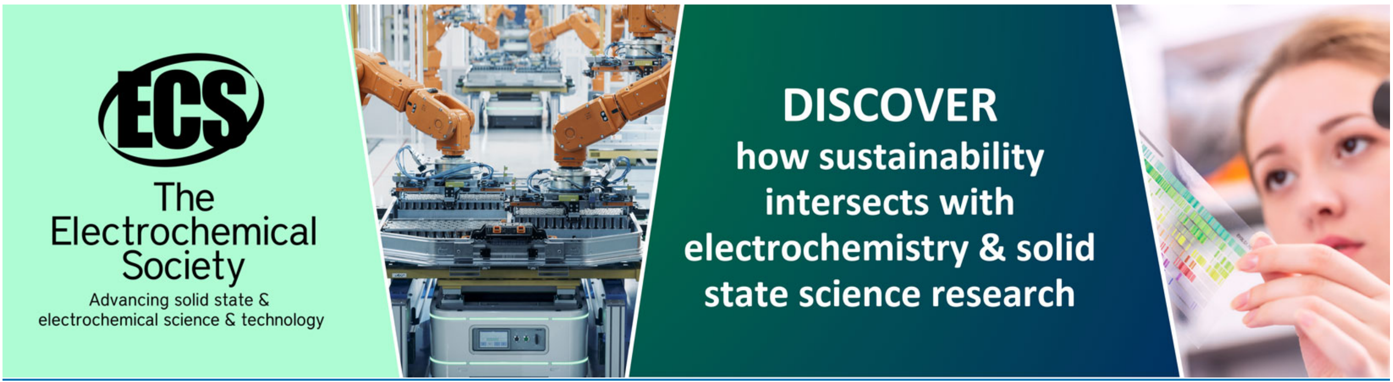
Journal of Physics: Conference Series

The Reconstruction Toolkit (RTK), an open-source cone-beam CT reconstruction toolkit based on the Insight Toolkit (ITK)

To cite this article: S Rit *et al* 2014 *J. Phys.: Conf. Ser.* **489** 012079

View the article online for updates and enhancements.


ECS
The Electrochemical Society
Advancing solid state & electrochemical science & technology
DISCOVER
how sustainability intersects with electrochemistry & solid state science research

The Reconstruction Toolkit (RTK), an open-source cone-beam CT reconstruction toolkit based on the Insight Toolkit (ITK)

S Rit[1], M Vila Oliva[1,2], S Brousmiche[2], R Labarbe[2,3], D Sarrut[1], G C Sharp[4]

[1]Université de Lyon, CREATIS; CNRS UMR5220; Inserm U1044; INSA-Lyon; Université Lyon 1; Centre Léon Bérard, France
[2]iMagX project, ICTEAM Institute, Université catholique de Louvain, Louvain-la-Neuve, Belgium
[3]IBA, B-1348, Louvain-la-Neuve, Belgium,
[4]Department of Radiation Oncology, Massachusetts General Hospital, Boston, MA 02114, USA

E-mail: simon.rit@creatis.insa-lyon.fr

Abstract. We propose the Reconstruction Toolkit (RTK, http://www.openrtk.org), an open-source toolkit for fast cone-beam CT reconstruction, based on the Insight Toolkit (ITK) and using GPU code extracted from Plastimatch. RTK is developed by an open consortium (see affiliations) under the non-contaminating Apache 2.0 license. The quality of the platform is daily checked with regression tests in partnership with Kitware, the company supporting ITK. Several features are already available: Elekta, Varian and IBA inputs, multi-threaded Feldkamp-David-Kress reconstruction on CPU and GPU, Parker short scan weighting, multi-threaded CPU and GPU forward projectors, etc. Each feature is either accessible through command line tools or C++ classes that can be included in independent software. A MIDAS community has been opened to share CatPhan datasets of several vendors (Elekta, Varian and IBA). RTK will be used in the upcoming cone-beam CT scanner developed by IBA for proton therapy rooms. Many features are under development: new input format support, iterative reconstruction, hybrid Monte Carlo / deterministic CBCT simulation, etc. RTK has been built to freely share tomographic reconstruction developments between researchers and is open for new contributions.

Cone-beam CT scanners are used worlwide for guiding radiation therapy since it has been proposed to mount them on the gantry of linear accelerators [1]. Several commercial solutions are available today based on this principle and it is currently being extended to gantries for proton therapy. In contrast to diagnostic CT scanners, these scanners allow access to raw data, i.e., the set of 2D projection images and the corresponding acquisition geometry, and many research groups have been applying their new reconstruction algorithms to patient data acquired for image-guided radiotherapy (IGRT).

A large part of these developments are internal to each research group and are not publically available. An exception is Plastimatch (http://www.plastimatch.org/) which provides fast cone-beam CT reconstruction. Extending Plastimatch to new CT reconstruction algorithms is not straightforward because the development model is not modular. This observation has lead to the launch of the RTK initiative in collaboration with the Plastimatch team.

