

## 3D Slicer as an image computing platform for the Quantitative Imaging Network

Andriy Fedorov<sup>a,\*</sup>, Reinhard Beichel<sup>b</sup>, Jayashree Kalpathy-Cramer<sup>c</sup>, Julien Finet<sup>d</sup>,  
Jean-Christophe Fillion-Robin<sup>d</sup>, Sonia Pujol<sup>a</sup>, Christian Bauer<sup>b</sup>, Dominique Jennings<sup>c</sup>,  
Fiona Fennessy<sup>a</sup>, Milan Sonka<sup>b</sup>, John Buatti<sup>b</sup>, Stephen Aylward<sup>d</sup>,  
James V. Miller<sup>e</sup>, Steve Pieper<sup>f</sup>, Ron Kikinis<sup>a</sup>

<sup>a</sup>Brigham and Women's Hospital, Harvard Medical School, Boston, MA 02115, USA

<sup>b</sup>University of Iowa, Iowa City, IA 52242, USA

<sup>c</sup>Massachusetts General Hospital, Harvard Medical School, Charlestown, MA 02129, USA

<sup>d</sup>Kitware Inc, Clifton Park, NY 12065, USA

<sup>e</sup>GE Research, Niskayuna, NY 12309, USA

<sup>f</sup>Isomics, Inc., Cambridge, MA 02138, USA

Received 1 February 2012; revised 26 April 2012; accepted 29 May 2012

### Abstract

Quantitative analysis has tremendous but mostly unrealized potential in healthcare to support objective and accurate interpretation of the clinical imaging. In 2008, the National Cancer Institute began building the Quantitative Imaging Network (QIN) initiative with the goal of advancing quantitative imaging in the context of personalized therapy and evaluation of treatment response. Computerized analysis is an important component contributing to reproducibility and efficiency of the quantitative imaging techniques. The success of quantitative imaging is contingent on robust analysis methods and software tools to bring these methods from bench to bedside.

3D Slicer is a free open-source software application for medical image computing. As a clinical research tool, 3D Slicer is similar to a radiology workstation that supports versatile visualizations but also provides advanced functionality such as automated segmentation and registration for a variety of application domains. Unlike a typical radiology workstation, 3D Slicer is free and is not tied to specific hardware. As a programming platform, 3D Slicer facilitates translation and evaluation of the new quantitative methods by allowing the biomedical researcher to focus on the implementation of the algorithm and providing abstractions for the common tasks of data communication, visualization and user interface development. Compared to other tools that provide aspects of this functionality, 3D Slicer is fully open source and can be readily extended and redistributed. In addition, 3D Slicer is designed to facilitate the development of new functionality in the form of 3D Slicer extensions.

In this paper, we present an overview of 3D Slicer as a platform for prototyping, development and evaluation of image analysis tools for clinical research applications. To illustrate the utility of the platform in the scope of QIN, we discuss several use cases of 3D Slicer by the existing QIN teams, and we elaborate on the future directions that can further facilitate development and validation of imaging biomarkers using 3D Slicer.

© 2012 Elsevier Inc. All rights reserved.

**Keywords:** Cancer imaging; Quantitative imaging; Software tools; Medical imaging; Imaging biomarkers; Image analysis; MRI; PET; CT; Brain; Head and neck; Prostate; Glioblastoma; Cancer treatment response

### 1. Introduction

Cancer is the leading cause of death in the developed world and the second leading cause of death in the developing countries [1]. With the incidence of cancer rapidly increasing, there is an immediate need for better understanding of this disease and for the development of the

\* Corresponding author.

