Open-source platforms for navigated image-guided interventions

Tamas Ungi, MD, PhD, Andras Lasso, PhD, Gabor Fichtinger, PhD

Laboratory for Percutaneous Surgery, School of Computing, Queen's University, Kingston, ON, Canada

To appear in Medical Image Analysis. Accepted on June 13, 2016.

Abstract

Navigation technology is changing the clinical standards in medical interventions by making existing procedures more accurate, and new procedures possible. Navigation is based on preoperative or intraoperative imaging combined with 3-dimensional position tracking of interventional tools registered to the images. Research of navigation technology in medical interventions requires significant engineering efforts. The difficulty of developing such complex systems has been limiting the clinical translation of new methods and ideas. A key to the future success of this field is to provide researchers with platforms that allow rapid implementation of applications with minimal resources spent on reimplementing existing system features. A number of platforms have been already developed that can share data in real time through standard interfaces. Complete navigation systems can be built using these platforms using a layered software architecture. In this paper, we review the most popular platforms, and show an effective way to take advantage of them through an example surgical navigation application.

1. The “slow revolution” of navigated interventions

Over the past two decades, imaging technology has changed the landscape of medical interventions. Procedures guided by imaging became minimally invasive and more accurate. In fact, many interventions are only possible with image guidance, such as biopsy of small tumors, blocking of deeper nerves for regional anesthesia, or catheterization of small blood vessels. Most procedures require only preoperative images that the operator uses to visually estimate the targets relative to surface markers. This helps in determining the entry point and initial orientation of the interventional tool, but provides no further guidance or assistance during the intervention. Imaging during the procedure is also possible, but none of the available options are ideal. Regular use of CT and X-ray imaging comes with radiation to the patient and the operator. Interventional MRI systems are scarcely available due to their high cost. Ultrasound can only be used in regions of the body where acoustic properties allow visualization.

Position tracking is revolutionizing image-guided interventions by providing real-time feedback without intraoperative imaging. When tracking is registered with preoperative images, tools can be visualized relative to internal organs. When image-guidance is combined with real-time position tracking, we call these procedures navigated interventions. Enabling devices for navigation such as optical and electromagnetic trackers have been available on the market for over two decades, and spatial navigation has been implemented in some commercial systems. Navigation has become the standard in a few orthopedic and neurosurgery procedures, and it is an available option in a number of other interventions as well.

But the revolution of navigated interventions is happening slower than most researchers had initially imagined. The clinical acceptance of navigation is hindered by limited, single-center studies on phantom models or small clinical samples. While many clinical studies conclude with positive results, others often shy away from trying to reproduce the results, due to major costs and efforts needed to re-implement the navigation system. Thus many projects remain a “one-off”, inevitably reporting longer procedure times and higher costs compared to conventional methods and failing to convincingly show benefit to patients; slowing down and often preventing clinical translation of new intervention techniques. Arguably the greatest obstacle to clinical translation of experimental navigated interventional systems is their technological complexity. Putting together even the smallest and simplest navigated intervention system demands a great deal of mundane engineering effort that has no scientific value. Moreover, the level of robustness and safety required in clinical systems demands high-quality engineering that may exceed the capacity of many research labs.

We believe that the burden of system engineering can be taken off of researchers by using shared, open-source application platforms. Many common features can be implemented in clinical translation platforms, while allowing full customization of the user interface for procedure-specific applications. If new methods are deployed as modules of a platform, results from different research groups will be more comparable to each other, this may resolve controversial findings and facilitate multi-center clinical trials. Platforms are also a great way to share implementation details, which plays an important role in the outcomes of clinical evaluation studies, but often do not fit in traditional journal publications.

The field of navigated interventions is rapidly evolving. The pursuit of new and exciting technology often takes the focus away from reproducibility. Some groups create their own navigation systems and have fast initial results, but isolated efforts can easily fail to make a real change in clinical practice. Today, clinical decisions are based on the principles of evidence-based medicine. Stronger evidence is associated with reproducible results by independent research groups. But multi-center studies and clinical acceptance is delayed when new systems are built around every new method. These systems can get overwhelming for new users to learn. Also, valuable resources are wasted on implementing already existing software features or hardware components.

