COST AID ASL post-processing Workshop
Toulouse October 2013

This workshop runs thought the post-processing of ASL data using tools from the FMRIB Software Library (www.fmrib.ox.ac.uk.uk/fsl), we will primarily focus on the BASIL tools specifically designed for ASL data.

The ASL tools first appeared in FSL 5.0 and this workshop is best performed using v5.0.4 or later. The majority of the analysis performed here can also be achieved using the graphical user interface (from v5.0.4 onward), at a command prompt type either `Asl` (linux) or `Asl_gui` (OS X).

1. Simple analysis

In the rawdata directory you will find some single delay ASL data: sti_data.nii.gz. This was acquired using the following parameters:

- pcASL labelling
- Post labelling delay = 1.5s
- Tag duration 1.4s
- 2D EPI readout with 24 slices
- Voxels 3.4x3.4 mm, slice thickness 5 mm
- Partial Fourier 7/8ths.
- TR = 4320ms, TE = 24ms
- FOV: 214 mm
- Matrix size: 86
- Bandwidth: 1938 Hz/Px

Note that in this case background suppression, which would customarily be applied, has been turned off.

Firstly, it is a good idea to look at the data, this can be done in FSL using:

```
fslview rawdata/sti_data &
```

It is useful to view the data as a movie and if you do you will see a flickering pattern that comes from the perfusion signal and the fact that the data is a series of tag and control pairs.

In this first set of exercises we are going to do some simple manipulation of the data and a basic first quantification of perfusion. We start by subtracting tag and control images. This is a relatively simple procedure and we could do this manually using various FSL commands (like fslmaths), but asl_file is specifically designed for this task:

```
asl_file --data=rawdata/sti_data --ntis=1 --iaf=tc --diff
  --out=sti_diffdata --mean=sti_diffmean
```

View the difference data (sti_diffdata) as a movie and notice how noisy this datasets is (and that there are some clear artefacts in here). Now view the average difference image (sti_diffmean) and notice that we do have a reasonable looking perfusion image.

Now we will do motion correction. We will keep it simple and use the default motion correction algorithm and apply it to the raw data (this would have been a bit harder if our data had background suppression turned on).

```
mcflirt -in rawdata/sti_data -out sti_data_mc –meanvol
asl_file --data=sti_data_mc --ntis=1 --iaf=tc --diff
  --out=sti_diffdata_mc --mean=sti_diffmean_mc
```

Compare the results pre and post motion correction, you will not see massive differences (most obvious outside the brain) as this subject was fairly stationary.

The average data isn't too bad here, although we do have quite a few volumes in the acquisition. However, we will do some smoothing to improve it a little further:

```
fslmaths sti_diffmean_mc -s 2  sti_diffmean_mc_smooth
```

Now we have a relative perfusion image we want to do some very basic quantification. For this we will use a separate calibration image (calibhead) that was acquired using exactly the same parameters as the ASL data, but with a longer TR (6 s in this case) and no background suppression (assuming that it was applied to the main data).
Throw away the first volume (as it will not have ready a steady state in one TR) and take the mean over the remaining volumes and smooth to match the data:

```
fslroi rawdata/calibhead calib 0 1
fslmaths calib -Tmean calib
fslmaths calib -s 2  calib_smooth
```

We will want a brain mask we create it now from the calibration image:

```
bet calib brain -m
```

Now we quantify the perfusion using the (simplified) formula for pcASL:

$$CBF = \frac{6000 * \lambda \left(M_{control} - M_{label} \right) \exp^{\frac{PLD}{T_{1,blood}}}}{2\alpha T_{1,blood} M_{calib} \left(1 - \exp^{-\frac{\tau}{T_{1,blood}}} \right)} [ml/100g/\min]$$

Putting in $\lambda = 0.9$, $T_{1blood} = 1.65$, $\alpha = 0.85$ and the sequence parameters (PLD = 1.5 and $\tau = 1.4$)

```
fslmaths sti_diffmean_mc_smooth -div calib_smooth -mul 8350
  -mas brain_mask sti_perfusion
```

Thus we have our final perfusion image (sti_perfusion). Note that it might have been a good idea to make sure the ASL data and the calibration image were aligned (to account for any motion between the two), we could have achieved this using a call to flirt.