E-mail address: [fedorov@bwh.harvard.edu](mailto:fedorov@bwh.harvard.edu) (A. Fedorov).

targeted, personalized treatment approaches. Successful translation of such treatments from the lab to the clinic is contingent on the availability of biomarkers — objective and testable characteristics indicative of normal or pathologic processes that ideally should allow for quantitative measurement of the response to therapy [2,3]. In this regard, in vivo imaging biomarkers are particularly promising, as they can be highly specific and minimally invasive, providing both anatomical and functional understanding of the response patterns. However, the potential of quantitative imaging remains largely underutilized. The Response Evaluation Criteria in Solid Tumors — the only imaging-based biomarker accepted by the US FDA as a surrogate end point for clinical outcome in therapy — rely primarily on the anatomical imaging of the lesion measured by its largest diameter [4,5]. Continuous advances in multimodality three-dimensional (3D) imaging technology and analysis, along with improvements in computer science and bioinformatics, create an opportunity for a paradigm shift in quantification of treatment response. To advance the role of imaging as a biomarker of treatment, the National Cancer Institute launched the Quantitative Imaging Network (QIN) initiative [6]. The goal of QIN is to form a community of interdisciplinary teams engaged in the development of imaging-based biomarkers and their optimization in the context of clinical trials. Research software platforms are essential in prototyping, development and evaluation of novel algorithmic methods as a mechanism for discovering image-based surrogate end points. Such platforms should also support integration of the algorithmic advances into the clinical trial work flows. In this paper, we discuss the capabilities and the utility of 3D Slicer (Slicer) as an enabling research platform for quantitative image computing research.

3D Slicer is a free open-source extensible software application for medical image computing and visualization. Slicer emerged as a culmination of several independent projects that focused separately on image visualization, surgical navigation and graphical user interface (GUI). David Gering presented the initial prototype of the Slicer software in his MIT Master's thesis in 1999 [7] based on the earlier experience of the research groups at MIT and Surgical Planning Lab (SPL) [8]. Subsequently, Steve Pieper assumed the role of the Chief Architect, commencing the work of transforming 3D Slicer into an industrial-strength package. Since 1999, Slicer has been under continuous development at the SPL under the leadership of Ron Kikinis. Today it is developed mostly by professional engineers in close collaboration with algorithm developers and application domain scientists, with the participation of Isomics Inc., Kitware Inc. and GE Global Research and with significant contributions from the growing Slicer community. Initially envisioned as a neurosurgical guidance, visualization and analysis system [7,9], over the last decade, Slicer has evolved into an integrated platform that has been applied in a variety of clinical and preclinical research applications, as well as for the analysis of nonmedical images [10–21].

Improvement and maintenance of the software have been possible primarily through the support from the National Institutes of Health (NIH). At the same time, its development has grown into a community effort, as numerous groups and individual users not funded directly to develop 3D Slicer are continuously improving it by reporting software problems and contributing solutions, suggesting new features and developing new tools. As described more fully below, Slicer integrates a number of powerful open-source projects into an end-user application suitable for clinical researchers.

The breadth of functionality, extensibility, portability across platforms and nonrestrictive software license are some of the main features that differentiate Slicer from the commercial and open-source software tools and workstations that aim to cover similar aspects of functionality. Numerous choices of radiology workstations and image analysis tools are available from the commercial vendors. Some of the popular tools used in the clinic as well as in research are AW Workstation (GE Healthcare), syngo.via (Siemens), PMOD (PMOD Technologies Ltd., Zurich, Switzerland), Definiens (Definiens Inc., Parsippany, NJ, USA), MimVista (MIM Software Inc., Cleveland, OH, USA). These packages provide users with a set of analysis tools (some of which may be specifically approved by the FDA for certain clinical tasks), compatibility with the clinical Picture Archiving and Communication System (PACS) systems and customer support. Such clinically oriented systems are not always affordable to the academic researchers. Commercial solutions are typically not extensible by the end user, are not oriented towards prototyping of the new tools and may require specialized hardware, limiting their applicability in projects that involve development of new image analysis methods. In the research domain, MATLAB (Mathworks, Natick, MA, USA) has traditionally been the “Swiss army knife” of scientific computing. Many researchers use MATLAB for initial prototyping and experimentation, while some end-user tools, such as SPM [22], are built on top of MATLAB. Being a generic prototyping tool, MATLAB is not designed for medical applications and thus lacks support for interface and display conventions common in clinical environments. As a result, deployment of the developed tools for the use by clinical researchers requires translation of the code into more generic languages to minimize dependencies and simplify integration.

