Report of the first CCP PETMR hackathon

Logistics

Date: 26-27 Aug 2018

Location: Atlas Visualization Facility, RAL

Attendees: David Atkinson (UCL), Richard Brown (UCL), Casper da Costa-Luis (KCL), Nikos Efthimiou (Hull), Matthias Ehrhardt (Cambridge), Ashley Gillman (CSIRO), Johannes Mayer (PTB), Evgueni Ovtchinnikov (STFC), Edoardo Pasca (STFC), Kris Thielemans (UCL), Ben Thomas (UCL), Palak Wadhwa (Leeds).
Jakob Jorgensen (Manchester) and Daniil Kazantsev (Manchester) - CCPi flagship CoIs - attended the first day.

Visit to Diamond: Dr. Francesco Maccherozzi, beamline scientist at i06, gave a quick tour of Diamond, touching basics of synchrotron physics and experimental techniques.

Code-setup and mechanisms

- Created Hackathon-SIRF, Hackathon-SIRF_SuperBuild and Hackathon-STIR repos on https://github.com/CCPPETMR
- Attendees were given write access
- Projects set-up
- Create issues inside Hackathon repos
- Initially, suggested to use one branch per topic. Later modified to have one branch per feature (potentially merging to topic branch)
- No intention to create “clean” code, but to get something working
- People were divided in 3 groups (one per topic) with Evgueni and Kris talking to all.

Topics

Adding extra algorithms

Group members: Matthias Ehrhardt, Edoardo Pasca

The aim of this group was to be able to implement new reconstruction algorithms, in Python using the existing CCPi Reconstruction Framework. Matthias has developed a reconstruction algorithm for PET developed in ODL, which he ported to CIL.

First of all this only applies to Python.

Given that CCPi Framework has adhered to naming conventions of SIRF it should be rather easy to use SIRF object within the CCPi optimisation package (CIL). However in the CIL, the optimisation algorithms are totally unaware of the details of AcquisitionData/ImageData, as they are general algorithms.

The optimisation algorithms need to be able

- to perform forward and back projection using a class named Operator.

· data fidelity functions, regularisers are held within the Function class, needs eval, gradient and proximal .
· DataContainer algebra and some other operation (pow, sqrt…)

It was decided that:
1) The CCPi-Framework and CCPi-Regularisation-Toolkit should be added to the SuperBuild (optionally). DONE
2) The AcquisitionModel class should be used instead of Operator. We added a number of methods to the AcquisitionModel for this reason. (This is partially implemented for PET and MR. It's not possible to operate on subsets)

Adding geometrical info to SIRF images

Group members: David Atkinson, Casper da Costa-Luis, Ashley Gillman, Johannes Mayer, Evgueni Ovtchinnikov, Richard Brown

The aim of this project is to have SIRF "aware" of image geometry. In essence, this means the ability for a SIRF image to be able to identify the position of a given voxel in DICOM LPS space. More broadly, this will allow:

- Resampling of MR and PET data between spaces, thereby allowing propagation of information from one to another in synergistic reconstructions.
- Reconstructions will be in the same space as vendor reconstructions, allowing comparisons.
- Information from other image processing from vendor or other-modality reconstructions. For example, motion estimates can be modelled via registration on vendor or Gadgetron MR reconstructions, and applied in PET reconstruction.

Over the two days of the hackathon, some preliminary code was written, but the predominant activity was planning what implementation would adequately solve the current problems, and tasks should be delegated to which classes within the SIRF, STIR and Gadgetron code bases.

Currently, a class implementing this functionality has been written in SIRF, and is used by the PETImageData class. A new class hierarchy will also be implemented. A plan for implementation in STIR has been formulated. Current plans for geometry in Gadgetron are still in infancy.

Progressing GE Signa support in STIR

Group members: Nikos Efthimiou, Kris Thielemans, Ben Thomas, Palak Wadhwa

This project is aimed at including GE SIGNA IO in STIR. To support GE SIGNA PET reconstruction with STIR, STIR library needs to have classes to read HDF5 files and write it in STIR interfile format. This should allow:

- Reading projection data from uncompressed RDF projection data which stores time-of-flight viewgrams of size 1981x27x357 in the HDF5 tag called "/SegmentData/Segment2/3D TOF Sinogram/viewn" where n is the view number. Scanner view number starts from 1 and there are 224 views. Then, saving the particular selection of the entire dataspace for axial positions corresponding to segment numbers for STIR-based viewgram, which would eventually output projection data in STIR space to allow PET image reconstruction directly from scanner data.
- Reading geometrical correction factors from 'geo' file that stores these factors in HDF5 tag called "/SegmentData/Segment4/3D Norm correction/slicem" where m corresponds to view numbers 1 to 16. This block of dataset corresponds to views for one module. These factors

must be repeated 14 times to get the entire projection data and applied to normalisation data.

- During the hackathon, we decided to shift all the HDF5 based functionalities in an IO class called HDF5wrapper.cxx. This is implemented now and this class now contains functionalities to read listmode data, singles, efficiency factors and projection data and write it as 1D array of respective sizes. There are individual buildbock classes and utilities that then converts these 1D arrays to STIR-based interfiles.
  The uncompressed sinogram array is read as STIR-based viewgram for respective view and segment numbers with the developed class ProjDataFromGEHDF5.cxx. This is converted and written into STIR projection data with developed utility called conv_GEHDF5_to_interfile which writes out STIR interfile when given uncompressed HDF5 sinogram and Signa template as inputs.
- We discovered that there could be opposite conventions for detector encoding between GE and STIR. This will need testing.

A new application was written during the Hackathon called nm_signa2mu as part of petmr-rd-tools. This application takes the mu-map (patient and hardware) data from the Signa, scales the voxels values to HU, and produces an Interfile header and image pair that is suitable for reconstruction with SIRF/STIR.

Overall evaluation

- Overall very positive feedback.
- 2 day format was good.
- Next hackathon in ~6 months!
- Future adjustments to mechanisms:
  - Hackathon repos worked during the day but now creates difficulties how to proceed.
  - Projects/Issues should remain on main repo to keep discussions in one place
  - Provide writing materials (paper, pens, whiteboards, …)