A generic module interface in MeVisLab using OpenIGTLink for defining real-time communication workflows among robotic devices

S. Kraß¹, L. Chen², J. Tokuda², T. Pätz³,⁴, R. Kikinis¹,²,³

¹ University of Bremen, Medical Image Computing, Bremen, Germany
² Brigham and Women’s Hospital and Harvard Medical School, Department of Radiology, Boston, USA
³ Fraunhofer MEVIS, Bremen, Germany
⁴ Jacobs University, Bremen, Germany

Contact: Scheherazade.krass@uni-bremen.de

Abstract:

The goal of this work is to provide a software platform that can seamlessly plug-in and plug-out software and hardware components to prototype robot navigation system with interactive path and motion planning capability. We designed and implemented a plug-in module in MeVisLab that exchanges data with external hardware device(s) over the TCP/IP network using the OpenIGTLink network protocol. This interface allows developers to define a desired workflow and dataflow in sending / receiving a wide variety of messages including commands and feedbacks to / from external hardware using MeVisLab’s graphical programming environment. This information can include real-time communication among robotic devices, intraoperative imaging systems like ultrasound devices and sensor elements. This module interface can easily be used in different research and prototyping works.

Keywords: MeVisLab, OpenIGTLink, robotic devices communication, image guided therapy, surgical navigation

Background

Robotic navigation systems are growing rapidly and finding their way more and more in the clinical routines supporting minimally invasive surgery. In particular, a number of robotic navigation systems have been developed for precise placement of interventional tools, such as needles and catheters. One of the challenges in precise placement of interventional tools is that the robot has to deal with the in-vivo tissues, which moves significantly with organ movements and tissue pressure. One possible solution is to perform the path planning and motion planning in an interactive fashion; capture images to estimate the deviation from the planned target, and correct it dynamically. To develop a robotic navigation system that capable of interactive motion planning, a versatile prototyping environment that supports image processing and visualization in medical imaging, as well as closed-loop communication with the robotic device is crucial.

The goal of this work is to provide a software platform that can seamlessly plug-in and plug-out software and hardware components to prototype robot navigation system with interactive path and motion planning capability. Specifically, we implemented a network communication interface module to a generic framework for medical image processing research and development, namely MeVisLab. For the network communication, we used OpenIGTLink, which has been used for transferring data required for surgical navigation, such as images, transforms, and commands. The new interface module makes all capabilities of OpenIGTLink for robotic devices accessible from each component in MeVisLab. In this module the user can use the OpenIGTLink in modular level, without having to go to the details. It is especially of interest for defining a workflow for sending commands, receiving target information, displaying results, error estimation and so on. We aim to use this interface for our further goals in interventional motion planning research.

Material and Methods

The architecture proposed in this paper is inspired by a previous paper by Egger et al. [2]. They developed a third party library in MeVisLab for integrating the OpenIGTLink interface in MeVisLab for data and image transmission and tested with NDI tracking system. MeVisLab is a generic framework for medical image processing research and development. It contains most advanced image processing algorithms and offers visualization modules in 2D and 3D. The developers are
able to make rapid clinical prototypes for different goals using the modular framework of MeVisLab. As a network communication protocol, we chose OpenIGTLink, which is specifically designed for communications between hardware devices in the operating room. OpenIGTLink can be used to transfer data required in the image-guided therapy (IGT) setting, e.g. tracking and image data, device control/monitoring information, among the devices and software [1]. The advantage of using OpenIGTLink is that it has been used in a wide variety of research projects, and could provide access to a wide variety of hardware devices already; a popular open-source medical image computing software, 3D Slicer uses OpenIGTLink as a primary network communication interface; Tokuda et al used OpenIGTLink protocol for MRI-guided robotic transperineal prostate therapy [3]; Tauscher et al. developed an interface for real-time state control of robot and visualization purposes based on OpenIGTLink, and implemented and tested with Kuka LWR Robot [4].

We have designed a new generic interface module for simplified using OpenIGTLink in MeVisLab as a plug-in module. The new module is being implemented in C++ programming language. Fig. 1 shows the architecture and dataflow of a typical robot navigation system. This module serves as an interface for using OpenIGTLink, and provides the following features:

- define client/server devices operating in an own thread.
- define a set of commands in the given time stamp to send to the robot: Start, Stop, Idle, Pose.
- send/receive strings, image sets and control commands
- receive and show robot status, calibration result
- receive real-time feedback, e.g. end-effector pose in real world coordinates, images etc.
- show desired coordinates as planned in pre-operative imaging.
- calculate the errors in desired time segment

We have designed a new generic interface module for simplified using OpenIGTLink in MeVisLab as a plug-in module. The new module is being implemented in C++ programming language. Fig. 1 shows the architecture and dataflow of a typical robot navigation system. This module serves as an interface for using OpenIGTLink, and provides the following features:

- define client/server devices operating in an own thread.
- define a set of commands in the given time stamp to send to the robot: Start, Stop, Idle, Pose.
- send/receive strings, image sets and control commands
- receive and show robot status, calibration result
- receive real-time feedback, e.g. end-effector pose in real world coordinates, images etc.
- show desired coordinates as planned in pre-operative imaging.
- calculate the errors in desired time segment

Results & Discussion

The plug-in module is currently being implemented for further testing in a wide variety of robotic devices and navigation systems including Interventional Robotic system, Casination navigation system, and Kuka LWR. The new module will allow developers to design a robot navigation system by composing a block diagram on the MeVisLab’s graphical user interface interactively, and then test workflow and data flow with actual hardware device. This work is used for other navigation projects, which require real-time path correction via feedback controlling. The path correction has two different approaches, it can be calculated statically, i.e. based on initial target planning on preoperative imaging or dynamically, i.e. intraoperative feedback position controlling, such as ultrasound or sensory feedback. The plug-in would also
be extended to communicate with the other medical image computing software platforms or applications, which have already been adapted to OpenIGTLink, including 3D slicer, PLUS [5], CustusX [6], and IBIS [7].

Summary

The researchers and clinicians can use this simplified interface for interventional robotic devices inside MeVisLab.

Acknowledgement

This work is supported in part by the US National Institutes of Health R01EB020667 and P41 EB015902.

References