As opposed to the commercial workstations, 3D Slicer is meant to provide a research platform that is freely available and does not require specialized equipment. Slicer's use is not constrained to a single processing task or research application. Its generality and extensibility separate Slicer from such task-oriented packages as ITK-Snap (image segmentation) [23], DtiStudio (diffusion tensor analysis) [24], FreeSurfer,<sup>1</sup> FSL [25] and SPM [22] (neuroimaging applications). Several other tools, such as OsiriX [26],

<sup>1</sup> FreeSurfer. <http://surfer.nmr.mgh.harvard.edu>

BioImage Suite,<sup>2</sup> MIPAV [27] and ImageJ<sup>3</sup> [28], are similar to Slicer in that they provide extensible development platforms for biomedical imaging applications (for a comprehensive comparison of these tools, we refer the reader to the earlier surveys [29,30]). ImageJ is an extensible Java-based image processing platform that has been applied to a variety of applications, including radiological image processing [28], with the focus on two-dimensional (2D) analysis. MIPAV is a cross-platform Java-based package supported by NIH [27]. OsiriX is an open-source PACS workstation and DICOM viewer for Mac OS X that provides advanced capabilities such as image fusion, volume rendering and image annotation, and is extensible via a documented plug-in mechanism [26]. ClearCanvas is a Windows-based DICOM workstation adopted by the caBIG project.<sup>4</sup> The TCGA version of ClearCanvas workstation supports AIM model annotation capabilities [31,32] and is also extensible. A notable aspect of both OsiriX and ClearCanvas is that these systems are made available either in a free open-source version or as commercial, FDA-cleared products. A practical shortcoming is their dependency on specific operating systems (Mac OS X for OsiriX and MS Windows for ClearCanvas). Perhaps more importantly, some of the aforementioned packages that are similar in their intended purpose to 3D Slicer (including BioImage Suite, MIPAV, OsiriX and ClearCanvas) are distributed under restrictive open-source licenses that limit the ability of outside developers to redistribute parts of those systems, in particular, in commercial or other “closed-source” scenarios. This can be a practical constraint for QIN investigators collaborating with industry partners, as the solutions developed on top of these packages cannot be directly incorporated into commercial products. Another consideration is that critical functionality to work with modern imaging scenarios may only be available in the commercial version of the package. OsiriX and ClearCanvas, for example, do not support 64-bit processing architectures in their royalty-free versions, and this limits the maximum size of the image data the software can accept.

3D Slicer does not have any components specific to a particular operating system. Binary distributions are available for 32- or 64-bit versions of Windows, Mac OS X or Linux, and the software can be compiled on other systems, such as Oracle’s Solaris. It is distributed under a BSD-style license agreement [33] allowing free distribution of derivative software for academic and commercial use. Hence, image analysis tools developed within 3D Slicer can be adopted directly by the industry collaborators. Since new technologies can only become part of routine clinical care through their incorporation into FDA-regulated medical products, Slicer’s permissive software license furthers the

overall goal of lowering the barriers for translation of the successful research solutions into medical products. On the other hand, Slicer is not an FDA-approved device, and its license makes no claims about the clinical applicability of the software. It is the sole responsibility of the user to comply with appropriate safety and ethics guidelines, and any products incorporating Slicer technology must be fully tested to comply with applicable laws and regulations. Under these considerations, Slicer has been applied in a variety of projects under appropriate research oversight. In this manuscript, we aim to introduce the capabilities of 3D Slicer as a software platform for clinical imaging research and outline its use in the context of biomarker development for cancer treatment by several QIN sites.

