

# Recent Progress in STIR 5.0

Ander Biguri, Palak Wadhwa, Daniel Deidda, *Member, IEEE*, Georg Schramm, Kuan-Hao Su, Charles W. Stearns, *Fellow, IEEE*, Robert Twyman, *Student Member, IEEE*, Evgueni Ovtchinnikov and Kris Thielemans, *Senior Member, IEEE*

**Abstract**—STIR is an open source software for Emission Tomography data manipulation and image reconstruction, covering both PET and SPECT. In this work recent additions to the STIR code base are highlighted, namely the ability to read General Electric (GE) Raw Data Format 9 (RDF9) files, incorporation of GPU operators for forward and back projection, as well as work towards quantitative imaging for both PET and SPECT.

**Index Terms**—STIR, PET-CT, PET-MR, file formats, image reconstruction, GPU

## I. INTRODUCTION

STIR (*Software for Tomographic Image Reconstruction*)[1] is an open source software (OSS) for PET imaging that contains utilities for data manipulation and image reconstruction. The code base of STIR is in constant update and currently STIR has reached version 4.1. This work highlights three of the many features have been added recently into STIR: GE RDF9 data format readers, GPU projectors and calibration factors for quantitative imaging.

## II. METHODS

### A. GE RDF9 Data Format

PET data from scanners from GE Healthcare are stored in the GE-specific Raw Data Format (RDF) format. Since RDF9, this format is based on the Hierarchical Data Format HDF5. In previous work [2] we reported on the extension of STIR to be able to process data from the GE Signa PET/MR. This capability has now been extended towards GE PET/CT scanners.

Data are exported from the scanner in DICOM format. RDF files can be extracted using the `pet-rd-tools` OSS [3].

STIR uses the HDF5 library for accessing the data fields, including scanner information, acquisition timing etc. STIR 4.1 is able to read (uncompressed) list mode data and (uncompressed)

Manuscript received November 13, 2021. This work is supported by the UK EPSRC under grants EP/T026693/1, EP/M022587/1 and EP/P022200/1, by GE Healthcare and by the UK National Physical Laboratory through the National Measurement System and the UK STFC. We gratefully acknowledge assistance from Michael Spohr (GEHC) for the phantom data.

A. Biguri, R. Twyman, D. Deidda and K. Thielemans are with the Institute of Nuclear Medicine, University College London, UK

K. Thielemans is also with the Centre for Medical Image Computing, University College London, UK

P. Wadhwa was with the Biomedical Imaging Science Department, University of Leeds, UK

K. H. Su and C. W. Stearns are with Molecular Imaging and Computed Tomography Engineering, GE Healthcare, USA

D. Deidda is with the National Physical Laboratory, UK

G. Schramm is with the Department of Imaging and Pathology, Division of Nuclear Medicine, KU Leuven, Belgium

E. Ovtchinnikov is with the Scientific Technology Facilities Council, UK

sinogram data, as well as the normalisation file. In addition, singles per crystal can also be read. The latter are used for constructing a randoms estimate, as GE scanners normally no longer acquire delayed coincidences. At the time of writing, dead-time information (while present in the raw data file) is not yet read or processed.

STIR also provides tools to convert CT DICOM images to mu-maps suitable for attenuation and scatter correction. Work is in progress to read GE PET Image for Attenuation (PIFA) files suitable for MRAC.

The modular nature of STIR allows the user to read data directly from the RDF9 file without extra conversion tools. All STIR processes including scatter estimation, image reconstruction and listmode binning or reconstruction therefore directly work on RDF9 data.

### B. Calibration

With the aim of improving support to quantitative imaging, especially in dosimetry application, a new derived class, `BinNormalisationWithCalibration` was added. This class allows the application of a calibration factor that may be stored in the raw data information in a generic way. The strategy used to read such information however, needs to be different according to modality and vendor. For example, for SPECT sinograms the calibration factor can be read in the DICOM header, while for PET data from the Siemens mMR it is found in the normalisation header. In addition, since not all SPECT scanners are quantitative the option of setting a calibration factor was added in the reconstruction parameter file. An important step to allow the propagation of this information from the data to the normalisation class is the extension of the object `ExamInfo` which now contains a calibration factor member and its set/get functions.

