Research Interfaces for Image Guided Neurosurgery: From VVLink to OpenIGTLink

Xenios Papademetris
Departments of Diagnostic Radiology and Biomedical Engineering
Yale University

A Non-Disclaimer

- While there was extensive scientific collaboration between the Yale and BrainLAB teams during the development of the VVLink interface, it must be stressed that the collaboration was purely scientific.
- None of the Yale researchers received (or have rights to) any financial benefit from the sales of this BrainLAB project.
- Neither, have they received any compensation from BrainLAB AG during this time.
- Further, none of the research work performed by Yale researchers was supported by BrainLAB AG during this time.
Talk Outline

- Image Guided Navigation Systems
- Interfacing to Commercial Image Guided Navigation Systems
- Application to Epilepsy Neurosurgery
  - OR Examples
  - Visualization Work
- Open IGT Link work
  - An example and lots of thoughts/suggestions

Image Guided Navigation Systems

BrainLAB, Neurosurgery 2002
Procedure

- Structural and Functional Imaging
- Surgical Planning
- Patient to Image Registration
- Track the position of the tools and display their position over the images.
  - Real Time Navigation.

Putting it all together ...
BrainLAB VVCranial System

Research + Surgery = Trouble

- Image Analysis Research
  - Flexibility
  - Adaptability
  - Constant Change

- Image Guided Neurosurgery Platforms
  - Stability
  - Reliability

How can we accomplish both?
Bringing Research Algorithms into the Operating Room

- Commercial Systems are not extensible – regulatory limitations

Custom Image Guided Navigation System

VS

Commercial Image Guided Navigation System

Custom Image Navigation Systems

- Has been done MIT/Harvard, Vanderbilt, IGSTK etc.
- Particularly useful in introducing new forms of Image Guided Surgery

BUT
- Commercial Neurosurgery IGS widely installed
- Overly complex when primary interest is image analysis research and NOT surgery
The Case for Research Interfaces

- Allows Surgeons to use familiar interface
- Researchers can focus on task at hand, not the clinical aspects of surgery
- Takes Vendor/FDA out of the research loop
- Stability – a program is as stable as its least stable part
- Licensing/ Programming Language/OS neutrality

The VVLink Story

- Original Design June 2002
- First restricted release (July 2005)
- Commercial Availability ~ Fall 2005
Research Interface Design

BrainLAB VV Cranial System

Network Connection

Research Workstation
Yale BioImage Suite

What can it do

Labeled Points

Research Workstation

Real Time (Multiple) Tool Coordinates

BrainLAB VV Cranial System

Image Slice-sets

Streaming visualizations
Initialization Procedure

1. Enable VVLink on BrainLAB System
2. Connect from client system
3. Accept connection on BrainLAB System
4. Transfer one image over to establish coordinate system mapping

(See BioImage Suite manual for details www.bioimagesuite.org/doc)
A Little Code

VVLConnection *VVLinkServer = VVLConnection::New();

VVLinkServer->ConnectTo("192.168.1.2","password");

// now, perform the tasks
// ...
// disconnect & clean up
VVLinkServer->Disconnect();
VVLinkServer->Delete();
Step 2 : Transfer Image

```cpp
const vvlink::VVLSliceSet *sliceSet = VVLSliceSets->GetSliceSet( imagename );
vtkMatrix4x4* matrix = sliceSet->GetOrthogonalTransformation();
vtkImageData* img = sliceSet->GetOrthogonalVolume();
vtkLookupTable* lut = sliceSet->GetLookupTable();
```

Transfering an Image
Initialization Completed

Navigation
Real Time Tool Coordinates

```cpp
tkObjectContainer<vvlink::VVLTrackingData> data( vvlink::VVLTrackingData::New() );
VVLNTrackingData->GetTrackingData( data );

for( int i=0; i<data->GetNumberOfInstruments(); i++ ) {
    const vvlink::VVLInstrument *instr = data->GetInstrument(i);
    fprintf(stdout,"Instrument: %s\n", instr->GetName().c_str());

    fprintf(stdout,"Tip = (%.3f,%.3f,%.3f)\t",
            instr->GetTip()[0],
            instr->GetTip()[1],
            instr->GetTip()[2]);
}
```

Performance Evaluation Summary

- Dual system interface is essentially as fast as single system
  - Limiting factors have to do with tool tracking hardware not IGS

- Data transfer speed is essentially as fast as native operating system “file copy”
  - Limiting factor is underlying network speed

- System stability demonstrated by random network disconnections
In Action I

In Action II
Non-Orthogonal Volume Cropping

- Planar Cropping can be problematic as it removes “too much” i.e. half a brain
- Often removes the context around which navigation is taking place
- In recent work we have explored non-orthogonal cropping techniques for visualization (Joshi et al IEEE TVCG in-press)
  - Spherical, Cubical, Cylindrical probes
  - Highlighting of “inside” surface
  - Adaptive Cropping (e.g. leave function but remove anatomy)

Setup VVLink
Highlighting “Inside” Surface

Intracranial Electrode Visualization
Access to Image Guided Navigation Systems is key to testing image analysis algorithms in the Operating Room.