In the remainder of this paper, we first present an overview of 3D Slicer by discussing its architecture, main features and guiding development principles. Next, we focus on the capabilities of 3D Slicer viewed from the perspective of the clinical researcher. We follow with the overview of the 3D Slicer platform from the standpoint of a biomedical engineer and discuss how Slicer can facilitate development of new software tools for clinical research. To demonstrate how 3D Slicer is currently used by some of the existing teams of QIN, we discuss the clinical research projects investigated at Brigham and Women’s Hospital (BWH) (PI Fiona Fennessy), University of Iowa (PI John Buatti) and Massachusetts General Hospital (MGH) (PI Bruce Rosen). We conclude with the summary of our findings, discussing some of the features and functionalities that would further improve applicability of 3D Slicer to biomarker development by the QIN investigators.

## 2. Overview of 3D Slicer

Computerized image analysis plays an increasing role in supporting clinical and research needs. Promising methodologies that may lead to new imaging biomarkers often involve custom image processing software. The role of software evolves over the different stages of the imaging biomarker life cycle. In the inception stage, promising methodological concepts are identified and translated into early prototypes. Such early prototypes are often cobbled together from parts of tools designed for other tasks. They are typically suitable for use by engineers for small pilot or phantom studies. Their purpose is to demonstrate that the task can be done. The next step is to demonstrate that the task is worth doing. For this to happen, the method has to be optimized and thoroughly validated outside the research lab. This step requires a software tool that can be used reliably by trained clinical researchers and can be applied to a larger population of patients, possibly in a multisite clinical trial. At this point, the software must be robust and the interface intuitive. After establishing the value of a tool, the next step is to translate the software and the biomarker into a clinical, FDA-approved product

<sup>2</sup> BioImage Suite, <http://www.bioimagesuite.org/>.

<sup>3</sup> ImageJ, <http://rsbweb.nih.gov/ij/>.

<sup>4</sup> caBIG AIM ClearCanvas Workstation, [https://cabig.nci.nih.gov/tools/AIM\\_ClearCanvas](https://cabig.nci.nih.gov/tools/AIM_ClearCanvas)

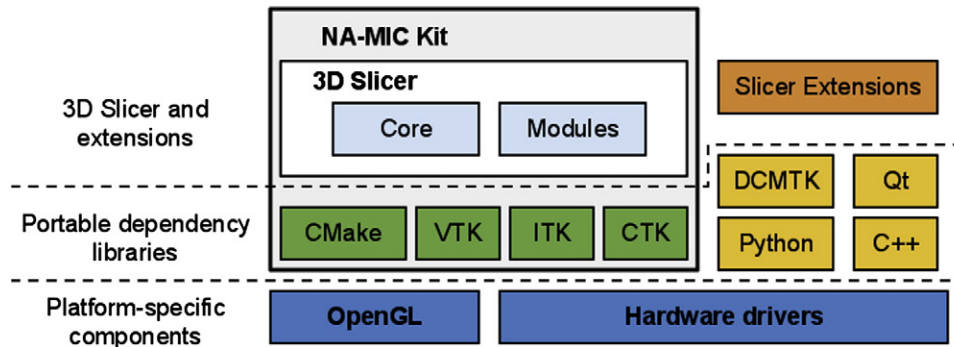


Fig. 1. 3D Slicer “ecosystem”. Slicer is a cross-platform package, but certain requirements on the graphics system, such as OpenGL drivers, should be met to support accelerated rendering. Its dependency libraries are portable across platforms and are distributed under compatible licenses. Those dependencies that are part of NA-MIC Kit often share the 3D Slicer developer community and are developed in synergy with the Slicer efforts. Slicer itself consists of the main application framework (core) and plug-ins (modules). Custom functionality is introduced by implementing external modules (Slicer extensions).

