3D Slicer
for clinical use,
for radiotherapy research,
and for your research work

Andras Lasso
Laboratory for Percutaneous Surgery, School of Computing
Queen’s University, Kingston, ON
Canada
3D Slicer for clinical use

software application for MRI-guided prostate interventions
MRI-guided prostate biopsy

Multiple supported devices
- Transrectal robot-assisted (TRR)
- Transperineal template (TPT)
- Transperineal robot-assisted (TPR)

Multiple clinical sites
- NIH-NCI (Bethesda, MD): TRR
- JHH (Baltimore, MD): TRR
- BWH (Boston, MA): TPT, TPR
- PMH (Toronto, ON): image sharing only

Krieger et al. 2005
Song et al. 2010
Tokuda et al. 2011
Image-guided biopsy workflow steps

- Pre-procedural planning
- Calibration
- Planning
- Targeting
- Verification
Planning

- Register/show available images
- Mark point targets
Targeting

- Simplified display on procedure-room monitor
- Robot, scanner control
Verification

- Verify patient, robot, and needle position
- Using automatic image registration
3D Slicer clinical use – summary

• Successful example: same software, multiple devices, multiple sites
• Use existing features in 3D Slicer
• Customization
  – Software development: algorithms and graphical user interface
  – Quality assurance process: testing, change control, releases, issue tracking
3D Slicer for radiation therapy research
Active projects

• Adaptive radiotherapy for head and neck cancer
  – Funded by NA-MIC, 2010-2013
  – PI: Greg Sharp (MGH, Boston, MA)
  – 4 researchers, software engineers

• SparKit: toolkit and platform for radiotherapy
  – Funded by Cancer Care Ontario, 2011-2016
  – PI: G. Fichtinger (Queen’s University, Kingston, ON)
    Co-investigator: Terry Peters (Robarts Institute, London, ON)
    Project leader: Andras Lasso (Queen’s University, Kingston, ON)
  – 6-8 software engineers and infrastructure

• NA-MIC collaborations in preparation
SparKit: Software Platform and Adaptive Radiotherapy Kit

Software Platform (SP): shared, reusable, and customizable basic software components for radiotherapy

Adaptive Radiotherapy Kit (ARKit): Specific toolkit for adaptive radiation therapy and associated image-guided interventions

Goals:

• Validate clinical hypotheses in clinical trials
• Ready-to-use image analysis and visualization capabilities => avoid re-development
• Quickly deployable systems => minimize system engineering effort
SparKit tools
Based on 3D Slicer and the NA-MIC kit
SparKit infrastructure

- **Development**
  - Develop software application
  - Test & optimize
  - Fix errors
  - Deliver software releases
  - Use data/images

- **System Engineers** (SparKit personnel)
  - Download software
  - Report errors
  - Upload data/images

- **Research Scientists**
  - Develop computing algorithms
  - Use test data/images

- **Clinical Users**
  - Download software
  - Report errors
  - Upload data/images

- **Dashboard** (CDash)
  - Issue tracking
  - Source code control
  - Messages
  - Documentation
  - Files, Releases

- **Project 1**
  - Documentation
  - Source code control
  - Issue tracking
  - Messages
  - Files, Releases

- **Project 2**
  - Documentation
  - Source code control
  - Issue tracking
  - Messages
  - Files, Releases

- **Project …**
  - Documentation
  - Source code control
  - Issue tracking
  - Messages
  - Files, Releases

- **Automatic build & test machines** (CMake/CTest)

- **(Assembla + image database)**
Project scope (tentative)

• DICOM-RT support in 3D Slicer: import/export structure sets and dose maps
• Visualization: dose volume histogram, isodose lines
• Better support for temporally changing data (2D+t, 3D+t)
• 3D Slicer performance optimization
• Image and protocol data sharing infrastructure
• ... still collecting inputs from the community
Current SparKit activities

• Set up team & infrastructure
  – www.assembla.com/spaces/sparkit
  – Software engineers hiring

• Identify needs
  – Survey, meetings

• Set up collaborations
3D Slicer for your own problem

Programming 3D Slicer
Main concepts

• All information is stored in MRML (Medical Reality Modeling Language) nodes
  – Node types: images, models, transforms, fiducial lists, etc.
  – Observer pattern: MRML nodes notify their observers of any state changes

• Extension/customization via plugin modules
  – Define new nodes, observe existing MRML nodes
  – ITK, VTK, Teem, Curl, OpenIGTLink, QT already available
Programming 3D Slicer

• Command-line module: .exe file (with specific command-line parameters)
  – simple, executable without Slicer
  – no access to Slicer internals, Slicer compilation needed

• Scripted module: Python or TCL scripts
  – simple, no compilation needed
  – limited access to Slicer internals

• Loadable (interactive) module: .dll (with specific Slicer API interface)
  – full access to Slicer internals
  – Slicer compilation needed, requires Slicer core knowledge
Getting started

• Download: http://www.slicer.org/pages/Special:SlicerDownloads

• Latest stable version (recommended)
  – Type of download: Stable Releases
  – File to download: latest date

• Documentation, examples, step-by-step tutorials, etc:
  http://www.slicer.org/

• 3D Slicer training courses, developer meetings:
  http://www.na-mic.org/Wiki/index.php/Events

• Slicer4 (faster, nicer, ...) is expected to be released for RSNA 2011
Thank you.

Andras Lasso
lasso@cs.queensu.ca
Laboratory for Percutaneous Surgery
School of Computing
Queen’s University, Kingston, ON
http://perk.cs.queensu.ca

http://www.assembla.com/spaces/sparkit