1. Method

RTK (`http://www.openrtk.org/`) is an open-source and cross-platform toolkit for fast tomographic reconstruction based on the Insight ToolKit (ITK). RTK is developed by the RTK consortium under the same Apache 2.0 license as ITK v4, which is a non contaminating BSD-like license approved by the Open Source Initiative. The current members of the RTK consortium are CREATIS, the Massachusset General Hospital, the Université catholique de Louvain and IBA. The members are bound by a gentleman's agreement available from the website.

1.1. Development tools

RTK is developed in collaboration with Kitware and uses the software process suite: CMake, CDash, CTest, etc. The source repository is hosted on GitHub. The user's documentation is available on the RTK wiki and the developer's documentation uses Doxygen. Tests and examples datasets are hosted on MIDAS, a Kitware open-source server for managing data. The quality of the platform is daily checked with regression tests.

1.2. Dependencies

Insight Toolkit (ITK) RTK is based on ITK v3.20 or v4.3 which is its only mandatory dependency. Many ITK features are essential to the functioning of RTK: input-output factories, filters pipelining, streaming, mini-pipelines, multi-threading, smart pointers, FFTW wrapping, etc. Every RTK filter is meant to be ITK compliant and RTK can be thought of as an ITK module.

Gengetopt (`http://www.gnu.org/software/gengetopt/gengetopt.html`) RTK also provides a set of command-line applications that use Gengetopt to define their options. Gengetopt is a tool to generate command line option parsing code for C programs.

GPU: CUDA and OpenCL RTK proposes CUDA implementation of a few time-consuming filters. Part of the code has been taken and adapted from Plastimatch and NiftyRec (`http://sourceforge.net/projects/niftyrec/`), two other open-source tomographic packages. RTK also proposes OpenCL implementation of one filter but less efforts have been put so far in using OpenCL.

1.3. Features

3D circular geometry RTK has yet been developed for the 3D cone-beam CT geometry only, i.e., a source and a flat panel rotating along a circle. Deviations from this strict geometry can be described for each projection image by means of nine degrees of freedom in total: 3 coordinates for the position of the punctual source, 3 coordinates for the flat panel position and 3 angles for the flat panel orientation. RTK provides classes to read and write the geometry in its own XML file format as well as readers for the geometry of several commercial scanners.

Input and Output file formats RTK uses the ITK factories for the input and output of projection images. As for the geometry, we also provide new image readers for several commercial scanners. Simple conversion from raw data to linear attenuation is provided which we aim at improving in the future with calibration, beam hardening correction and scatter correction to reach quantitative CT, i.e., pixel values in Hounsfield units.

Forward- and back-projectors At the core of tomographic reconstruction are forward- and back-projectors. They have been designed in a typical object-oriented fashion to allow testing of various implementations. GPU versions are also provided for faster reconstruction.

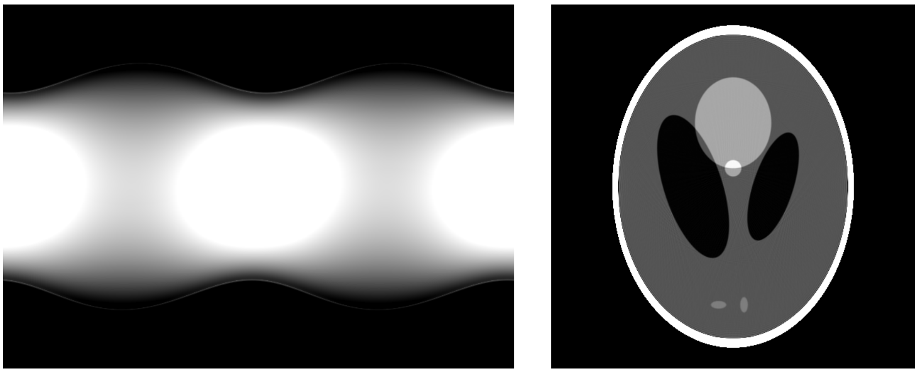


Figure 1. Slices of the 3D simulated sinogram (left) and the corresponding CT image (right) of the Shepp-Logan phantom obtained with RTK.

Reconstruction algorithms The main reconstruction algorithm that is currently implemented is the Feldkamp-David-Kress (FDK) algorithm [2]. Weighting of the projection images prior to FDK reconstruction is available to handle short scans [3] and displaced detector scans for the reconstruction of larger field-of-views [4]. There is also an implementation of the Simultaneous Algebraic Reconstruction Technique (SART) [5] and other iterative algorithms should follow.

Numerical phantoms RTK can simulate projection images and reference CT images of geometric numerical phantoms such as the Shepp-Logan phantom (Figure 1). These phantoms are the most commonly used to validate tomographic reconstruction algorithms. They are also used in RTK functional tests that are nightly run on the dashboard.

2. Results

2.1. Numerical phantom

Figure 1 illustrates RTK ability to simulate sinograms and reference CT images of the Shepp Logan phantom. Examples of reconstruction times are provided in Table 1.

Table 1. Multi-threaded CPU vs. CUDA GPU timing of several RTK filters. The volume size was 512^3 voxels and 512 projection images with 512^2 pixels were used. The measurement excludes hard drive input/output but includes the required transfers between the RAM and the memory of the GPU. The tests have been peformed with 48 threads on a cluster node equipped with two Intel Xeon E5-2630L CPUs and one Nvidia Tesla M2090 GPU.

