MACRo 2015- 5th International Conference on Recent Achievements in Mechatronics, Automation, Computer Science and Robotics

Joint platforms and community efforts in surgical robotics research

Árpád TAKÁCS[1], Sándor JORDÁN[2], Dénes NAGY[3], Péter PAUSITS[4], Tamás HAIDEGGER[5,6], József K. TAR[7] and Imre J. RUDAS[8]

[1]Antal Bejczy Center for Intelligent Robotics,
Óbuda University, Budapest, Hungary, email: arpad.takacs@irob.uni-obuda.hu
[2]Antal Bejczy Center for Intelligent Robotics,
Óbuda University, Budapest, Hungary, email: sandor.jordan@irob.uni-obuda.hu
[3]Antal Bejczy Center for Intelligent Robotics,
Óbuda University, Budapest, Hungary, email: denes.nagy@irob.uni-obuda.hu
[4]Antal Bejczy Center for Intelligent Robotics,
Óbuda University, Budapest, Hungary, email: pausits.peter@hok.uni-obuda.hu
[5]Antal Bejczy Center for Intelligent Robotics,
Óbuda University, Budapest, Hungary, email: tamas.haidegger@irob.uni-obuda.hu
[6]Austrian Center for Medical Innovation and Technology,
Wiener Neustadt, Austria
[7]Antal Bejczy Center for Intelligent Robotics,
Óbuda University, Budapest, Hungary, email: tar.jozsef@nik.uni-obuda.hu
[8]Antal Bejczy Center for Intelligent Robotics,
Óbuda University, Budapest, Hungary, email: rudas@uni-obuda.hu

Abstract: In modern medical research and development, the variety of research tools has extended in the previous years. Exploiting the benefits of shared hardware platforms and software frameworks is crucial to keep up with the technological development rate. Sharing knowledge in terms of algorithms, applications and instruments allows researchers to help each other's work effectively. Community workshops and publications provide a throughout overview of system design, capabilities, know-how sharing and limitations. This paper provides sneak peek into the emerging collaborative platforms, focusing on available open-source research kits, software frameworks, cloud applications, teleoperation training environments and shared domain ontologies.

Keywords: surgical robotics, shared hardware platforms, software frameworks, cloud applications, teleoperation training

1 Introduction

Medical robotics is one of the most rapidly developing fields of modern robotics, which is partly due to its competitiveness. The manufacturers and developers consider these high–end hardware platforms and software programs the key assets of the research, protecting them in various ways. Nevertheless, there is a growing need for open–source and easily accessible platforms and software/hardware solutions to facilitate future development in all fields of robotics. Various high-end robot controllers have already been available for such purposes, such as the Real-Time Application Interface (RTAI) for Linux platforms [1]. There has been a significant rise of open-source efforts in the field of Computer-Integrated Surgery (CIS), encouraging numerous key industrial stakeholders to support these efforts.

In medical robotics, due to the uniqueness and physical dimensions of hardware platforms, there is a lack of mobility and accessibility for most of the developers in the developer community. The sharing of program codes, toolkits and frameworks are usually carried out through online databases, granting access to the hardware through cloud–based control platforms. In the past decades, the concept of medical robotics could never been separated from the terms of telerobotics and teleoperation. With the recent rise of cloud robotics, a promising perspective has appeared where not only the development and research, but the process of testing and operation could also be applied through cloud-based platforms.

The paper is organized as follows. In Section 2, some the most relevant medical robotics software platforms are presented. In Section 3, some issues are discussed with the System-related application programming interfaces, addressing the research hardware environment in Section 4. Section 5 is dedicated to the da Vinci Research Kit, followed by the discussion and the projection of a future roadmap.

2 Software in Medical Robotics

In this paper, some of the open-source and free platforms and solutions are discussed, where one has complete control over software components, allowing program code customizations and other software development efforts. In most cases, there exists a wide community of developers, which continuously maintains, develops and updates the software. Some of these open-source platforms are listed below.

2.1 3D Slicer

3D Slicer is most likely the most popular and most widely used open-source, free software package that can be used for visualization and image analysis, particularly for medical imaging [2]. The software was design so that it would be natively available for multiple operation system platforms, including Windows, Linux and Mac OS X. 3D Slicer is operated based on the NA-MIC kit and other various software components [3].

