Performing radiation therapy research using the open-source SlicerRT toolkit

Csaba Pinter[1], Andras Lasso[1], An Wang[2], Gregory C. Sharp[3], Kevin Alexander[4], David Jaffray[2,5], and Gabor Fichtinger[1]

[1]Laboratory for Percutaneous Surgery, School of Computing, Queen's University, Kingston, ON, Canada
[2]Radiation Medicine Program, Princess Margaret Cancer Centre, University Health Network, Toronto, ON, Canada
[3]Department of Radiation Oncology, Massachusetts General Hospital, Boston, MA, USA
[4]Department of Physics, Engineering Physics, and Astrophysics, Queen's University, Kingston, Ontario, Canada
[5]Techna Institute for the Advancement of Technology for Health, University Health Network, Toronto, ON, Canada

Abstract— **Radiation therapy (RT) is a common treatment option for a wide variety of cancer types. Despite significant improvements in this technique over the past years, software tools for research in RT are limited to either expensive, closed, proprietary applications or heterogeneous sets of open-source software packages with limited scope, reliability, and user support. Our SlicerRT toolkit aspires to overcome these limitations by providing an extensive set of RT research tools leveraging the advanced visualization and image analysis features of its base platform 3D Slicer.**

The SlicerRT toolkit comprises of a set of 3D Slicer extensions: SlicerRT core, Matlab Bridge, Multi-dimensional Data, and Gel Dosimetry. The SlicerRT core extension contains 26 modules, many of which provide common RT tools used in most RT research scenarios. Matlab Bridge provides a convenient way for connecting the researchers' existing MATLAB algorithms to the SlicerRT ecosystem. Multi-dimensional Data offers a feature set for handling multi-dimensional datasets, such as longitudinal studies or 4D data. Finally, Gel Dosimetry facilitates gel dosimetry analysis workflows through a streamlined, workflow-based end-user application. It serves as an example and proof of concept for such applications implementing advanced clinical or research workflows.

Using these open-source software tools makes it possible to conduct cutting edge RT research without parallel development efforts. It acts as a medium into which researchers can integrate their methods into, and which they can use to perform comparative validation, develop novel RT techniques, or transition advanced methods into routine clinical practice.

Keywords— **Radiation therapy, 3D Slicer, DICOM-RT, dosimetry, adaptive radiation therapy**

I. Introduction

There are numerous aspects of radiation therapy (RT) that have a potential to be improved upon by different techniques. Examples for dose optimization techniques include adaptive radiation therapy (ART) that aims to improve radiation delivery accuracy over the treatment fractions by modifying the daily plans according to the current situation, image-guided radiation therapy (IGRT) that also addresses the same issue by geometrically aligning the patient at each fraction, and a variety of dosimetry techniques that make sure the delivered radiation is as similar as possible to the treatment plan. Other, mostly algorithmic, data processing techniques include automatic image segmentations and registrations, comparison methods for images and structures, and data handling mechanisms. These algorithmic methods enable RT researchers to perform the above mentioned dose optimization and safety-related techniques.

Various software packages exist that provide the necessary features for enabling researchers to employ an existing experimental technique or to investigate a novel one. These can be divided into two groups: (1) commercial treatment planning systems (TPS), which are closed, proprietary, mostly expensive applications with fixed workflows and limited scripting abilities; (2) free, mostly open-source in-house toolkits that offer a subset of the needed functions, but often have serious shortcomings in reliability, stability, extensibility, code quality, maintainability, or user documentation and support. Some of the tools that belong in the latter group depend on commercial closed-source software packages (*e.g.* MATLAB), such as CERR [1] and MINERVA [2], while others are standalone packages, such as PLUNC [3], dicompyler [4], or MMTCP [5]. These tools often provide the necessary functionality when combined, but the inconvenience of performing such a workflow makes such a research effort quite laborious due to their different data formats and user interfaces. Since none of these in-house toolkits provide all the functions required for a powerful RT research platform, their potential to become a standard platform for RT research is limited.

SlicerRT [6] aims to overcome these limitations and aspires to be a comprehensive, reliable and easy-to-use platform applicable for a wide variety of RT research scenarios.

II. Materials and methods

Main requirements, list of features, and priorities of the core toolkit were identified through a series of consensus discussions with RT researchers, considering the clinically most relevant use cases. The participating researchers are

D.A. Jaffray (ed.), *World Congress on Medical Physics and Biomedical Engineering, June 7-12, 2015, Toronto, Canada*,
IFMBE Proceedings 51, DOI: 10.1007/978-3-319-19387-8_152

members of the Ontario Consortium for Adaptive Intervention in Radiation Oncology (OCAIRO).

