SIEMENS

Algorithm Validation Toolkit
AVT2EXT

Algorithm Validation Toolkit
AVT2EXT

Vision, Scope and Technical Overview
Revision 1.0

Last Change: May 3, 2010

Robert W. Schwanke

History

Document History

Version/ Status	Date of Issue	Author	Change and Reason of Change
0.1 Draft		Robert W. Schwanke	Final draft prior to AVT2EXT release
1.0 / released	14-May-2010	Robert W. Schwanke	Released

History of released Versions

Version	Release date	Product Version

Table of Contents

1. Introduction

1.1. Vision

The Algorithm Validation Toolkit (AVT) provides a suite of applications and a database for conducting image quantification ("measurement") experiments and analyzing the results. It is provided to the caBIG community as open source software, so that caBIG-sponsored quantification studies can adapt it to their needs and then publish their software along with their results, facilitating follow-on research that adds new images, new quantification algorithms, new analysis methods, and other innovations.

As AVT is used and extended by caBIG-sponsored projects, the growing code base will become more and more useful for collaborative research on image quantification.

1.2. Scope

The scope of AVT2EXT is tumor quantification experiments on the thoracic phantom images collected by the FDA's Center for Devices and Radiologic Health (CDRH), comparing RECIST, WHO, and volumetric measurements of single tumors. The software should be easily adaptable for use on images of real tumors, but this capability has not been tested.

1.3. Definitions and abbreviations

See *AVT2EXT Definitions and Abbreviations,* a separate document.

1.4. References

[1] AVT2EXT Administration Guide , version 1.0
[2] AVT2EXT Definitions and Abbreviations, version 1.0
[3] AVT2EXT Design Specification, version 1.0
[2] AVT2EXT Functional Specification, version 1.0
[4] AVT2EXT Image Annotation User Manual, version 1.0
[5] AVT2EXT Implementation Plan, version 1.0
[6] AVT2EXT Installation Guide, version 1.0
[7] AVT2EXT Measurement Variability Tool User Manual, version 1.0
[8] AVT2EXT Programming Guide, version 1.0
[9] AVT2EXT Requirement Specification, version 1.0
[10] AVT2EXT Vision, Scope, and Technical Overview, version 1.0
www.openxip.org

2. Technical Overview

2.1. Data

AVT deals with 4 main kinds of data:

- DICOM Series
- AIM/XML objects, in 4 kinds

  - Tumor Seed Annotation: contains a line segment lying entirely within an observed tumor. (Segmentation algorithms requiring only a single seed point would typically start at the midpoint of this line segment.)
  - Tumor RECIST Annotation: contains a line segment representing the RECIST diameter of a tumor.
  - Tumor WHO Annotation: contains two perpendicular line segments representing the World Health Organization's method of measuring tumors by measuring the largest pair of orthogonal diameters.
  - Tumor Volume Annotation: contains contours and/or a reference to (technically, a SOPInstanceUID for) a DICOM segmentation object, representing the 3D surface of a tumor.
- DICOM Segmentation objects
- Statistical analysis results, in several formats.

Note: AVT does not have the necessary capabilities and safeguards for handling Protected Healthcare Information (PHI), as defined by HIPAA. It should only be used with de-identified data.

2.2. Software Components

AVT consists of the following main components:

- **Image Annotation (IA)**, an interactive tool for annotating and marking up images
- **Algorithm Execution (AE)**, a batch tool for executing an annotation and/or markup algorithm on a batch of DICOM series
- **Measurement Variability Tool (MVT)**, an interactive tool for exploring and statistically analyzing a collection of annotations and markups
- **Annotation Database (AD)**, a database containing the DICOM series and AIM/XML objects needed for and/or produced by a quantification experiment.
- **XIPHost/AVT**, a variant of XIP Host™ that connects the components above to each other and to caGrid. (XIP Host™, XIP Builder™ and XIP Libraries™ are freely available at www.openxip.org.)