These components include the Visualization Toolkit (VTK) and the Insight Segmentation and Registration Toolkit (ITK), which will be discussed later in this paper. The modularity of the Slicer 3D is shown in *Fig. 1.* Both research and clinical projects employed the 3D slicer in great numbers [4], such as brain tumor removal [3] or prostate biopsy, using the OpenIGTLink robotic platforms [5]. The 3D Slicer has a remarkable flexibility and adaptability to other software platforms, such as the Open Core Control software for surgical robotics [6]. One of the main features of the 3D Slicer platform is that besides visualizing the actuator positions in a given application, so–called *virtual fixtures*—control boundaries that should not be crossed during an intervention—can also be specified.

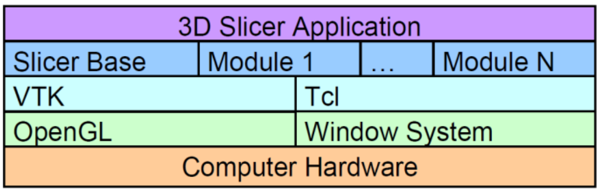

Figure 1: The modularity of Slicer 3D [2]

2.2 Visualization Toolkit

The Visualization Toolkit (VTK) is a free toolkit, its primary use are image processing, visualization, 3D volume rendering in computer graphics, scientific visualization and information visualization [7]. Due to the object oriented design, other modules can be integrated into the software for modification and expansion purposes [8]. VTK includes a C++ class library and other interpreted interface layers such as Tcl/Tk, Java and Python. The 3D Slicer framework is also based on VTK.

2.3 Insight Segmentation and Registration Toolkit

Similarly to the VTK, the Insight Segmentation and Registration Toolkit (ITK) is an open-source, cross-platform system, mostly used in segmentation and image registration problems [9]. ITK includes a vast collection of biomedical image analysis that was created within the framework of Visible Human Project [10]. ITK is also used by 3D Slicer, as a component.

2.4 Computer Integrated Surgical Systems and Technology

Computer Integrated Surgical Systems and Technology (CISST) is an extended collection of libraries, a useful tool for many CIS and medical robotics applications. The function and libraries are used by the Surgical Assistant Workstation, a cross-platform framework based on C++ for device integration in computer assisted intervention applications. CISST supports interchangeability, therefore all the devices that meet the basic requirements are interoperable with each other [11].

2.5 Image-Guided Surgery Toolkit

The Image-Guided Surgery Toolkit (IGSTK) was created to support the development of various image-guided applications [12], where intra-operative tracking is also possible. The main features of the IGSTK toolkit include [13]:

1. reading and display of medical images,
2. interface to common tracking,
3. GUI and visualization capabilities,
4. multi-scale axial view,
5. four-quadrant view (axial, sagittal, coronal or 3D),
6. point-based registration,
7. robust common internal services for logging, exception-handling and problem resolution.

2.6 Medical Imaging Interaction Toolkit

The Medical Imaging Interaction Toolkit (MITK) is an open-source software system for development of interactive medical image processing software. MITK combines the VTK and ITK toolkits with several customized interactive components [14]. The versatility of the hardware platforms is increased due to the combination of these elements. The most prominent module of the software system is the in-built interactive image segmentation.

2.7 Public Software Library for UltraSound

The Public software Library for UltraSound (PLUS) is a software platform written in C++ and built on the NA-MIC Kit [15]. PLUS contains library functions and applications, supporting tracked ultrasound image acquisition, calibration and processing. The software package is equipped with numerous tools that are related to ultrasound data processing, extended with the support of optical and electro-magnetic trackers and other imaging device [16].

2.8 National Alliance for Medical Image Computing

The National Alliance for Medical Image Computing (NA-MIC) is an interdisciplinary team of medical experts, software engineers and computer scientists, striving to develop new computational tools for medical image data visualization and analysis. Therefore, the NA-MIC kit is not a standalone software, rather a collection of methodologies and tools [17]. Numerous software packages are integrated in this kit, such as the 3D Slicer, VTK and ITK.