Among all the advantages of common application platforms, probably the greatest advantage is that they make it easier to integrate different methods in one system. Interventional systems reach their full potential when the navigated tools are not only shown in the context of a preoperative or intraoperative images, but when these images are processed to highlight important information and suppress noise. Segmentation of tumors or anatomical structures, as well as registration of multiple images make a navigation system complete for clinical users. These image processing algorithms can be challenging to integrate into clinical systems, unless the algorithms are already implemented in the same platform, or the platform provides examples and standards for integration.

2. Pursuit of a common platform

Researchers recognized the need for software platforms in image-guided and navigated interventions, and several significant efforts have been made towards such a platform in the past decade. Some platforms were successfully adopted by a number of other research groups. The most popular platforms are all made available as open-source projects with a license that allows academic and commercial use without restrictions. Sharing of the source code increases the quality of the software by allowing more people to discover and fix problems, and users can also enhance or customize the platform without relying on the original developers.

The most widely accepted and shared component of computer-assisted interventional systems is a network communication protocol, called OpenIGTLink (www.openigtlink.org). This relatively simple network protocol defines data packages for all kinds of devices used in medical interventions (Tokuda et al., 2009). It is used in all major research applications, and also supported by a few commercial device manufacturers. Such devices can be accessed in real time from research applications. OpenIGTLink has proven to be crucial in the compatibility between different software platforms as well, and it has had a tremendous beneficial effect on the evolution of both image-guided and robotic medical interventions research.

The Image-guided Surgery Toolkit (IGSTK) was the first common software platform for developers of image-guided interventions (www.igstk.org) (Enquobahrie et al., 2008). While IGSTK was carefully designed to enforce good programming practices and robust applications, considerable programming experience is necessary to build applications using IGSTK, or to further evolve the toolkit itself. IGSTK was also one of the first popular software in the medical imaging domain to implement an abstraction layer over different position tracker devices. Hardware abstraction saves significant time when different tracker devices are used with the same application.

The PLUS toolkit (www.plustoolkit.org) implements communication to probably the widest variety of commercial devices used in medical interventions. It can communicate with image scanners, cameras, and position trackers, but also a whole array of other devices including microcontrollers for custom device prototyping including robots (Lasso et al., 2014). PLUS is unique among major platforms in that it can simultaneously communicate with any number of devices, and synchronize, save, or relay data streams to other applications in real time through OpenIGTLink. This feature saves software development time when experimenting with systems, as additional devices can be added without changes in existing configurations.

Some platforms are built as extensions of open-source medical image analysis application platforms. The advantage of this approach is that image-guided interventions can readily take advantage of visualization, data import/export, and image processing algorithms. Such an open-source image analysis application is the Medical Imaging Interaction Toolkit (MITK). An extension of MITK for image-guided therapy is the MITK-IGT (http://mitk.org/wiki/IGT), which includes hardware communication interfaces, real-time data processing, and visualization features. Another extension of MITK is NiftyIGI that can be customized for image-guided interventions and delegates communication with hardware devices to other applications through OpenIGTLink (Claskson et al., 2015).

Currently the most popular research application in medical imaging is 3D Slicer (www.slicer.org). 3D Slicer offers extensions though a web-based app store to facilitate distribution of extensions (Fedorov et al., 2012). 3D Slicer can be downloaded and installed for all major operating systems. Besides stable releases, a nightly version is compiled and uploaded to the 3D Slicer website every day, so people without software developer skills can always benefit from the latest features. Image-guided and navigated interventions are supported by an extension called SlicerIGT. This extension contains a collection of modules for commonly needed tasks in interventions, and also provides features for software developers to create fully customized graphical user interfaces for intraoperative use. SlicerIGT does not directly communicate with hardware devices. It uses the OpenIGTLink communication module of 3D Slicer, therefore it requires devices with OpenIGTLink support, or PLUS to relay the information between 3D Slicer and the hardware.