3. Image Annotation Technical Overview

IA is an image reading tool, designed for controlled experiments in image annotation and markup as shown in Figure 1. It receives a DICOM series as input, displays it in a double-oblique, 3D MPR (multiplanar reconstruction) viewer, receives annotations and markup of the image, and stores them in the AIM/XML format. For markup, the user creates a seed stroke on the image, then invokes a segmentation algorithm to give a rough first draft of the tumor boundary. The user then edits the contour on each slice of the tumor by drawing closed contours that are added to or subtracted from the existing contour. In addition, the user draws RECIST and WHO diameters on the tumor, the latter drawn with a special widget that keeps the two diameters orthogonal. In addition, the user has the opportunity to record the shape or margin characteristics of the tumor, his confidence that the marked tissue is in fact a tumor, and a free-text comment. The shape, margin characteristics and confidence are restricted to RadLex terms chosen from pull-down lists. When the user hits the Save button, IA creates four separate AIM objects for the seed, volume, RECIST, and WHO observations, so that they can be queried separately in the database for use in AE and MVT.

IA can also load previously-created AIMs (Volume, RECIST, and WHO) for review.

IA contains an example segmentation algorithm, constructed from open-source components drawn from the ITK library. This algorithm is not intended to represent the state of the art, but only a possible starting point for building better segmentation algorithms.

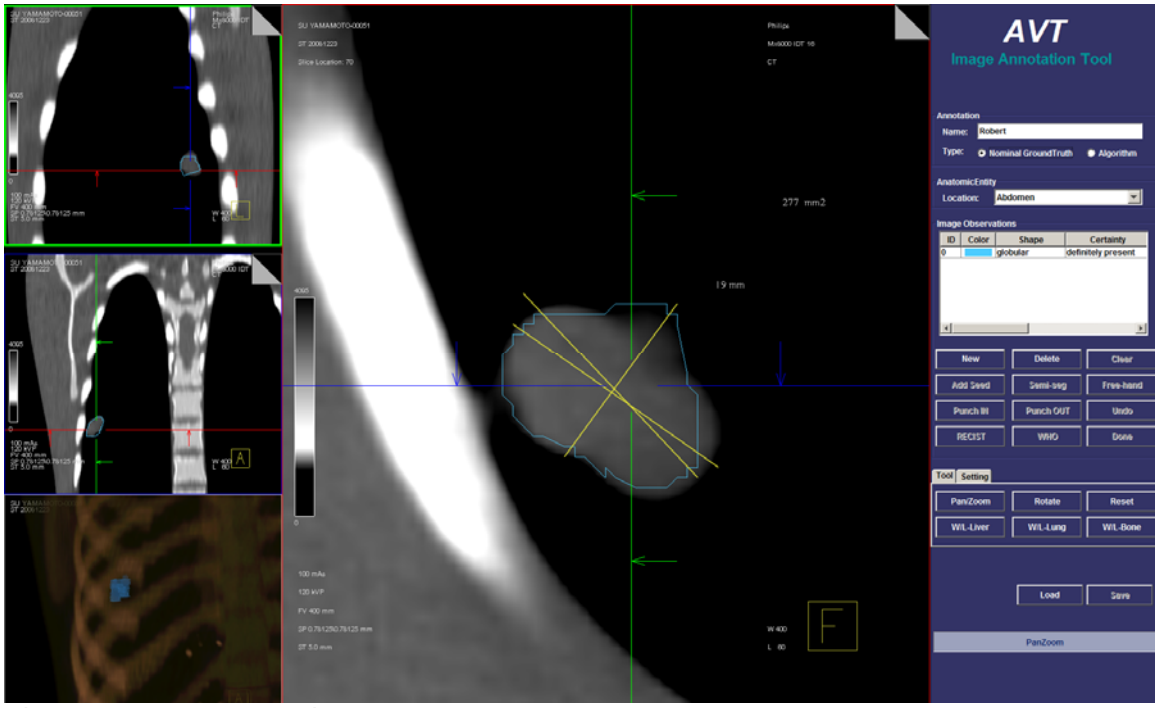

Figure 1 Image Annotation UI

4. Algorithm Execution (AE) Technical Overview

AE receives a batch of Tumor Seed Annotations as inputs, retrieves the referenced DICOM Series from AD, runs a preconfigured segmentation algorithm on each of the seeds shown in Figure 2, generates a sample slice image showing the contour on that slice shown in Figure 3, and produces Tumor Volume Annotation objects.