2.9 Surgical Assistant Workstation

The main purpose of the Surgical Assistant Workstation (SAW) is to integrate different components of a robotic surgical system, using and reusing the elements in the system structure, as shown in *Fig. 2*. Developed by the Johns Hopkins University, SAW supports the most common tools of CIS, such as tracking systems, stereo viewers, haptic devices, and other common hardware platforms, including 3D Slicer, various medical research robots and the da Vinci master console and robotic arms, created by Intuitive Surgical Inc. [18]. SAW is written in C++ and the research was founded by the National Science Foundation (NSF). Thanks to the high level of modularity, SAW can be extended with new components in any research system. The easy connectivity among many devices allows one to integrate them into one sophisticated surgical system. One of these examples was demonstrated by JHU by integrating a snake robot with the da Vinci console for laryngeal surgery [19].

2.10 The Common Toolkit

Common Toolkit (CTK) supports biomedical image computing, licensed under Apache 2.0. The toolkit can be used for academic, commercial and other purposes free of any restrictions. The main scope of the current CTK development efforts includes the *DICOM, DICOM Application Hosting, Widgets and Plugin Framework* [20].

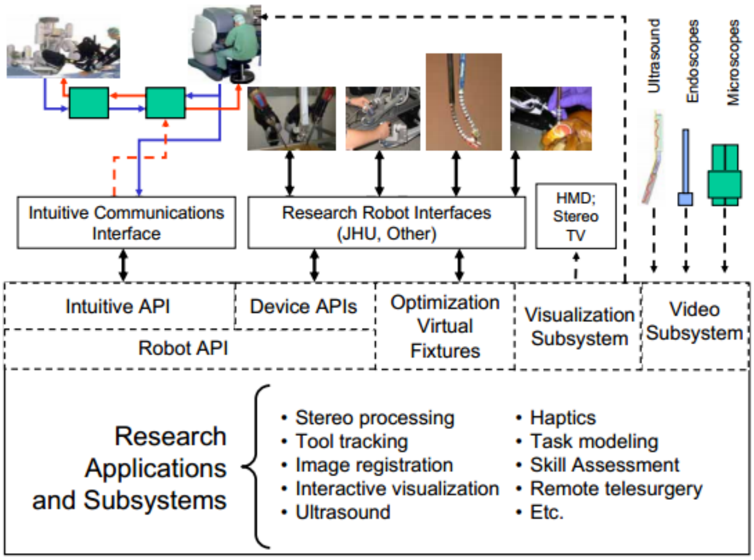

Figure 2: Architecture of Surgical Assistant Workstation [19]

2.11 The NifTK

NifTK is a translational imaging platform combining various toolkits developed at the Centre for Medical Image Computing (CMIC), University College London (UCL). These toolkits include the NiftyReg, a collection of programs to perform rigid, affine and non-linear registration for medical images; the NiftySim, a solver for non-linear elastic or viscoelastic deformation; the NiftyRec, a software package for fully 3D Stochastic Emission Tomographic Reconstruction; the NiftySeg for image segmentation; and NiftyView, a cross-platform graphical user interface providing an entry point to the above mentioned packages. The NifTK is widely used in rigid instrument tracking and computer aided surgery planning [21].

3 System-related APIs

In general, the academic community aims for the development of generic solutions, supporting a wide range of components and devices. However, some manufacturers have recently developed particular Application Programming Interfaces (APIs) for various systems, such as Medtronics' intra-operative navigation platforms feature a StealthLink research interface [22].

Today's most deployed surgical system, the da Vinci Surgical System is not provided with open access by default. Data cannot be retrieved from the robot, programs and components are not subject to change and one cannot extract any information about the basic operation principle, mostly due to liability issues. Limited amount of information can be recorded using various data collection tools [23]. In some cases, the manufacturer allows access to previous generations of their systems, which become transparent by the provided open-source software [24].

4 Research Hardware Environment

In close systems, it is fairly difficult to conduct fundamental research. Therefore, in order to achieve technological development, some of the manufacturers grant partial accessibility to their closed systems. In the case of the da Vinci Surgical System, there exists a real-time stream of kinematic and user event data from the robot that can be read, provided by the de Vinci Application Programming Interface. It is important to mention that the total replacement of certain components, such as the controller body, can transform the da Vinci system into and open-source platform.