We could have achieved (almost) the same result here using the control images from the main data since we have background suppression turned off (although with lower accuracy since the TR of the data is quite short compared to T1 for full relaxation and we really should correct for this discrepancy)

```
asl_file --data=rawdata/sti_data --ntis=1 --iaf=tc --spairs
  --mean=sti_static
fslmaths sti_static_odd -s 2  calib_smooth_alt
fslmaths sti_diffmean_mc_smooth -div calib_smooth_alt
```

```
-mul 8350 -mas brain_mask sti_perfusion_alt
```

2. Automated analysis using oxford_asl

Now we are going again to quantify perfusion but use the fully automated oxford_asl tool. Firstly we will run oxford_asl without the M0 calculation stage just to get relative perfusion:

```
oxford_asl -i sti_diffdata -o /sti_result --tis 2.9 --casl
  --bolus 1.4 --bat 1.3 --slicedt 0.065 --artoff –fixbolus
```

Oxford_asl uses the 'standard' model from Buxton *et al*. Magn. Reson. Med. 1998, so we need to supply a BAT value (although the perfusion is relatively insensitive to this choice). The 1.3 s here is based on our experience with this particular sequence. Note that:

- We have defined a TI (inversion time in pASL language), for pcASL this is the bolus duration plus the PLD, i.e. TI = 1.4 + 1.5 = 2.9
- We have specified the bolus duration as set by the sequence: 1.4 s.
- We have included the fact that this is a 2D slice readout and thus higher slices are read 0.065 s later than ones below them.
- We have told oxford_asl that the data is pcASL (--casl) otherwise it will use the pASL model.
- We have specified --artoff, we do not expect any arterial signal to be present in this data since the PLD is fairly long
- We have specified --fixbolus, since the bolus duration is well defined by pcASL (and we cannot hope to infer it from single delay data).

Once this has run look in the results directory sti_result you should find in the native_space subdirectory a (relative) perfusion image and this should look similar to the one we calculated earlier.

What we want is perfusion in absolute units so we need to calculate M0.We will do this talking the CSF as our reference region (rather than the voxelwise approach we adopted earlier). The command to oxford_asl is now:

```
oxford_asl -i sti_diffdata -o /sti_result --tis 2.9 --casl
  --bolus 1.4 --bat 1.3 --slicedt 0.065 --artoff --fixbolus
  -c rawdata/calibhead --cref rawdata/calibbody --tr 6 --csf
  extras/csfmask
```

We are supplying the same calibration image (with TR=6 s) as before, we also have an extra 'reference' image acquired using the body coil to help correct for the variations in sensitivity of the head coil. Here we have given oxford_asl a predefined mask of the CSF to do its calculation, this was actually automatically created by oxford_asl in a previous analysis and we will see how to ask it for this later.

You should now find that in sti_result/native_space the (relative) perfusion image has been joined by a perfusion_calib image that is in absolute units. In sti_result there is also a new subdirectory containing the results of the calibration including the M0 value and the sensitivity image

A further thing we might like to do is smooth our noisy perfusion image. Previously we did this manually by specifying the degree of smoothing, how can we be sure we choose the right value? Oxford_asl will attempt to determine the

right degree of smoothing directly from the data and apply it adaptively, we just need to add the --spatial option:

```
oxford_asl -i sti_diffdata -o /sti_result --tis 2.9 --casl
  --bolus 1.4 --bat 1.3 --slicedt 0.065 --artoff --fixbolus
  --spatial -c rawdata/calibhead --cref rawdata/calibbody
  --tr 6 --csf extras/csfmask
```

In general you will probably want oxford_asl to generate the CSF mask automatically for which you will need a structural image for segmentation. If you supply a structural image you also have the opportunity to register the results to structural space to use in further group analysis. The first steps to perform are to extract the brain in the structural image and also do the same for the raw ASL data, as we will use this as the basis for registration (the difference data is generally not very good as a basis for registration).

```
fslmaths rawdata/sti_data -Tmean sti_mean
bet rawdata/struct struct_brain
bet sti_mean sti_mean_brain
```