The end-user application Gel Dosimetry and the higher level tools Matlab Bridge and Multi-dimensional Data were implemented based on requests from the local and international user base of the core toolkit.

A. Software platform and architecture

We chose 3D Slicer [7] as base platform for our toolkit, as it is a popular, well-maintained, stable, and versatile medical imaging platform and end-user application that contains many of the selected features and provides advanced visualization and image analysis tools. 3D Slicer shows great flexibility in the levels of integration its extensions may employ. Three integration levels are supported: (1) command-line interface (CLI) modules that provide standalone executables operating on a limited set of data types and offering basic user interaction; (2) scripted loadable modules that leverage the flexible python language to exercise most of the internals of the host application; (3) loadable modules written in C++, armed with the full range of functions available in 3D Slicer and its base toolkits. The developer chooses the module type according to the complexity of the interaction needed with the 3D Slicer internals, and based on the performance needs.

SlicerRT contains four extensions that can be easily installed from the Extension Manager in 3D Slicer. The architecture of the toolkit and the host platform can be seen in Fig. 1, and the toolkit logos that the user needs to find in Extension Manager to install them are shown in Fig. 2.

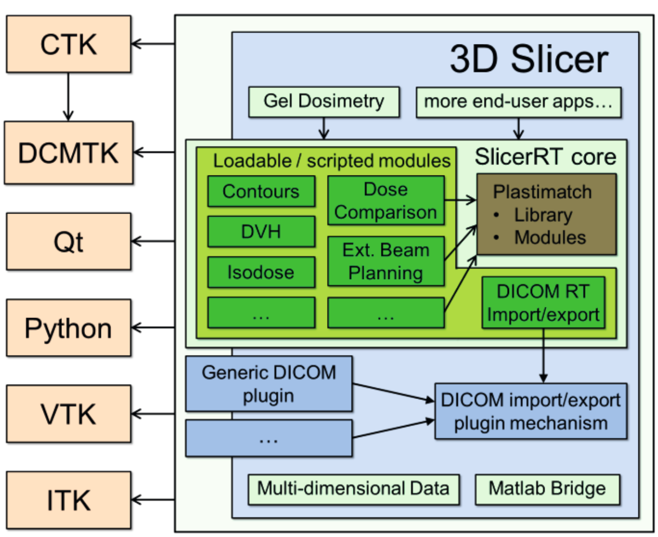

Fig. 1 Architecture of SlicerRT: Four extensions (light green boxes) each containing custom modules added to 3D Slicer

The core toolkit SlicerRT core contains the common RT functions we identified. SlicerRT core depends on the Plastimatch [8] toolkit, which provides additional advanced image registration tools and RT-related functions. SlicerRT core consists of 26 modules which are packaged into an extension. Most of the modules are loadable modules written in C++, and the rest are python scripted modules.

Gel Dosimetry contains one scripted module that launches a *slicelet*, which is a custom user interface for performing a specific workflow in a streamlined way. It is packaged as a separate extension, for which SlicerRT core needs to be installed first.

The other two extensions do not depend on the core toolkit, but provide supplemental functionality for specific user needs. Matlab Bridge contains a CLI and a loadable module, and generates CLI modules for MATLAB-Slicer communication. Multi-dimensional Data is comprised of three loadable and three scripted modules.

Fig. 2 Logos of the SlicerRT extensions. From left to right: SlicerRT core, Matlab Bridge, Multi-dimensional Data, Gel Dosimetry

SlicerRT is open-source, and is distributed under a BSD-style license [9], which means the toolkit can be freely used, extended, or even re-distributed.

B. Development processes

The source code, the documentation, and the test data are stored in a publicly accessible revision control system. This repository is linked to an issue tracking system that allows the developers to keep track of tasks, priorities, and to establish connection between a development effort and the actual committed code. It also enables users to request features or report errors and monitor their status. The toolkit also has a wiki system that contains the user documentation, guides for developers, and high-level project descriptions. The home page for the toolkit can be found at *http://slicerrt.org*.

C. Verification and validation

An integral maintenance concept in our processes is automated testing. Similarly to 3D Slicer and many of its base libraries, SlicerRT defines automated test cases for each module and algorithm, thus decreasing the maintenance overhead. The test system runs the tests each night, and notifies the developer on failures. This way we can make sure that most features are operational at all times.