Raven II is one of the most successful open-source robotic platforms. Developed at the University of Washington and supported by DARPA, the Raven II became the greatest competitor of the da Vinci system [25]. Furthermore, with the help of the National Institutes of Health, 8 robots have been created and distributed to European and North-American locations. The platform operates based on the Robot Operating System (ROS) architecture.

5 The da Vinci Research Kit

The da Vinci Research Kit (DVRK) is one of the most capable research platforms in surgical robotics. In fact, the kit is a collection of retired, first-generation da Vinci robot components and tools, provided with additional open-source control electronics and software.

5.1 Hardware Components

The DVRK contains the components listed below:

1 Two da Vinci Master Tool Manipulators (MTMs)
2 Two da Vinci Patient Side Manipulators (PSMs)
3 A stereo viewer
4 A foot pedal tray
5 Manipulator Interface Boards (dMIBs)
6 Basic accessory kit

The research kit therefore contains the original, unmodified mechanical components. It is possible to transform a da Vinci Classic system into a research kit, although some of the components are not available for researchers due to its commercial use and development. The Endoscopic Camera Manipulator (ECM) is not included along with several other components from the original system, but the lack of these elements is not a major issue from the development point of view. In general, for research purposes, the control electronics and control software are the most essential parts of the system.

Recently, a novel, open controller platform was created by JHU, Worchester Polytechnic Institute (WPI) and their partners [26]. The source files of the control electronics were also published online. The research platform is equipped with an IEEE 1394a Firewire interface, capable of maintaining a communication speed of 400Mbit/sec. In order to achieve a satisfactory degree of security and reliability, in surgical robotics it is crucial to create real-time communication between the devices in the system. The control box includes two FPGA modules and two Quad Linear Amplifiers (QLA), shown in *Fig. 3*. The assembly described above is capable of driving and controlling a single robotic tool. Two da Vinci Master Tool Manipulators (MTMs) and two da Vinci Patient Side Manipulators (PSMs) can be controlled using four sets of control electronics, requiring a total of 8 pieces of FPGAs and QLAs.

The da Vinci Research Kit is based on the centralized computation and distributed I/O architecture [27]. The main advantage of this structure is that there is only one control electronics that maintains contact with the peripheral inputs and outputs, allowing one central computer unit to perform the calculations, located at the control units. In general, the central unit is a Linux-based computer with some real-time component expansion.

5.2 Low Level Software Architecture

The FPGA module firmware is available online, which was given a BSD license, therefore it can be freely modified. The RT-FireWire is one of the best approaches to solve the real-time communication between the subsystems over firewire, while the communication implementation is achieved through standard Linux C++ libraries [28].

The PC-side operating system is preferably Linux-based, as there exists a real-time extension (RTLinux), a Linux OS that runs under the supervision of a hard real-time microkernel [29]. The software architecture, as a whole, can be divided into five *functional layers* (I-V) and three *development layers* (A-C) [30]. The functional layers, implemented on the PC side, are stratified by the complexity of their function, while the development layers are sorted by the programming language complexity they use. The open-source property is extensively supported by the previously described SAW and CISST libraries, allowing the system to be used as a completely open research platform.