Another application platform that uses PLUS as an abstraction layer between applications and hardware devices is called CustusX (Askeland et al., 2016). CustusX provides a user interface template. Applications built on this template have familiar features for their users, therefore they are easier to learn and use in new kinds of procedures (http://www.custusx.org/).

Despite all the efforts towards a common software platform, most research teams and institutions can only continue momentum if they keep their existing applications. A complete and immediate switch to a common platform is not a realistic expectation. Therefore, some of the most prominent researchers in the field of medical imaging joined forces in an open-source project called CTK (Common Toolkit). CTK is a software library that implements shared components that were not covered in previously existing shared toolkits. Currently, its scope is DICOM communications, graphical user interface components, and plugin mechanisms to support sharable software modules (http://commontk.org). CTK has been an important step towards harmonization of software components. It is already used in many popular applications, such as 3D Slicer, MITK, MeVisLab, and medInria.

3. System design case study

In this section, we describe general design principles for rapid development of interventional navigation systems. We demonstrate the principles through an example, a navigation system for breast cancer surgery (Ungi et al., 2016). The software application in this example is called LumpNav. All source code of the example system is publicly available (www.SlicerIGT.org). The LumpNav example shows that application source code is extremely small in size compared to the code for reusable features in underlying system components. The application code is intentionally kept minimal to reduce the effort on development of further applications or modifying existing ones.

Research systems for medical interventions need to be designed around existing devices that are approved for clinical use. System designs separated in three layers above the hardware layer have distinctive advantage in modularity (Figure 1). If these layers are not clearly separated, the resulting system will have limited hardware compatibility and it will be difficult to reuse software components. Although system designs often have further sub-layers, from a system engineering point of view the most important ones are hardware abstraction, application platform, application.

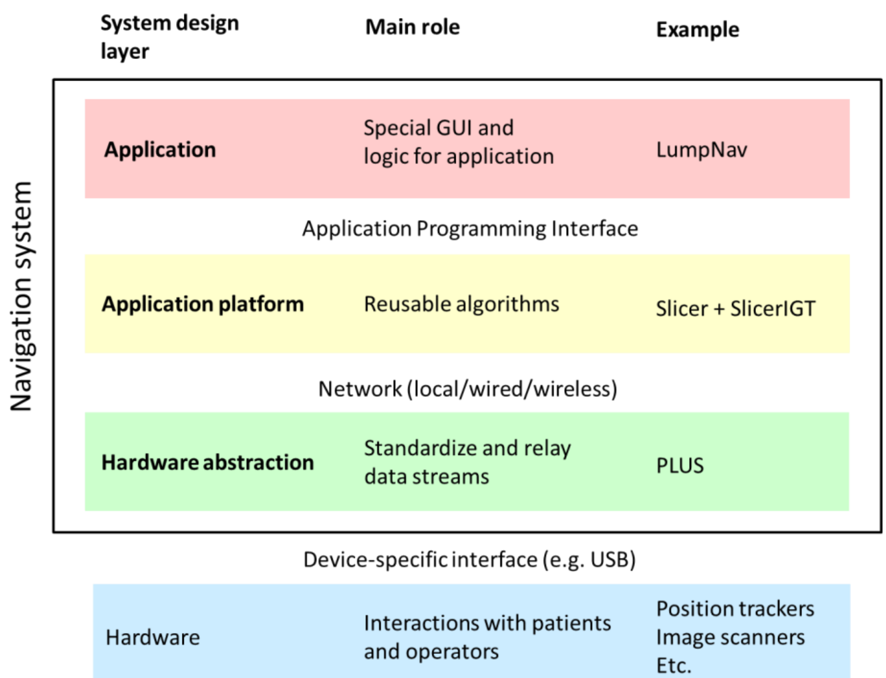

Figure 1. Architecture of medical intervention navigation software.