It is also important to make sure that the algorithms produce valid output data. We have validated [6] our Dose Volume Histogram (DVH) algorithm and implicitly the contour rasterization method against a commercial TPS and CERR, to make sure that such a fundamental tool yields similar results as the other widely used tools.

III. Results

Four 3D Slicer extensions have been implemented, augmenting its immense general medical image computing feature set to support RT research. Each extension can be downloaded from the 3D Slicer Extension Manager. One or more additional modules appear in 3D Slicer after installation that provide the functionality of the installed extension.

SlicerRT core offers numerous common RT research tools that facilitate evaluation of existing procedures or developing new ones. The modules that provide this basic feature set are: *DICOM-RT Import/Export* that provides read/write support for the standard DICOM-RT data format; *Dose Volume Histogram* that allows calculation and visualization of dose volume histogram (DVH) curves and metrics (see Fig. 3); *Isodose* for generating isodose surfaces that can be shown as surface models or planar curves; *Dose Accumulation* allowing weighted summation of dose distributions; *Dose Comparison* that provides the widely used gamma comparison method for dose maps; *DVH Comparison* for quantitative comparison of DVH curves; *Contours* providing a unified storage and conversion mechanism for multiple data representations of RT structures; *Contour Comparison* facilitating quantitative comparison of contours (RT structures) via Dice coefficient and Hausdorff distance; *Contour Morphology* allowing applying logical and morphological operations on contours, for example for adding a target margin; *Plastimatch B-Spline Deformable Registration* and *Plastimatch LANDWARP Landmark Deformable Registration*, adding two powerful registration methods to the registration toolset of 3D Slicer. Numerous SlicerRT core modules are powered by algorithms in Plastimatch. In addition to these modules providing the most common features needed for RT research, SlicerRT core contains several support modules for testing and data handling. The *External Beam Planning* module is a (yet experimental) tool for basic interactive treatment planning, providing an extensible infrastructure for development and evaluation of dose calculation algorithms for photon and proton beams.

Our algorithm validation results show matching with the output of a few widely used research and clinical applications. Validation covers data import, contour conversion, and DVH computation.

Matlab Bridge extension facilitates direct communication between MATLAB and 3D Slicer, allowing RT researchers to use their existing MATLAB algorithms from within 3D Slicer, or use their familiar MATLAB environment to create new algorithms to be used from within the platform. Thus it is possible to leverage the full power of 3D Slicer in terms of visualization and data handling, while making the data transfer transparent.

The *Multi-dimensional Data* extension provides native support for creating, analyzing, and visualizing higher-dimension datasets in 3D Slicer. These datasets may be longitudinal data with numerous time points, models changing in time, sequences of transforms, or even sequences of sequences. Visualization features include manual browsing and continuous real-time replay of sequences in 2D and 3D views. Analysis functions include plotting of voxel intensity changes in time, spatial registration of 4D volumes, basic statistics of structure sets changing in time.

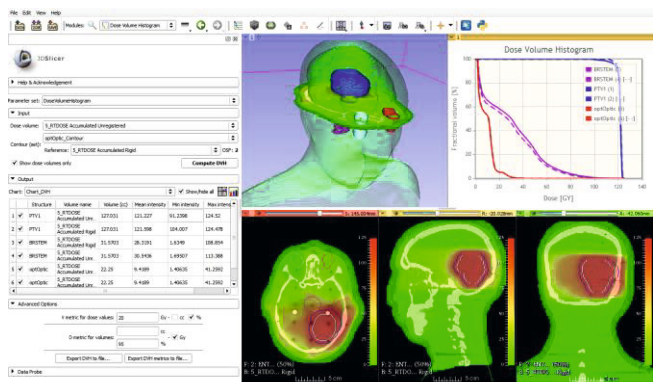

Fig. 3 Appearance of the Dose Volume Histogram module (to the left) and a typical SlicerRT workspace (right) showing a 3D and three 2D views, and a chart view comparing DVHs for dose maps accumulated in different ways (demonstrating the difference between no adaptation and IGRT)

Gel Dosimetry is an end-user application, which implements a streamlined workflow for performing a full gel dosimetry analysis procedure. It is implemented as a *slicelet*, which is a python scripted module that defines a custom user interface (UI) replacing the general UI of 3D Slicer, to allow executing well-defined workflows in a more streamlined manner. This is a very powerful concept, as it bypasses the very rich and complex default user interface, thus reducing required training and execution time and also probability of user error. The extension currently supports an optical CT-based clinical 3D gel dosimetry analysis workflow, but it can be easily extended to support MR-based, 2D film-based or pre-clinical workflows. Our preliminary tests performed in Kingston General Hospital [10] show that compared to the existing workflow, which uses in-house algorithms and several different applications, the execution time was reduced from 3-4 hours to 10-15 minutes, while also making the process less error-prone.