The class `BinNormalisationWithCalibration` also allows to apply the branching ratio factor to PET data. This information depends on the radionuclide used in that specific acquisition, for this reason a database was created for the most used radionuclides. The database, in JSON format, is installed as a configuration file and contains information about branching ratio, isotope name, half-life, and modality, which are taken from standardised data on the "laboratoire national Henri Becquerel" (LNHB) website, with the relative uncertainties.

Therefore, when reading the data, STIR will automatically associate the isotope name with its characteristics. This is particularly useful for decay correction. The `BinNormalisationSPECT`, applies the decay factors per detector head position, whereas a slight modification still needs to be introduced for static scanners (including PET).

### C. GPU Projectors

PET acquisition modelling is generally computed as

$$y_{\text{estimated}} = \epsilon a G \lambda + b$$

with  $\epsilon$  the detection efficiency,  $a$  attenuation,  $G$  a projection matrix,  $b$  a background term incorporating randoms and scatter and  $\lambda$  the current image estimate. The size of the matrix  $G$  prevents storing it in memory, even for modern computers. STIR allows for two different strategies: computation of the matrix with storing in memory exploiting symmetries and on-the-fly projection. STIR already contained several methods for these strategies, including a matrix calculation based on Siddon's method (with optionally multiply rays). These implementation are executed on CPU using multithreading via OpenMP.

With the advent of graphical processing units (GPUs) and their cheap yet powerful parallel capabilities, the ability to perform massively parallel computations is more commonplace in personal computers and this has been exploited to create PET projector operators[4]. STIR 4.1 added the capability to call the NiftyPET[5] GPU projector for the Siemens mMR. In addition, STIR now implements the projectors available as `parallelproj`[6], which uses Joseph's method [7]. The code will parallelize on all GPUs visible in the node where it is running. All geometry information is computed by STIR and this projector can therefore be used for any scanner supported by STIR. However, some modifications are still required for the recent block-detector geometry additions [8].

### III. RESULTS

In this section, we illustrate some of these new capabilities using data of a torso-like phantom scanned on a GE Discovery MI 3-ring, and stored in RDF9 data format. The body-shaped phantom has a "liver" and "lung" inserts, as well as the capability to add several spheres in the lung. In the current acquisition, the lungs were filled with (cold) polystyrene balls. The activity ratios for background: liver: lesion were 1 : 2 : 4. Data were acquired in listmode for 3600s. 660M prompts were acquired with a randoms-to-true ratio of 0.30.

Fig. 1 shows example non-TOF reconstructions using Siddon's ray tracing on a CPU and Joseph's ray tracing on a GPU. The images were reconstructed by unlisting listmode data, calculating randoms from single events, correcting for geometric and detector normalization factors and computing single scatter estimations. The reconstruction incorporated randoms, normalisation, scatter and attenuation correction, all computed in STIR. The images are  $53 \times 305 \times 305$  voxels of  $2.76 \times 2.2 \times 2.2$  mm per voxel. ROI mean values relative to the background mean values for background : liver : lung : spine were 1 : 1.998 : 0.056 : 0.008. Reconstruction has been performed using OSEM, but other reconstruction algorithms are available in STIR.

Example of reconstruction computational times for the same scanner and image size are reported in Table III for CPU (AMD Ryzen 9 3950X) with ray tracing and for GPU using an Nvidia RTX 2080. GPU times are currently slower in comparison for OSEM because, due to limitations in the STIR design, GPU projectors compute full sinograms, even when only a subset is required, thus computation is being wasted.

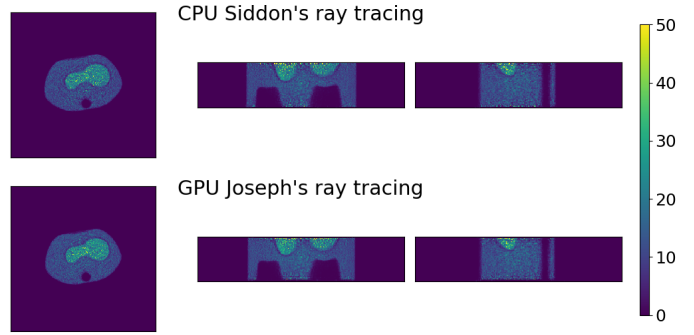


Fig. 1. OSEM reconstruction using 17 subsets and 140 subiterations (no post-filtering has been applied) for both CPU ray tracing functionality (5 rays per bin) (top) and the GPU accelerated Joseph's projector (bottom).