Research software needs to communicate with commercial imaging, tracking, and other devices to obtain data in real time. New applications can only be developed efficiently if these communication interfaces are shared and reused. Copying source code from a previous application is not an efficient way of reuse. If an issue is fixed in one application, it should be automatically applied in all applications that use the same module. Network communication is a way to connect software modules that work on all operating systems and with all devices. If the modules are on the same computer, it still allows fast communication. When a module is on a separate computer, it can connect with the data acquisition module on another computer through both wired and wireless network. This allows to connect modules to the system that can only run on one operating system, without modifying anything in the rest of the navigation system. OpenIGTLink is the common communication protocol between system modules. Some commercial device manufacturers have not adopted OpenIGTLink yet, therefore an intermediate hardware abstraction software can be used between commercial devices and research applications. The PLUS toolkit (www.plustoolkit.org) is probably the most commonly used hardware abstraction software in navigated medical interventions. In our example, PLUS communicates with the ultrasound machine and the electromagnetic position trackers mounted on surgical instruments. PLUS synchronizes all data streams and sends them in real time to 3D Slicer through OpenIGTLink.

It is not just the hardware and the application that needs to be separated in system design. Reusable application components also need to be separated from application-specific components. Reusable components form an application platform. Reimplementation of common application features is the single greatest cause of inefficient research system development. Navigation systems have so much in common that only a fraction of the source code is specific to an application or workflow. Most visualization, calibration, and data processing components are shared between applications. The SlicerIGT platform implements these features and provides both programming interfaces and graphical user interfaces for its modules. This allows testing ideas without programming. Most system concepts can be tested only by using the existing SlicerIGT modules. And when a special functionality is needed, it can be added to the system with minimal coding, since all modules come with a programming interface that can be called from a new program module. The LumpNav application borrows most of its functionality from SlicerIGT modules through Python function calls. SlicerIGT also provides a user interface template that minimizes programming for a full-screen interface customized for the lumpectomy procedure workflow.

Distribution of the LumpNav application source code in different design layers is an excellent demonstration of minimal application-specific coding. Although the application has a fully customized user interface, optimized for touchscreens, implementing all application-specific functions, still the LumpNav module source code takes only 0.01% of the total size of source code (970 lines) compiled into the executable application (Figure 2). The advantage of such a small application-specific source code is that it can be maintained or modified extremely fast, and with minimal software engineering skills.

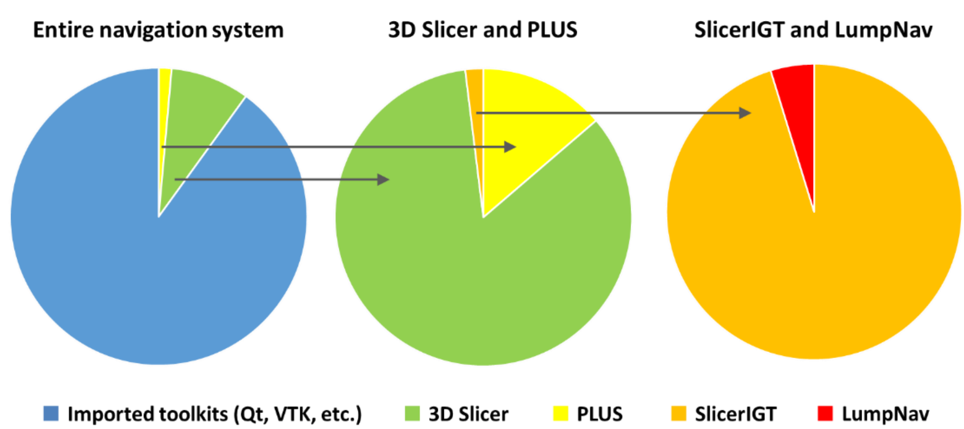

Figure 2. Size of source code in software components of the LumpNav navigation system. The LumpNav module that implements all application-specific functions, is only 970 lines of code in the Python programming language.