Our SlicerRT toolkit has been adopted at numerous institutions around the world to perform RT research. Approximately 30 groups have contacted us directly or through collaborators with various requests. These projects are typically comparison studies between existing and proposed methods, routine execution of known techniques without relying on commercial software, and implementing and testing open-source dose computation engines or ART procedures. We suppose that many more groups use SlicerRT, but as we do not require registration or reporting of results, we cannot be certain about their numbers.

IV. Future Work

Although the most needed features are already available in SlicerRT core, some of them are not convenient enough to use, in particular, those related to managing contours. Our next goal is to perform a complete overhaul of the current Contours mechanism. The implementation will be more complete, more flexible, and at the same time compatible with DICOM Segmentation Object, which is one of the most important media for image segmentation results.

We will continue to work towards making SlicerRT core easy to use and easy to extend. This effort includes further development of modules that we contributed to 3D Slicer core: the Subject Hierarchy module and infrastructure and the closely related DICOM export mechanism.

As the SlicerRT user base grows, we get more and more valuable feedback that shed light on the shortcomings of the toolkit. Besides addressing these requests, we also continuously work on improving SlicerRT's documentation.

Validation of the algorithmic results is an essential step for showing the correctness and reliability of our own tools. Although we have performed such testing, it only covers a subset of the tools SlicerRT provides. It is also among our goals to perform validation on the rest of the algorithms.

V. Conclusions

The SlicerRT package fulfills the requirements the OCAIRO consortium has set for a common RT research platform. The toolkit is an open-source resource that is developed and maintained according to the highest standards in the industry. It is highly flexible and extensible, thus allowing teams with different focus to also adapt it. The SlicerRT core extension is a downloadable package for 3D Slicer, offering the most common tools needed for RT research that are missing from 3D Slicer base. Matlab Bridge is a vital extension for MATLAB users who would like to keep using their algorithms, but in the same time exploiting the visualization and interactive capabilities of SlicerRT. The Multi-dimensional Data extension allows easy browsing and handling higher-dimensional datasets. Gel Dosimetry is our first SlicerRT-based end-user, which potentially supports a multitude of dosimetry workflows. It serves as a proof of concept for quick prototyping of advanced applications accommodating complex workflows, research or clinical. The SlicerRT team is happy to provide help to new and existing users to create their similar applications, to navigate the user and scripting interface, and to give any answers required to reach their research goals.

This work was supported by Cancer Care Ontario Research Chair, Applied Cancer Research Unit grants, and the OCAIRO consortium.

Conflict of Interest: none

References

1. J. O. Deasy, A. I. Blanco, and V. H. Clark, "CERR: A computational environment for radiotherapy research," Med. Phys. 30(5), 979–985 (2003).
2. Lehmann J et al "Monte Carlo treatment planning for molecular target radiotherapy within the MINERVA system" Phys. Med. Biol. 50 947–58 (2005).
3. https://sites.google.com/site/planunc/
4. https://code.google.com/p/dicompyler/
5. A. Alexander, F. DeBlois, G. Stroian, K. Al-Yahya, E. Heath, and J. Seintjens, "MMCTP: A radiotherapy research environment for Monte Carlo and patient-specific treatment planning," Phys. Med. Biol. 52, N297–N308 (2007).
6. Pinter C, Lasso A et al. "SlicerRT: Radiation therapy research toolkit for 3D Slicer". Medical physics 39(10): 6332-6338 (2012).
7. S. Pieper, M. Halle, and R. Kikinis, "3D SLICER". Proceedings of the 1st IEEE International Symposium on Biomedical Imaging: From Nano to Macro (Brigham and Women's Hospital, Boston, MA) pp. 632–635. (2004).
8. G.C. Sharp,R. Li et al. "Plastimatch: An open source software suite for radiotherapy image processing", Proceedings of the XVIth International Conference on the Use of Computers in Radiotherapy (ICCR) (Amsterdam, the Netherlands) (2010).
9. http://opensource.org/licenses/BSD-2-Clause.
10. K M Alexander et al. "Implementation of an efficient workflow process for gel dosimetry using 3D Slicer" Phys.: Conf. Ser. 573 012042 (2015).

Author: Csaba Pinter
Institute: Queen's University
Street: 25 Union St.
City: Kingston
Country: Canada
Email: csaba.pinter@queensu.ca