TABLE I  
MLEM (14 ITERATIONS) AND OSEM (17 SUBSETS 238 SUBITERATIONS)  
TOTAL RECONSTRUCTION TIME (IN SECONDS) ON CPU (WITH OPENMP)  
VS GPU.

	OSEM	MLEM
CPU 8 threads	685	655
CPU 16 threads	400	337
CPU 28 threads	321	248
GPU	531	74

### IV. DISCUSSION AND FUTURE WORK

This work presents some of the new features in STIR that are of general interest for PET and SPECT researchers on image reconstruction. Other items include speeding-up the reading of GATE ROOT files and various small improvements to make the connection between GATE and STIR easier, see [9].

STIR forms the basis of the PET support in the Synergistic Image Reconstruction Framework (SIRF) [10]. Most of the new features added to STIR are automatically available in SIRF. SIRF 3.0 added wrapping to the RDF9 features as well as `parallelproj`.

A new version of the RDF data format has been recently released by GE, which is based on the same technology as RDF9. Among other things, RDF10 includes support for seamless inline decompression (with appropriate GE plug-ins), as well as direct random access to blocks of list data based on time stamps. Next steps will add support for the RDF10 format.

GPU usage on subset algorithms is currently wasteful and thus future work will focus on minimizing the unnecessary overhead. The GPU code in `parallelproj` also support listmode and time of flight projections, but this functionality remains to be exported to STIR.

### REFERENCES

- [1] K. Thielemans *et al.*, "STIR: Software for tomographic image reconstruction release 2," *Phys. Med. Biol.*, vol. 57, no. 4, pp. 867–883, Jan. 2012, Publisher: IOP Publishing, ISSN: 0031-9155. DOI: 10.1088/0031-9155/57/4/867.
- [2] P. Wadhwa *et al.*, "Implementation of image reconstruction for GE SIGNA PET/MR PET data in the STIR library," in *2018 IEEE (NSS/MIC)*, ISSN: 2577-0829, Nov. 2018, pp. 1–3. DOI: 10.1109/NSSMIC.2018.8824341.
- [3] B. Thomas *et al.*, *UCL/pet-rd-tools: V2.0.1*, Oct. 26, 2020. DOI: 10.5281/zenodo.4136503.

- [4] P. Després and X. Jia, “A review of GPU-based medical image reconstruction,” *Physica Medica*, vol. 42, pp. 76–92, Oct. 1, 2017, ISSN: 1120-1797. DOI: 10.1016/j.ejmp.2017.07.024.
- [5] P. J. Markiewicz *et al.*, “NiftyPET: A high-throughput software platform for high quantitative accuracy and precision PET imaging and analysis,” *Neuroinform*, vol. 16, no. 1, pp. 95–115, Jan. 1, 2018, ISSN: 1559-0089. DOI: 10.1007/s12021-017-9352-y.
- [6] G. Schramm, *Parallelproj*, [github.com/gschramm/parallelproj](https://github.com/gschramm/parallelproj), May 4, 2021.
- [7] P. M. Joseph, “An improved algorithm for reprojecting rays through pixel images,” *IEEE Transactions on Medical Imaging*, vol. 1, no. 3, pp. 192–196, 1982. DOI: 10.1109/TMI.1982.4307572.
- [8] P. Khateri *et al.*, “Implementation of cylindrical PET scanners with block detector geometry in STIR,” *EJNMMI Physics*, vol. 6, no. 1, p. 15, Jul. 29, 2019, ISSN: 2197-7364. DOI: 10.1186/s40658-019-0248-9.
- [9] R. Twyman *et al.*, “A demonstration of STIR-GATE-connection,” in *submitted, 2021 IEEE NSS/MIC*, 2021.
- [10] E. Ovtchinnikov *et al.*, “SIRF: Synergistic image reconstruction framework,” *Comput. Phys. Commun.*, vol. 249, p. 107087, 2020, ISSN: 0010-4655. DOI: <https://doi.org/10.1016/j.cpc.2019.107087>.