4. From bench to bedside

Clinical trials are very important in interventional research, because the human body and its diseases are often too complex and unique to simulate in synthetic phantoms, cadavers, or animals. The real value of a new methods is uncertain until they are tested in clinical trials. Experimental systems need many features before they can be used in patients, and the implementation of these features rarely have any research aspect. Systems need to fit in an often already crowded operating room, any hardware parts for invasive procedures need to be sterilized, and operators with limited technical background should be comfortable at operating the system while they are operating on a patient (Figure 3). Just to name a few of the required features.

Figure 3. Navigated breast cancer surgery using open-source software (PLUS, 3D Slicer, and SlicerIGT) and open-source 3D-printed tools. A simple, custom user interface based on SlicerIGT is provided for the surgeons to operate the system through a sterile touch-screen tablet interface.

PLUS toolkit and SlicerIGT are designed both for researchers not familiar with software programming, and for those who plan to make their custom user interfaces built on a robust platform. A complete clinical application can be created with these tools for both research and commercial applications. PLUS includes a hardware abstraction server that runs based on a user-defined configuration file. The configuration file specifies the hardware devices, calibrations, and data processing routes. PLUS server takes real-time data from devices and streams it to 3D Slicer or other applications. PLUS data streams can also be simultaneously saved into files for offline analysis. SlicerIGT contains an array of modules for 3D Slicer that implement all commonly needed functions in interventional applications. To take advantage of these modules, users need to get familiar with the somewhat complex 3D Slicer user interface. To create simplified custom user interfaces, SlicerIGT offers a module to be used by application developers. This template module, called Guidelet, can be used to fully customize what the user sees on the screen, while the same computational methods run in the background as with the full 3D Slicer application. Implementing a custom clinical workflow using Guidelet takes only a few hundred lines of Python code.

Although shared, open-source software is a great advantage for the research community, an effective platform should also support other aspects of the clinical translation process. When using position trackers, it is a common problem to mount them on existing medical tools without interfering with its normal functions. The PLUS toolkit includes a set of hardware components that can be replicated with 3D printers (Figure 4). These components readily interface with some commercial devices, but can also be edited and used as templates when creating similar components for new devices. Additionally, the PLUS model catalog includes other tools such as calibration and training phantoms. SlicerIGT tutorials contain information on how to use the software, and also provide best practices for system calibration, verification, and workflow design.

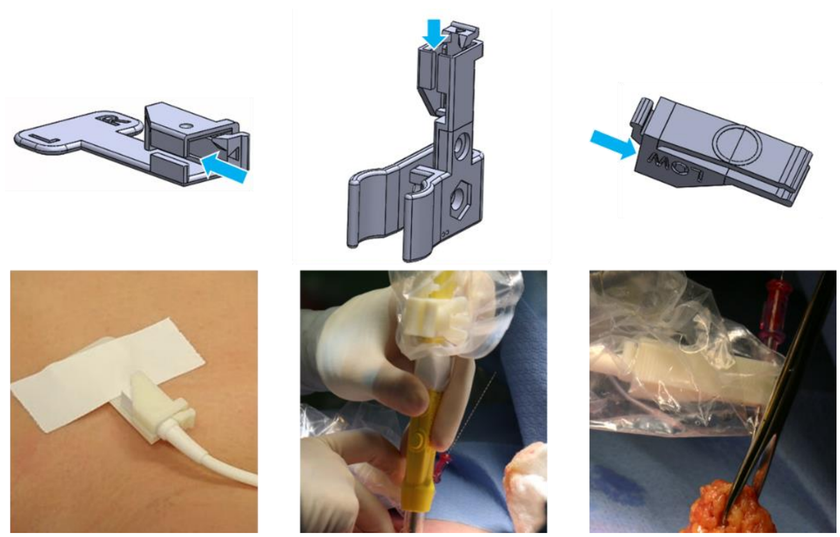

Figure 4. Open-source 3D printable device connectors for tracking patient (left), electrocautery (middle), and needle (right). Tools used in sterile field are bagged. Blue arrows indicate slots for electromagnetic tracking sensors.

5. Ongoing efforts

Existing platforms have maturated through generations of researchers and developers. New applications can benefit from readily available platform components more than ever in the past. Efforts should now be focused on improving user experience and creating systems prepared for translation to clinical products.

