Introduction:

The acpcdetect program is a module of the Automatic Registration Toolbox (ART). The program takes a 3D T1-weighted structural MRI of the human brain as input. It automatically detects the mid-sagittal plane (MSP) using the method described in [1]. It then detects the AC and PC intersection points on the MSP using the method described in [2]. Finally, it detects 8 additional landmarks[[1]](#footnote-1) (the so-called Orion landmarks) on the MSP using the method described in [3]. This information is used to tilt-correct the input volume into a standard orientation. In this orientation: (1) the MSP is precisely aligned with the central plane of the FOV; (2) the anterior-posterior (AP) axis is on the MSP and aligned with the AC-PC line; (3) the inferior-superior (IS) axis is on the MSP and perpendicular to the AC-PC line; (4) the left-right (LR) axis is perpendicular to the MSP; and (5) the FOV center is approximately the mid-point between the AC and the PC on the MSP. The FOV center can alternatively be placed on the AC point using the -center-AC option.

Required argument:

-i (-input) <*input-image*>.nii: 3D T1-weighted structural MRI in ‘n+1’ NIFTI1 format of type short or unsigned short

Optional arguments:

-v (-verbose): enables verbose mode

-center-AC: places the FOV center at AC.

-output-orient <orientation code>: specifies the orientation of the output image

-nx <*integer*>: number of voxels in i direction (fastest varying index) of the output image

-ny <*integer*>: number of voxels in j direction (2nd fastest varying index) of the output image

-nz <*integer*>: number of voxels in k direction (slowest varying index) of the output image

-dx <*float*>: voxel size (mm) in i direction (fastest varying index) of the output image

-dy <*float*>: voxel size (mm) in j direction (2nd fastest varying index) of the output image

-dz <*float*>: voxel size (mm) in k direction (slowest varying index) of the output image

-o (-output) <*output-image*>.nii: output image names (default: <input-image>\_<orientation code>.nii

Example 1:

$ acpcdetect -i $ARTHOME/example1/v1.nii -v

Input image: /Users/ardekb01/babak\_lib/example1/v1.nii

Input image orientation: ASL

Input image matrix size: 256 x 256 x 128

Input image voxel size: 1.0000 x 1.0000 x 1.2500

Output image: /Users/ardekb01/babak\_lib/example1/v1\_ASL.nii

Output image matrix size: 256 x 256 x 128

Output image voxel size: 1.0000 x 1.0000 x 1.2500

Output image orientation: ASL

Output transformation matrix: /Users/ardekb01/babak\_lib/example1/v1.mrx

Output transformation matrix (FSL format): /Users/ardekb01/babak\_lib/example1/v1\_FSL.mat

Example 2:

$ acpcdetect -i $ARTHOME/example1/v1.nii –v --output-orient LAI

Input image: /Users/ardekb01/babak\_lib/example1/v1.nii

Input image orientation: ASL

Input image matrix size: 256 x 256 x 128

Input image voxel size: 1.0000 x 1.0000 x 1.2500

Output image: /Users/ardekb01/babak\_lib/example1/v1\_LAI.nii

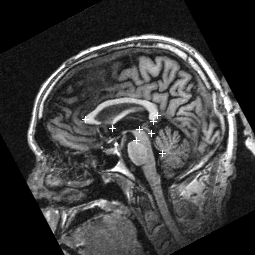
Output image matrix size: 128 x 256 x 256

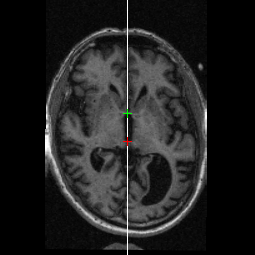
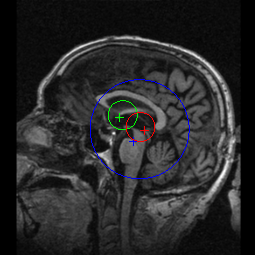
Output image voxel size: 1.2500 x 1.0000 x 1.0000

Output image orientation: LAI

Output transformation matrix: /Users/ardekb01/babak\_lib/example1/v1.mrx

Output transformation matrix (FSL format): /Users/ardekb01/babak\_lib/example1/v1\_FSL.mat



1. Actually, one of the 8 Orion landmarks happens to be very close to the AC. [↑](#footnote-ref-1)