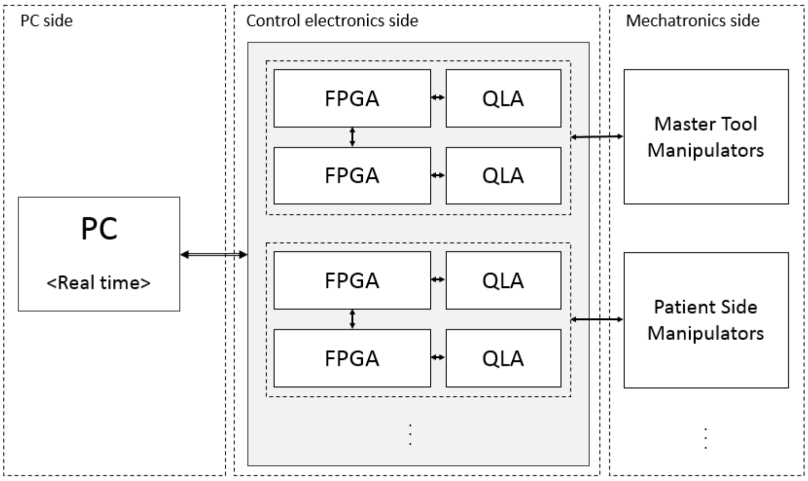

Figure 3: Schematic representation of the hardware structure Workstation

6 Conclusion and Future Roadmap

The da Vinci Research Kit is one of the greatest breakthroughs in the field of open-source surgical robotic research and development, which is mostly due to the direct access to an actual clinical system, even though these systems are retired and out–of-date. As of October, 2014, there are 17 DVRK research teams operating around the world, maintaining an active community through meetings and workshops. Particularly in Europe, several actions and projects (EuroSurge, I-SUR, ACTIVE) have given a boost to synchronized robotics research, mostly founded by the EU committee. IEEE Robotics and Automation Society has also contributed to the generalization of surgical robotics through study groups, and there are initiatives for forming workgroups for surgical robotics ontologies. A great impact on the entire research field is expected, where more and more attention is to be given to open-source research instead of strictly commercial development.

The effectiveness of surgical robotics will evidently become higher with the use of open-source platforms and software. This paper reviewed the most widely-used, currently available software and hardware research platforms, aided with some highlights to the features and recently realized projects they supported.

7 Acknowledgement

T. Haidegger is a Bolyai Fellow of the Hungarian Academy of Sciences.

References

[1] L. Dozio and P. Mantegazza, "Real-time distributed control systems using RTAI," in Proc. of the 6th IEEE Intl. Symp. on Object-Oriented Real-Time Distributed Computing, Hakodate, 2003, pp. 11–18.

[2] Pieper, S., Halle, M., & Kikinis, R. (2004, April). 3D Slicer. In Biomedical Imaging: Nano to Macro, 2004. IEEE International Symposium on (pp. 632-635).

[3] Arata, J., Tada, Y., Kozuka, H., Wada, T., Saito, Y., Ikedo, N., ... & Fujimoto, H. (2011). Neurosurgical robotic system for brain tumor removal. International journal of computer assisted radiology and surgery, 6(3), 375-385.

[4] Haidegger, T., Kovacs, L., Fordos, G., Benyo, Z., & Kazanzides, P. (2008, January). Future trends in robotic neurosurgery. In 14th Nordic-Baltic Conference on Biomedical Engineering and Medical Physics (pp. 229-233). Springer Berlin Heidelberg.

[5] A. Lasso, J. Tokuda, N. Hata, G. Fichtinger, Robot-assisted MRI-guided prostate biopsy using 3D Slicer. NA-MIC Tutorial Contest, 2010. Available: http://www.na-mic.org/Wiki/images/4/43/ DBP2JohnsHopkinsTransRectalProstateBiopsy.pdf

[6] Arata, J., Kozuka, H., Kim, H. W., Takesue, N., Vladimirov, B., Sakaguchi, M., ... & Fujimoto, H. (2010). Open core control software for surgical robots.International journal of computer assisted radiology and surgery, 5(3), 211-220.

[7] Kitware, About Visualization Toolkit, http://www.vtk.org/VTK/project/about.html, 2014

[8] Schroeder, W. J., Avila, L. S., & Hoffman, W. (2000). Visualizing with VTK: a tutorial. Computer Graphics and Applications, IEEE, 20(5), 20-27.

[9] Kitware, About Insight Segmentation and Registration Toolkit, http://www.itk.org/ITK/project/about.html, 2014

[10] U.S. National Library of Medicine, The Visible Human Project – Appplications, http://www.nlm.nih.gov/pubs/factsheets/visible_human.html, 2014

[11] Deguet, A., Kumar, R., Taylor, R., & Kazanzides, P. (2008, September). The cisst libraries for computer assisted intervention systems. In MICCAI Workshop on Systems and Arch. for Computer Assisted Interventions, Midas Journal.

[12] Gary, K., Ibanez, L., Aylward, S., Gobbi, D., Blake, M. B., & Cleary, K. (2006). IGSTK: an open source software toolkit for image-guided surgery. Computer,39(4), 46-53.

[13] IGSTK. What is IGSTK? Available: http://public.kitware.com/IGSTKWIKI/index.php/What_is_IGSTK, 2014

[14] ITK. The Medical Imaging Interaction Toolkit (MITK), Available: http://www.mitk.org/MITK, 2014.