Development and marketing of medical devices are subject to extensive regulatory controls. Manufacturers are typically required to demonstrate safety and suitability for the claimed purpose, prove to follow good manufacturing practices, and monitor performance of sold devices and follow up with corrective actions as needed. Building systems on well-established, high-quality platforms may reduce time of product development to a fraction compared to a ground-up development approach and may also reduce efforts for fulfilling other requirements. Leading platforms are expected to bring product development support to the next level by formalizing their quality management system and providing documentation and test results that are compatible with regulatory requirements.

Platforms, such as PLUS and 3D Slicer already developed by following good manufacturing practices in several areas, such as change control, issue tracking, automatic testing, code and design reviews and documentation. Developers of the 3D Slicer are in the process of making the application's quality management system better aligned with medical device requirements by preparing complete approval application packages for the U.S. Food and Drug Administration (FDA) for a few clinical applications. The assumption is that clinical translation and regulatory approval of further products built on the same platform will be greatly simplified due the large amount of shared software, processes, and documentation.

Although a single, very high quality, universal platform for navigated image-guided interventions would be ideal, research groups cannot abandon their existing platforms developed over decades. Different platforms will continue to exist for a long time. However, the number of these platforms should not stand in the way of effective research and ease of clinical translation and commercialization. Developers of these platforms should continuously talk to each other, share their best practices, and create compatible software with as much interoperability and source code reuse as possible.

References

Askeland, C., Solberg, O.V., Bakeng, J.B., Reinertsen, I., Tangen, G.A., Hofstad, E.F., Iversen, D.H., Våpenstad, C., Selbekk, T., Langø, T., Hernes, T.A., Olav Leira, H., Unsgård, G., Lindseth, F., 2016. CustusX: an open-source research platform for image-guided therapy. Int J Comput Assist Radiol Surg. 11(4):505-19.

Clarkson, M.J., Zombori, G., Thompson, S., Totz, J., Song, Y., Espak, M., Johnsen, S., Hawkes, D., Ourselin, S., 2015. The NifTK software platform for image-guided interventions: platform overview and NiftyLink messaging. Int J Comput Assist Radiol Surg. 10(3), 301-16.

Enquobahrie, A., Gobbi, D., Turek, M., Cheng, P., Yaniv, Z., Lindseth, F., Cleary, K., 2008. Designing Tracking Software for Image-Guided Surgery Applications: IGSTK Experience. Int J Comput Assist Radiol Surg. 3(5), 395-403.

Fedorov, A., Beichel, R., Kalpathy-Cramer, J., Finet, J., Fillion-Robin, J.C., Pujol, S., Bauer, C., Jennings, D., Fennessy, F., Sonka, M., Buatti, J., Aylward, S., Miller, J.V., Pieper, S., Kikinis, R., 2012. 3D Slicer as an image computing platform for the Quantitative Imaging Network. Magn Reson Imaging. 30(9), 1323-41.

Lasso, A., Heffter, T., Rankin, A., Pinter, C., Ungi, T., Fichtinger, G., 2014. PLUS: open-source toolkit for ultrasound-guided intervention systems. IEEE Trans Biomed Eng. 61(10), 2527-37.

Tokuda, J., Fischer, G.S., Papademetris, X., Yaniv, Z., Ibanez, L., Cheng, P., Liu, H., Blevins, J., Arata, J., Golby, A.J., Kapur, T., Pieper, S., Burdette, E.C., Fichtinger, G., Tempany, C.M., Hata, N., 2009. OpenIGTLink: an open network protocol for image-guided therapy environment. Int J Med Robot. 5(4), 423-34.

Ungi, T., Gauvin, G., Lasso, A., Yeo, C., Pezeshki, P., Vaughan, T., Carter, K., Rudan, J., Engel, C., Fichtinger, G., 2016. Navigated breast tumor excision using electromagnetically tracked ultrasound and surgical instruments. IEEE Trans Biomed Eng. IEEE Trans Biomed Eng. 63(3), 600-6.