One of the goals of 3D Slicer is to provide a common set of base functionality to facilitate development and validation of the medical image computing methods for the “can it be done?” and “is it worth doing?” steps of the imaging biomarker life cycle. Understanding of the imaging data by visualizing its different aspects, applying basic processing steps and displaying analysis results in context is critical at the initial stages. 3D Slicer is a powerful visualization tool that allows exploration of the imaging data sets in two, three and four dimensions. Slicer enables fusion of functional and anatomical data and provides a variety of generic and specialized tools for their processing and multimodal analysis. To ensure robustness, it is often required that image processing methods are initialized manually. The 3D Slicer framework includes the components needed to meet such development needs (e.g., “seed” the image segmentation algorithm, provide initial pose for image registration or “steer” the processing based on the dynamically updated computation result).

The use of an image analysis tool in a clinical research environment introduces new requirements. Support of the DICOM standard [34] for communicating image data is commonly required, as is a graphical user interface that is usable by a nontechnical operator. Slicer allows loading images in DICOM format from disk or directly from PACS. The Qt toolkit<sup>5</sup> provides a cross-platform GUI front end to Slicer that allows each processing module to easily define custom user interfaces. Image analysis tools implemented as Slicer processing modules can be developed in such a way that both interactive and batch execution is possible. This is particularly valuable in situations when experiments have to be performed on a large number of cases or when certain parameters of the algorithm have to be optimized.

Once a method is optimized and validated, it can potentially be developed into an FDA-cleared clinical device. At this stage, it can be important that the implementation that was developed in the earlier stages

can be used without the need to reimplement the key components. Slicer adopts a licensing model that does not place any restrictions on the use of its source code in the derivative works. This aims to broaden the user community to include both academic and industry partners and to simplify the transitioning of the research tool into a commercial product.

The architecture of 3D Slicer follows a modular and layered approach, as shown in Fig. 1. At the lower level of the architecture are the fundamental libraries provided by the operating system that are not packaged with Slicer, such as OpenGL and hardware drivers that allow efficient usage of the windowing and graphics resources of the host system. At the level above, there are languages (primarily C++ and Python, but increasingly JavaScript) and libraries that provide higher level functionality and abstractions. Some of the libraries used by the application are Qt (cross-platform GUI framework), the DICOM Toolkit (DCMTK)<sup>6</sup> (implements parts of DICOM standard and is used to interact with DICOM data and DICOM services) and jqPlot<sup>7</sup> (provides charting capabilities). All of the external dependencies of 3D Slicer are cross-platform portable and are distributed under licenses fully compatible with Slicer, which do not restrict their use in either commercial or open-source products.

Some of the libraries contributing to the foundation of 3D Slicer are designed in close collaboration and often share the same developer community. These libraries are distributed as part of the National Alliance for Medical Image Computing (NA-MIC) Kit [35], a collection of the software tools supported in part by the NA-MIC project. The tools and interfaces provided by the NA-MIC Kit components are largely focused on the needs of the developers of medical image computing applications. CMake<sup>8</sup> enables cross-platform build system configuration,

<sup>6</sup> DCMTK — DICOM Toolkit, <http://dicom.offis.de/dcmk>.

<sup>7</sup> jqPlot, <http://www.jqplot.com>.

<sup>8</sup> CMake: cross-platform build system, <http://cmake.org>.

<sup>5</sup> Qt cross-platform application and UI framework, <http://qt.nokia.com/>.

packaging and testing of 3D Slicer and NA-MIC Kit libraries. CDash<sup>9</sup> is a Web-based server that organizes the results of software testing. The Visualization Toolkit (VTK)<sup>10</sup> provides the key building blocks for 3D computer graphics and visualization. The Insight Toolkit (ITK)<sup>11</sup> [36] is a library developed specifically for the tasks related to medical image registration and segmentation and for implementing new image analysis algorithms. The Common Toolkit (CTK)<sup>12</sup> is a biomedical image computing library with a focus on application-level DICOM support, plug-in framework and specialized GUI widgets.