The interface between AE and the segmentation algorithm is an XIP scene graph. Typically, a new algorithm would be implemented using the XIP Builder™ and XIP Libraries™, wrapped in a scene graph, then plugged into AE.

Figure 2 Algorithm Execution UI

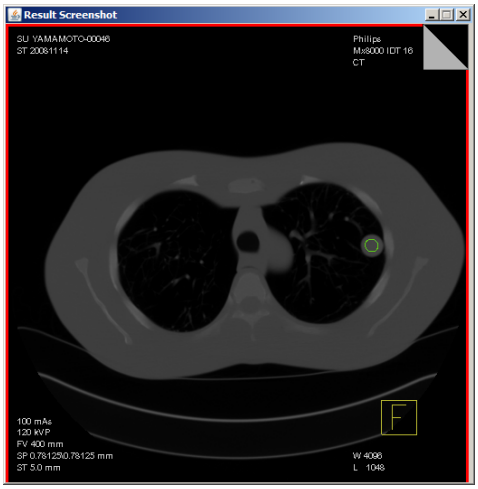

Figure 3 Screenshot of algorithm execution result

5. Measurement Variability Tool (MVT) Technical Overview

MVT receives a set of AIM annotations as input, calculates a selected set of measurements on them, then performs the following kinds of analysis on them shown in Figure 4:

- Summary statistics: mean, standard deviation, or a custom R script
- Multiple regression, ANOVA
- Outlier identification
- Plotting (Scatter, Bland-Altmann, Histogram)
- Drill-down Visualization

Each of these analysis methods is available with the data organized in each of three ways:

- Performance / Sources of Variation: each experimental reading is compared to a nominal ground truth reading of the same tumor.
- Inter-reader Variation: each experimental reading is compared to each other experimental reading, by a different reader, of the same tumor.
- Intra-reader Variation: each experimental reading is compared to each other experimental reading, by the same reader, of the same tumor.

For drill-down visualization, double-clicking on an entry in the results table causes MVT to display the relevant tumor in a 3D MPR viewer, showing the two compared segmentations of the tumor in different colors.

Results of MVT analysis can be exported (for plots) as graphics files (.jpg, .bmp, .png). (Export of the tables as spreadsheets would be a straightforward addition to AVT, but has not yet been implemented.)

