. (2003). A longitudinal study of brain volume changes in normal aging using serial registered magnetic resonance imaging. *Arch. Neurol.* 60, 989–94. doi:10.1001/archneur.60.7.989.

Slicer (2015a). ExtensionWizard - SlicerWiki. Available at: https://www.slicer.org/slicerWiki/index.php/Documentation/Nightly/Developers/ExtensionWizard.

Slicer (2015b). MRML - SlicerWiki. Available at: https://www.slicer.org/slicerWiki/index.php/Documentation/Nightly/Developers/MRML [Accessed March 28, 2016].

Smith, A. B., Deal, A. M., Yu, H., Boyd, B., Matthews, J., Wallen, E. M., et al. (2014). Sarcopenia as a predictor of complications and survival following radical cystectomy. *J. Urol.* 191, 1714–20. doi:10.1016/j.juro.2013.12.047.

Snell-Bergeon, J. K., Hokanson, J. E., Kinney, G. L., Dabelea, D., Ehrlich, J., Eckel, R. H., et al. (2004). Measurement of abdominal fat by CT compared to waist circumference and BMI in explaining the presence of coronary calcium. *Int. J. Obes. Relat. Metab. Disord.* 28, 1594–9. doi:10.1038/sj.ijo.0802796.

Sullivan, E. V., Pfefferbaum, A., Rohlfing, T., Baker, F. C., Padilla, M. L., and Colrain, I. M. (2011). Developmental change in regional brain structure over 7 months in early adolescence: Comparison of approaches for longitudinal atlas-based parcellation. *Neuroimage* 57, 214–224. doi:10.1016/j.neuroimage.2011.04.003.

Tabrizi, S. J., Reilmann, R., Roos, R. A., Durr, A., Leavitt, B., Owen, G., et al. (2012). Potential endpoints for clinical trials in premanifest and early Huntington's disease in the TRACK-HD study: analysis of 24 month observational data. *Lancet Neurol.* 11, 42–53. doi:10.1016/S1474-4422(11)70263-0.

Takahashi, T., Kido, M., Nakamura, K., Furuichi, A., Zhou, S.-Y., Kawasaki, Y., et al. (2012). Longitudinal MRI study of the pituitary volume in chronic schizophrenia: A preliminary report. *Psychiatry Res. Neuroimaging* 202, 84–87. doi:10.1016/j.pscychresns.2011.11.008.

Tang, Y., Whitman, G. T., Lopez, I., and Baloh, R. W. (2001). Brain volume changes on longitudinal magnetic resonance imaging in normal older people. *J. Neuroimaging* 11, 393–400. Available at: http://www.ncbi.nlm.nih.gov/pubmed/11677879 [Accessed October 22, 2015].

Treit, S., Lebel, C., Baugh, L., Rasmussen, C., Andrew, G., and Beaulieu, C. (2013). Longitudinal MRI reveals altered trajectory of brain development during childhood and adolescence in fetal alcohol spectrum disorders. *J. Neurosci.* 33, 10098–109. doi:10.1523/JNEUROSCI.5004-12.2013.

Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., et al. (2002). Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *Neuroimage* 15, 273–289. doi:10.1006/nimg.2001.0978.

Wang, H., Suh, J. W., Das, S. R., Pluta, J., Craige, C., and Yushkevich, P. A. (2012). Multi-Atlas Segmentation with Joint Label Fusion. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 611–623. doi:10.1109/TPAMI.2012.143.

Wang, H., and Yushkevich, P. A. (2013). Multi-atlas segmentation with joint label fusion and corrective learning—an open source implementation. *Front. Neuroinform.* 7, 27. doi:10.3389/fninf.2013.00027.

Weiner, M. W., Veitch, D. P., Aisen, P. S., Beckett, L. A., Cairns, N. J., Green, R. C., et al. (2012). The Alzheimer's Disease Neuroimaging Initiative: a review of papers published since its inception. *Alzheimers. Dement.* 8, S1–68. doi:10.1016/j.jalz.2011.09.172.

WHO (2015). Cardiovascular diseases. Available at: http://www.who.int/mediacentre/factsheets/fs317/en/ [Accessed February 24, 2016].

Williams, N. (2012). BRAINSImageEval. Available at: https://github.com/BRAINSia/BRAINSImageEval.