Now we can feed these images into oxford_asl

```
oxford_asl -i sti_diffdata -o /sti_result --tis 2.9 --casl
  --bolus 1.4 --bat 1.3 --slicedt 0.065 --artoff --fixbolus
  -c rawdata/calibhead --cref rawdata/calibbody --tr 6
  -s struct_brain --regfrom sti_mean_brain –report
```

The extra command we have added is --report this asks oxford_asl to calculate perfusion within both GM and WM based on the segmentation of the structural image (and the partial volume threshold of 0.8).
If we hadn't had the calibration 'reference' image we could also have used an estimate of the coil sensitivity from the structural image by supply the --senscorr command.

3. Multi-PLD data

Although single PLD data is easier to handle and is mostly insensitive to arrival time (with suitable choice of PLD), multi-PLD data offers the opportunity to estimate and correct for the effects of arrival time. At the same time that the data above was acquired a further set with the same total number of measurements was collected with a range of PLD: 0.25, 0.5, 0.75, 1.0, 1.25, 1.5. This can be handled by oxford_asl just like the single PLD data, but now it will also give arrival estimates. First we need to tag-control subtract

```
asl_file --data=rawdata/mti_data --ntis=6 --iaf=tc --diff
  --out=mti_diffdata --obf=rpt
```

Now we can run oxford_asl

```
oxford_asl -i mti_diffdata -o /mti_result
  --tis 1.65,1.9,2.15,2.4,2.65,2.9 --casl --bolus 1.4
  --bat 1.3 --slicedt 0.065 --artoff --fixbolus --spatial
  -c rawdata/calibhead --cref rawdata/calibbody --tr 6
  --csf extras/csfmask
```

With this choice of PLDs we can estimate the arrival times reasonably well – we expect the earlier PLDs to be sensitive to arrival time in this data. You might like to compare the perfusion result here to that we got from single PLD data earlier.

Since we have early PLDs we might expect to see some arterial signal which might be reflected in our perfusion estimates. We could try to correct for this by removing the --artoff option, this in turn will also supply an estimate of arterial CBV.

4. Partial volume correction

So far we have been calculating perfusion images ignoring the fact that within the voxels in our images there is a mixture of GM and WM which have markedly different perfusion. Thus our perfusion images are quite reflective of the underlying structure of the brain and not just the local perfusion variations. This will be particularly important when considering patients with morphological changes such as in dementia. We can correct for these partial volume effects using the information on GM and WM proportions from the segmentation of a structural image. Oxford_asl will do this all you need to do is add –pvcorr. Note though this the PV correction will take around 10 times as long to run as the analysis we have done earlier.

For the single PLD data:

```
oxford_asl -i sti_diffdata -o /sti_pv_result --tis 2.9
  --casl --bolus 1.4 --bat 1.3 --slicedt 0.065 --artoff
  --fixbolus -c rawdata/calibhead --cref rawdata/calibbody
  --tr 6 -s struct_brain --regfrom sti_mean_brain --report
  --pvcorr
```

For the multi-PLD:

```
oxford_asl -i mti_diffdata -o /mti_pv_result
  --tis 1.65,1.9,2.15,2.4,2.65,2.9 --casl --bolus 1.4
  --bat 1.3 --slicedt 0.065 --artoff --fixbolus
  -c rawdata/calibhead --cref rawdata/calibbody --tr 6
  -s struct_brain --regfrom sti_mean_brain --report --pvcorr
```

In the results directories you will now find perfusion (and perfusion_calib) images that are now of the GM perfusion, along with separate images for WM. It is probably more useful to examine the 'masked' images (e.g. perfusion_masked) as this masks out regions where the PV of GM (or WM) are so low that perfusion estimates are not meaningful. Notice that the perfusion values in GM are higher than they were previously, since partial voluming of GM with lower perfusion WM is predominant in ASL images.