3D Slicer itself consists of the lean application core, Slicer modules and Slicer extensions. The core implements the Slicer user interface and provides support for data input/output (IO), visualization and developer interfaces that support extension of the application with new plug-ins. Internally, Slicer uses a scene data structure to organize images and annotations, coordinate layouts and maintain the application state. The XML-based Medical Reality Markup Language (MRML) is used to serialize the content of the scene. Slicer modules are the plug-ins that depend on the Slicer core to implement new functionality. Individual modules can be independent or can rely on other modules (for example, a module that provides lesion segmentation functionality can depend on the volume rendering module to enable 3D visualization of the anatomy and segmented structure). Unlike Slicer modules that are packaged as part of Slicer distribution, Slicer extensions are external plug-ins installed “on demand” by the user, akin to the Web browser extensions. The extensions mechanism enables sharing of the Slicer-based tools that cannot be included into the package due to incompatible licenses, development timelines or other constraints. It also provides a “pathway” for integrating new functionality since any extension that satisfies specific requirements is a candidate to be included into the Slicer distribution. The requirements for integration to the distribution include the following: a designated maintainer to serve as a point of contact, a nonrestrictive Slicer-compliant license, adherence to the Slicer coding standards, availability of software tests and appropriate user-level documentation. We emphasize that tools that do not satisfy these requirements can still be available to Slicer users, clearly marked as Slicer extensions and accompanied by an appropriate disclaimer.

Since its inception in late 1990s, 3D Slicer has been evolving, with major architectural, functional and GUI redesigns occurring every 4–5 years. The current (fourth) generation of Slicer was released in November 2011. The most notable improvements of the software as compared to the previous (third) version are improved visualization performance, reengineered DICOM support, completely

redesigned GUI and availability of the Python development interfaces. With each major redesign, several policies are followed to ease the transitioning of users and developers to the updated platform. The base and core modules are released to the broader development community first, followed by the migration of the specialized modules. For example, the migration strategy for the modules in the image-guided therapy (IGT) category is the most conservative due to the mission-critical nature of these applications. As such, the modules in this category are usually migrated after the base of the new release is stable and thoroughly tested. Another policy has been to provide backward compatibility to support reading of data produced using the earlier version.

Within each generation, new and updated releases of Slicer are prepared every 2 to 6 months. These releases include performance improvements, bug fixes and new functionality, but no major changes to the base architecture or GUI. A release includes a tagged version of the source code in the source code repository as well as binary installation packages for the supported platforms (Windows, Mac OS X, Linux; 32-bit, 64-bit). Daily binary installation packages are also prepared for the major platforms to track the current version of the source code. These daily packages allow Slicer users to access the new functionality under development. The Slicer software development process is collaborative and geographically distributed, with over 80 authorized developers contributing source code through SVN and Git revision control systems. To ensure the stability of the software, the source code is compiled and tested on a daily basis on a variety of platform configurations. The testing results are summarized and reported using a Web-based centralized CDash dashboard. Users and developers of 3D Slicer can also report issues on the open mailing lists or using Web-based bug tracking system.

Documentation, training and user support are high priorities for the Slicer community. Hands-on training sessions are organized regularly as part of ongoing outreach initiatives at major conferences, such as the annual Radiological Society of North America (RSNA), Medical Image Computing and Computer Assisted Interventions (MICCAI) and International Society for Optics and Photonics (SPIE) meetings, or in response to request by host institutions at both national and international venues. Semiannual hands-on week-long project weeks are open for participation to anyone interested in developing or using Slicer tools, and are an ideal place to exchange ideas and experience. Remote learning of 3D Slicer is supported by the online resources and community mailing lists. Focused training materials that include sample data sets and step-by-step instructions are available for basic Slicer operation as well as advanced workflows. Wiki-based Web pages accompany every module of the 3D Slicer, providing reference documentation of functionality and usage examples. User and developer community mailing lists have been active for the last 10 years and

<sup>9</sup> CDash: software Web-based testing server, <http://cdash.org>.

<sup>10</sup> VTK: Visualization Toolkit, <http://vtk.org>.

<sup>11</sup> ITK: Insight Toolkit, <http://itk.org>.

<sup>12</sup> CTK: The Common Toolkit, <http://commonstk.org>.

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

## LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

## FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

## E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.