Report of the third CCP PETMR hackathon

Logistics

Date: 24-26 July 2019

Location: University of Hull

Attendees: Richard Brown (UCL), Evgueni Ovtchinnikov (STFC), Kris Thielemans (UCL), Alexander Whitehead (UCL), Gemma Fardell (STFC), Harry Tsoumpas (Leeds), Nikos Efthimiou (Hull), Casper da Costa-Luis (KCL)

Remote: Edoardo Pasca (STFC)

Aims

Combining SIRF and Machine Learning framework(s)

The hackathon followed the 24th Software meeting where 4 talks were given about use of ML/AI method in the reconstruction pipeline:

Casper da Costa-Luis School of Biomed. Eng. & Im. Sci., KCL, St Thomas' Hospital, London SE1 7EH

Machine Learning for Image Reconstruction:

This talk discusses initial experience with integrating machine learning into an image reconstruction pipeline for PET. I'll include full installation instructions for Ubuntu, and examples using `NiftyPET` and `keras` (in Python 2).

Kuang Gong, Department of Radiology, Massachusetts General Hospital and Harvard Medical School

Software aspects of applying deep learning to PET image reconstruction.

The combination of PET image reconstruction with deep learning, which includes the penalised reconstruction and unrolled reconstruction approaches, will be presented. In specific, implementation details, potential pitfalls and challenges across the process as a whole, will be introduced. In addition, optimisation directions and essential needs with regards to the platform and projectors will be discussed

Olivier Verdier, KTH-Royal institute of technology (Stockholm, Sweden), HVL-Western Norway University of Applied Sciences (Bergen, Norway)

Proof of concept: motion correction with deep learning

We show how deep-learning based registration can be used to decrease noise by motion correction. The software architecture is based on the deep learning framework Voxelmorph (itself based on a layer of Keras atop of TensorFlow) for the registration, and the operator library ODL to compute the alternate MLEM algorithm. I will explain how these are all put together.

Christopher Syben, Pattern Recognition Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU)

PYRO-NN: Python Reconstruction Operators in Neural Networks

The embedding of known operators in neural networks makes it possible to combine the power of deep learning with physics and signal processing. The software-related aspects of this approach are presented on the basis of the PYRO-NN framework. The PYRO-NN Framework brings CT reconstruction operators as CUDAkernels to Tensorflow. We present implementation details and challenges we faced during the development of PYRO-NN.

Topics

Group 1 Nikos, Edoardo

The aim of this group was to try to use the SIRF in an HPC cluster (Edo on STFC's SCARF https://www.scarf.rl.ac.uk). SCARF is a multi purpose cluster with a total of 577 nodes and 12376 cores; 17 nodes have dual K80 GPU cards.

The approach was to utilise the currently available SIRF docker image. However, HPC cluster managers are rarely happy with docker as it will allow root access to the guest and possibly also to the host machine. The singularity project (https://sylabs.io/docs/) aims at bridging this gap.

Singularity seems to behave mostly like docker, but the user on the guest will have the same permissions of the user on the host, which is very much why the cluster managers are happy with it. However, this means that the user in a singularity container (presumably) doesn't have root access in the guest and can't install anything in it (as root).

Edo tried to translate the Dockerfile (the description of the guest machine that docker uses to create the container) for singularity. For a number of problems related with version of kernel and other dependencies this way was not successful:

1) Singularity was built on a CentOS6 machine
2) a minimal singularity container was built, but was unable to run as kernel too old.
3) Casper and Nikos built a docker container with all the software needed for ML/AI + SIRF https://hub.docker.com/r/ccppetmr/sirf/tags (service-gpu tag)
4) Edo imported the docker container in singularity on SCARF but was unable to run it as the container pretended to create the `sirfuser` user on the container, which for some reasons is connected with the host, therefore the host tried to add the same user which clearly failed.

```
>> singularity run sirf
ensure sirfuser:sirf,sudo exists
100595: Creating and switching to: sirfuser:sirf (1000:1000)
groupadd: Permission denied.
groupadd: cannot lock /etc/group; try again later.
ensure secondary users exist
useradd: group 'sirf' does not exist
useradd: group 'sirf' does not exist
/usr/local/bin/entrypoint-service.sh: line 21: cd: /home/sirfuser: No
such file or directory
exec /usr/local/bin/service.sh
/ /home/vol05/scarf595
cp: cannot stat
'./INSTALL/share/gadgetron/config/gadgetron.xml.example': No such file
or directory
/home/vol05/scarf595
/usr/local/bin/service.sh: line 42: conda: command not found
```

Group 2 Alex, Harry

Image registration.

Group 3 Gemma, Harry, Edoardo (day 2)

The aim was to use the technique of Kuang to denoise?

Edo wrote an implementation of the ADMM algorithm within the CIL framework. https://github.com/vais-ral/CCPi-Framework/pull/358

Group 4a Casper, Richard, Kris

GPU projector in STIR.

Group 4b Evgueni, Richard, Kris

Gadgetron GPU gadgets.

Two GPU gadgets added to SIRF gadget library (gpuRadialSensePrepGadget, gpuCgSenseGadget).

GPU-using script added to SIRF/examples/Python/MR/Gadgetron new subfolder GPU.