[15] Lasso, A., Heffter, T., Pinter, C., Ungi, T., Chen, T. K., Boucharin, A., Fichtinger, G. (2011, October). PLUS: An open-source toolkit for developing ultrasound-guided intervention systems. In 4th Image Guided Therapy Workshop (Vol. 4, p. 103).

[16] Lasso, A., Heffter, T., Pinter, C., Ungi, T., & Fichtinger, G. (2012). Implementation of the PLUS open-source toolkit for translational research of ultrasound-guided intervention systems. MICCAI-Systems and Architectures for Computer Assisted Interventions, 1-12.

[17] Pieper, S., Lorensen, B., Schroeder, W., & Kikinis, R. (2006, April). The NA-MIC Kit: ITK, VTK, pipelines, grids and 3D slicer as an open platform for the medical image

computing community. In Biomedical Imaging: Nano to Macro, 2006. 3rd IEEE International Symposium on (pp. 698-701). IEEE.
[18] CISST, System Requirements for Surgical Assistant Workstation ,Available: https://www.cisst.org/main/images/c/cd/SAW-SystemRequirements-Rev2.pdf, 2007
[19] Vagvolgyi, B., DiMaio, S., Deguet, A., Kazanzides, P., Kumar, R., Hasser, C., & Taylor, R. (2008, September). The surgical assistant workstation. In Proc MICCAI Workshop: Systems and Architectures for Computer Assisted Interventions.
[20] I. Wolf, "Toolkits and Software for Developing Biomedical Image Processing and Analysis Applications," in Biomedical Image Processing, T. M. Deserno, Ed. Springer Berlin Heidelberg, 2011, pp. 521–544.
[21] G. Zombori, R. Rodionov, M. Nowell, M. A. Zuluaga, M. J. Clarkson, C. Micallef, B. Diehl, T. Wehner, A. Miserochi, A. W. McEvoy, J. S. Duncan, and S. Ourselin, "A Computer Assisted Planning System for the Placement of sEEG Electrodes in the Treatment of Epilepsy," in Information Processing in Computer-Assisted Interventions, D. Stoyanov, D. L. Collins, I. Sakuma, P. Abolmaesumi, and P. Jannin, Eds. Springer International Publishing, 2014, pp. 118–127.
[22] StealthLink Manual v09, Medtronic, 2005
[23] Reiley, C. E., Lin, H. C., Yuh, D. D., & Hager, G. D. (2011). Review of methods for objective surgical skill evaluation. Surgical endoscopy, 25(2), 356-366.
[24] Leven, J., Burschka, D., Kumar, R., Zhang, G., Blumenkranz, S., Dai, X. D., ... & Taylor, R. H. (2005). DaVinci canvas: a telerobotic surgical system with integrated, robot-assisted, laparoscopic ultrasound capability. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2005 (pp. 811-818). Springer Berlin Heidelberg.
[25] IEEE Pulse.Surgical Robots in Space: Sci-fi and Reality Intersect, Available: http://pulse.embs.org/july-2014/surgical-robots-space-sci-fi-reality-intersect/, 2014.
[26] AIM WPI. daVinci Surgical Robot Robot Research System (dVRK), Available: http://aimlab.wpi.edu/research/projects/ daVinci_Robot_Research_System, 2014.
[27] Chen, Z., Deguet, A., Taylor, R. H., DiMaio, S., Fischer, G. S., & Kazanzides, P. (2013, August). An open-source hardware and software platform for telesurgical robotics research. In MICCAI Workshop on Systems and Arch. for Computer Assisted Interventions, Midas Journal: http://hdl. handle. net/10380/3419.
[28] Zhang, Y., Orlic, B., Visser, P., & Broenink, J. (2005). Hard real-time networking on FireWire.
[29] Barabanov, M., & Yodaiken, V. (1996). Real-time linux. Linux journal, 23.
[30] Kazanzides, P., Chen, Z., Deguet, A., Fischer, G. S., Taylor, R. H., & DiMaio, S. P. An Open-Source Research Kit for the da Vinci R Surgical System. In Pro of the IEEE International Conference on Robotics & Automation (ICRA), Hong Kong, pp. 6434-6439, 2014.