	CPU	GPU
FDK ramp filtering	83 s (Vnl) 17 s (FFTW)	3 s (CuFFT)
FDK backprojection	110 s	22 s
Forward projection	84 s	17 s

2.2. Catphan acquisitions

The axial slice of the CT image of a quality assurance phantom is provided for real projection images acquired on 3 different cone-beam CT scanners: Elekta Synergy, Varian OBI and IBA

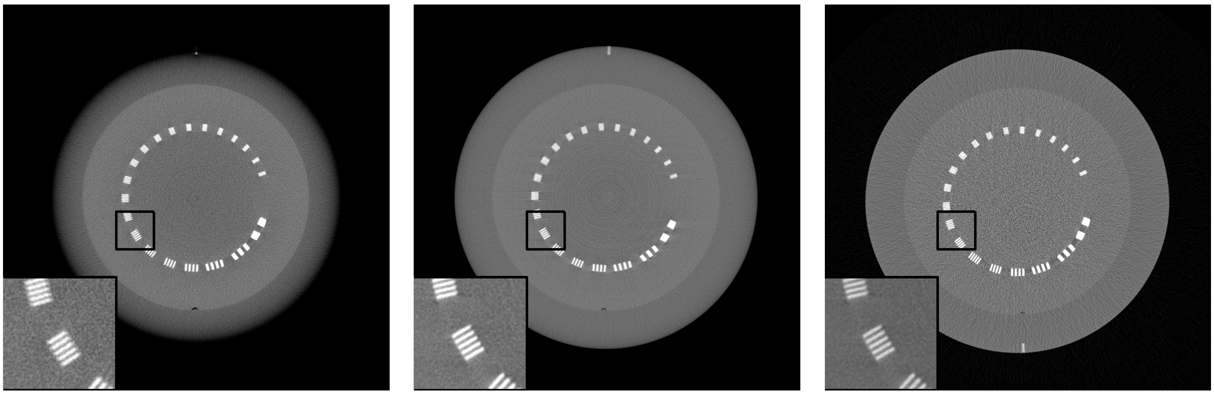


Figure 2. Axial slice with a zoom of cone-beam CT images of the CTP 528 resolution module of Catphan phantoms, reconstructed with the RTK implementation of the FDK algorithm from raw data acquired on an Elekta Synergy in France (left), on a Varian OBI in the USA (middle), and on an IBA prototype scanner in Belgium (right). Raw datasets are publicly available.

ImagX testbench for their prototype scanner. The phantom was the Catphan CTP 503 for the first two and the Catphan 600 for the later.

3. Discussion and Conclusion

RTK is a software toolkit and its code is meant to be used in other platforms. We have already used it to develop proton CT [6] and 2D/3D registration in Elastix [7, 8]. We will also use RTK in Gate [9] to enhance its X-ray simulation capabilities. Others have used it via the command line tools [10]. RTK will also be used in the new cone-beam CT scanner for proton therapy that will be sold by IBA.

The development of RTK is on-going and we intend to add many more features, e.g., new forward- and back-projectors, iterative reconstruction algorithms, etc. Other geometries could also be added, for example the 3D parallel geometry that is currently being developed. RTK is open to new users and developers and those who have made a significant contribution to the platform could join the RTK consortium.

Acknowledgments

Funding for this project has been provided by iMagX, a private-public partnership between IBA and the UCLouvain funded by the convention #1017266 with the Walloon Region of Belgium.

References

[1] Jaffray D, Siewerdsen J, Wong J and Martinez A 2002 *Int J Radiat Oncol Biol Phys* **53** 1337–1349

[2] Feldkamp L, Davis L and Kress J 1984 *J Opt Soc Am A* **1** 612–619

[3] Parker D 1982 *Med Phys* **9** 254–257

[4] Wang G 2002 *Med Phys* **29** 1634–1636

[5] Andersen A and Kak A 1984 *Ultrason Imaging* **6** 81–94

[6] Rit S, Dedes G, Freud N, Sarrut D and Létang J 2013 *Med Phys* **40** 031103 (pages 9)

[7] Klein S, Staring M, Murphy K, Viergever M and Pluim J 2010 *IEEE Trans Med Imaging* **29** 196–205

[8] Delmon V, Vandemeulebroucke J, Pinho R, Vila Oliva M, Sarrut D and Rit S 2013 *International Conference on the Use of Computers in Radiation Therapy (ICCR)* (Melbourne, Australia)

[9] Jan S, Benoit D, Becheva E, Carlier T, Cassol F, Descourt P, Frisson T, Grevillot L, Guigues L, Maigne L, Morel C, Perrot Y, Rehfeld N, Sarrut D, Schaart D, Stute S, Pietrzyk U, Visvikis D, Zahra N and Buvat I 2011 *Phys Med Biol* **56** 881–901

[10] Martin J, McClelland J, Yip C, Thomas C, Hartill C, Ahmad S, O'Brien R, Meir I, Landau D and Hawkes D 2013 *Phys Med Biol* **58** 1809–1822