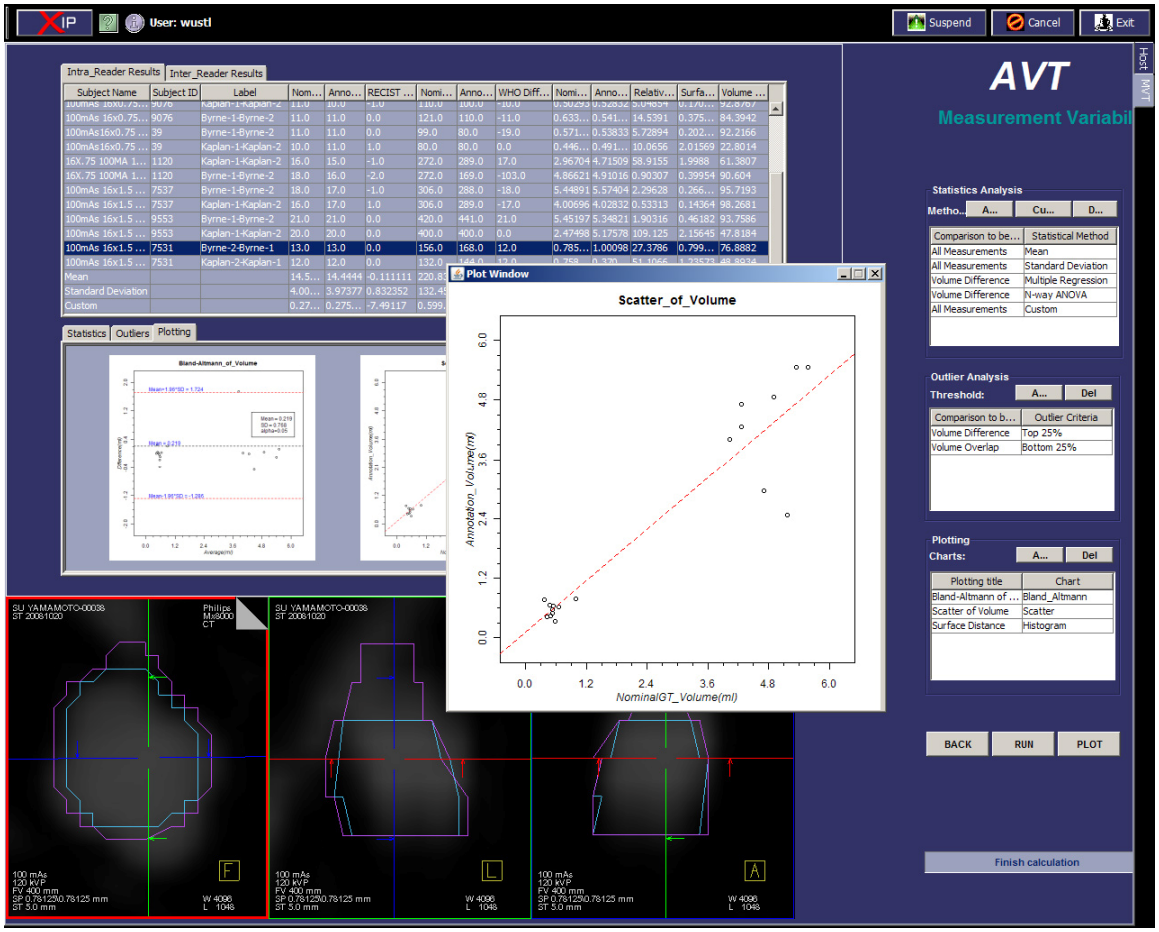

Figure 4 Measurement Variability Tool UI

6. Assessment Database (AD) Technical Overview

The Assessment Database (AD) provides integrated storage and retrieval of DICOM images and AIM annotations. The annotations and other meta-data describing the images are stored and indexed in a relational database for fast access. The DICOM files are stored in the file system to reduce database size and allow transparent access.

The AD includes an audit trail that tracks creation and modification of AIM objects in a database table. DICOM segmentation objects are only stored in the database as a logical part of an AIM object and never modified, so they do not merit separate log entries.

7. XIP Host™ for AVT: Technical Overview

XIP Host™ has been adapted for AVT by adding a tab card that queries AD for DICOM and or AIM objects, allows the user to browse and select portions of the query results for retrieval, and passes the retrieved results to the core XIP Host™for use in dispatching an AVT application, or any other application that conforms to the DICOM Application Hosting standard.

The purpose of using XIP Host™was to let the applications ignore the sources and sinks of data, letting XIP Host™take care of such “plumbing”. As a result, all three applications can be run with data coming from AD, the local file system, or a caGrid data service. (The TCGA version of AVT

was also able to write AIM/XML objects to the AIME data service at Emory University.) In theory, the AVT applications should also run on any Hosting System that conforms to the DICOM Application Hosting standard; however, we have not tested nor do we currently provide support for that capability.

The tab card adds the AIM section to the usual DICOM query fields. The query is interpreted as, "Give me all the DICOM series matching the DICOM criteria AND that are referred to by AIM objects that match the AIM criteria." The (de-identified!) Patients are then listed in the right-side tree browser, where the user can interactively expand parts of the tree to find the individual DICOM and AIM objects matching the query. The user then selects the Series of interest, and indicates whether he wants the application to receive the Series, the AIM and Segmentation objects, or both, as shown in Figure 5.

