<u>Instructions for using Uesugi-san's code for CT reconstruction</u>

Uesugi-san's code runs in DOS – it must be run from the location of the files, and will write to this location, so make sure the code itself is in the executable path (i.e. set the path environment variable appropriately). Reconstruction is a time and disk consuming process! If you use this code, please acknowledge Kentaro Uesugi (Spring-8/JASRI) in any publications etc.

Nomenclature:

- Dark refers to a dark field image – i.e. the shutter was closed
- Flat refers to a full field image – i.e. shutter open, no sample in beam (I)
- Projection = an image of the sample (P)
- Slice = a (horizontal) reconstructed slice, the slice number is therefore the y pixel number in the projection image
- COR = centre of rotation, the x pixel in the image about which the sample is rotating
- # = a number

1. If necessary (and it exists), run *conv.bat*. This batch file is written during data acquisition. It renames files, and creates the dark (dark.img) and flat images needed for image corrections to the projection images. This is usually only required if the images are named a####.img, since the reconstruction algorithm searches for images named q####.img
2. Find the centre of rotation for the start and end slices in the volume to be reconstructed. This can be done by using the hp2do routine:
   hp2do dark.img I_{0deg} P_{0deg} *dark.img* I_{180deg} P_{180deg} > *offset.dat*
   The required files for input are the dark, flat and projection images at zero and 180 degrees. The names of these files can be found by inspecting the output.log text file. They will be the first two and last two files (note that 0 indicates a flat image, whilst 1 indicates a projection). The routine then writes a text file (offset.dat) which contains the centres of rotation (COR) for all slices. These should be tested for accuracy!
3. To do this, reconstruct a slice using:
   reconst slice# COR
4. Alternatively, a batch file can be run which will reconstruct the same slice many times, incrementing the COR each time. The best COR is then found by inspecting the reconstructed images. An indicator of the best COR is the lack of semicircular artefacts in the image (note that if these appear, the difference between the COR used and the correct COR is about half the diameter of the semicircle).
   sfa slice# startCOR# endCOR# [deltaCOR] > output.bat
   Optional *deltaCOR* is the increment (in pixels) of the COR, the default is 1. Run the output.bat file to generate a series of slices.
5. Once the CORs for the start and end slices of the volume have been found, all the slices can be reconstructed in a batch process. Create the batch file:

srec startslice# COR(startslice)# endslice# COR(endslice)# [pixelsize] [theta] > output.bat

The *pixelsize* and *theta* options default to 1 and zero. *pixelsize* (microns) is required to maintain the correct absolute linear attenuation coefficient (LAC) values. If it is not used the values of LAC are only relative. Run the batch file. 16 bit tif format files will be written named rec####.tif where #### is the slice number.

6. Usually, it is desirable to reduce the volume size by converting the 16 bit tifs to 8 bit. To work out the range of intensity values to write to 8 bit dynamic range, run the batch file *tif2hst* on a typical reconstructed slice:

   tif2hst rec####.tif output.txt

   The resulting file is a histogram of the frequency of linear attenuation coefficient (LAC) values in cm^{-1}. Plot column 1 vs column 2 from this text file to chose the range of LAC values for the 8 bit files (I use gnuplot to do this, setting a logarithmic y scale). As a rule of thumb, choose LACmin and LAC max values just above and below those shown in the histogram (e.g. $\pm 0.5cm^{-1}$).

7. Run the batch file *tif_h2o* (i.e. hex to octal):

   tif_h2o recstart# recend# LACmin LACmax

   This will write ro####.tif, which are 8 bit versions of the rec####.tif slices.

Common artefacts and problems (see http://www-fp.mcs.anl.gov/xray-cmt/rivers/tutorial.html for an excellent tutorial by Mark Rivers):

- The reconstruction code crashes. Usually this is because a required file is missing. Check that dark.img exists in particular. The routine *img_ave* can be used to create averaged dark or flat images if required. If projection images are missing, some creative renaming can get around the problem, e.g. adjacent projections can be copied to replace missing ones without being too detrimental to the reconstruction provided the number of projections is high.
- Semicircular artefacts appear on the reconstructions. See 4 above.
- Dark streaks appear on the reconstructions. These are caused by highly attenuating (bright) parts of the sample. There is not much one can do about this as most corrections involve smoothing which is usually detrimental to image resolution (particularly for phase contrast tomography).
- Rings appear on the reconstructed slices. These are due to the uneven response of detector pixels and are minimised by the use of a full field image. Even with good full field correction, they still occur. They can be corrected for by smoothing the sinogram. Beyond the scope of these instructions!

Karen Siu
v.1.1
17 July 2006