Developed such an interface -- in collaboration with BrainLAB.

Medtronic has also moved in the same direction (STEALTHLink).

Eliminates the need for custom IGS design/testing and implementation if image analysis is the driving application.
Open IGT Link

- OpenIGT Link is more recent open-source attempt to create a generic network interface for image-guided therapy

- Current Status:
  - Can transfer images and tool positions (as 4x4 matrices)

- Initial Testing: Create a bridge (proxy) server to map VVLink to OpenIGT Link

---

A Schematic of Experimental Setup

1. Images
2. Tool Position
Alternatively save it for end of talk …. (5 more slides to go)

While VVLink is closed Source and hence code is for now inaccessible there is lots we can learn from this.

The VVLink experience has proven that
- Network interfaces of this form can be extremely effective, stable and reliable
- Minimize modifications to existing systems
- Allow for integration of systems based on different platforms, licenses, programming languages etc
Replacing Plugins as an integration mechanism

- From Plugins to “Application Groups”

- Plugins can be problematic as a strategy
  - Interface is often complex, need to understand API
  - Platform/Compiler restrictions
  - Often software license issues.

- Users are stubborn and like to control the main loop!

- Network interfacing can offer a way around all of these issues, as well as leverage multiple computers at once!

- Sun’s old motto “The Network is the Computer”

Application Groups

- Create a set of tools that one thing and one thing only
  - Examples
    - VVLink Bridge Server
    - Image Acquisition Controller
    - DICOM Server
    - Image Registration Server
      - Take two images and a set of parameters and return a transformation
  - Master Application connects/discovers all of these and invokes them for specific processing
    - Without modifying or needing to compile any of these!

- OpenIGTLink or maybe OpenIGTLink++ can be the glue that holds all of these together.
Some Suggestions

- Improve the documentation – this is critical
- Begin process of wrapping custom interfaces to create OpenIGTLink proxy servers
- Add an OpenIGTLink interface to VTK to make its application easier
  - Reference Server and Client Implementations
    - The current Biolmage Suite implementation (3 C++ classes) can be used as a starting point.

Suggestions II

- Extent OpenIGTLink to support more data types
  - Polygonal surfaces
  - Proper Image Headers (e.g. NIFTI-based)
  - More complex transformations (possibly displacement fields shipped as images)
  - Labeled landmarks
  - More as research needs direct ….
  - …..

- Extent OpenIGTLink to handle some queries
  - This will enable tool identification from master application
  - Perhaps for some tools a standard query would yield enough information to allow the automatic creation of a GUI to control the external application
Suggestions III

- Please keep interface simple and the learning curve relatively easy.

- Avoid unnecessary complexity in reference libraries (not everything needs to use templates and STL and …)
  - Design with the user, not the programmer, in mind.

- Make it all accessible from scripting languages cleanly.

- Maybe think of a Matlab implementation.

Acknowledgments

**Image Analysis**
- James Duncan, Ph. D.
- Lawrence Staib, Ph. D.
- Christine DeLorenzo, Ph. D.
- Alark Joshi, Ph.D.

**Biolmage Suite Team**
- Hirohito Okuda
- Marcello DiStasio
- Thomas Teisseyre
- Dustin Scheinost

**MRI/MRS Imaging**
- Todd Constable, Ph. D.
- Douglas Rothman, Ph. D.
- Robin de Graaf, Ph.D.
- Hoby Hetherington, Ph.D.
- Jullie Pan, M.D. Ph.D.

**Neurosurgery**
- Dennis Spencer, M.D.
- Kenneth Vives, M.D.

**Neurology**
- Susan Spencer, M.D.
- Edward Novotny, M.D.
- Robert Duckrow, M.D.
- Hal Blumenfeld, M.D.
- Hitten Zaveri, Ph. D.

**BrainLAB, AG Munich**
- Markus Neff
- Sven Flossman
- Robert Schmidt
- Nils Frielinghaus
Acknowledgments

- R01EB000473 (Duncan, J. PI)
- R01EB006494 (Papademetris, X. PI)
- R21EB007770 (Papademetris, X. PI)

Available at: www.bioimagesuite.org

NITRC
The Source for Neuroimaging Tools and Resources
nitrc.nih.gov

Available at: www.bioimagesuite.org