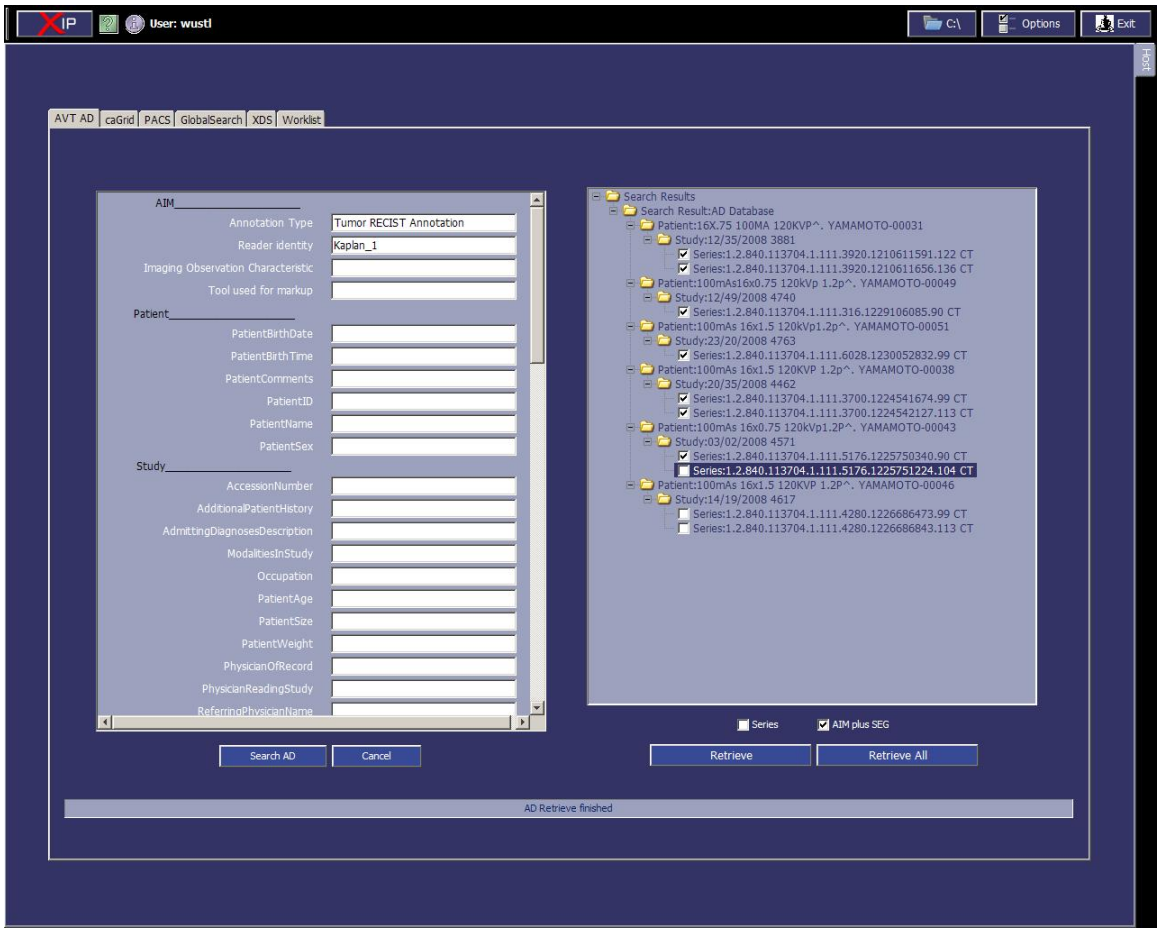

Figure 5 XIP Host for AVT UI

Appendix: Audit Trail Technical Overview

The AVT Audit Trail capability is not a software component in the normal sense. Rather, it is an infrastructure capability implemented by a service of the Annotation Database component and a collection of features of other components that use that service.

AVT2EXT creates a log entry each time an AIM object is added to the database. , containing the AIM object id, the user's login name, a timestamp, and a comment.

- IA and AE create AIM objects and pass them to XIP Host™ for storage. (MVT does not produce any AIM objects.)
- XIP Host™ passes the AIM objects to the local Annotation Database for storage.

Additional details that sometimes appear in audit trails appear in the AIM objects themselves, when the objects are created by AVT, including:

- URLs of the source DICOM images that the AIM object describes
- URLs of any segmentation objects created.
- Measurements made
- Name and version of the AVT software used
- Name and version of the segmentation algorithm used
- Subjective uncertainty of the observations
- Free-text comment containing other pedigree information (expert commentary)

The audit trail is stored in the database as a separate table. It can be examined by running SQL queries on the database, such as those listed in the Administration Guide.