

DBM Toolbox for Neuroimage Data

DBM Toolbox provides algorithms for deformable image registration of 3-D magnetic resonance brain images. The algorithms were implemented as functions and scripts in MATLAB® environment. Some of the functions, which provide computationally intensive tasks, have been compiled into *.mexw64 files for 64 bit Windows operation systems.

The following list of scripts provides execution of deformation-based morphometry including linear registration, deformable registration and the production of the outputs, i.e. statistical parametric maps. Important functions connected to the particular scripts are listed in the table.

| SCRIPT, PURPOSE | DEPENDANCY |
|---|--|
| DBM010ResliceVolumes.m Perform 3D affine transform defined by a NifTI format image in a batch. | SPM.reslice_nii() |
| DBM020VolumesToStx.m Resample images according to affine transformations computed in VBM8 toolbox | cg_vbm8_run() resize_img() |
| DBM025DoMasks.m Create a binary brain mask and a binary head mask from tissue probability maps | cg_morph_vol() |
| DBM030PreprocessSTXData.m Read image data in NIFTI format. The images should have been transformed into stereotaxic space with the use of affine transformations and resampling (VBM8 Toolbox). Mask a reference image (template) with the use the head mask. Perform padding of the images on the dimensions of multiples of 8. Convert the images into float (single) data format. Smooth the images with Gaussian kernels 1 mm, 2mm, 4mm, 8mm and convert the smoothed images into MHA UCHAR format (256 gray levels). | io.load_metainage() io.save_metainage() ITK.smoothingandcasting3d.exe |
| DBM035PreprocessTPMData.m Read the tissue probability maps, perform their padding and prepare their versions for multiresolution analysis. (1mm, 2mm, 4mm, 8mm) | NIFTI.load_nii() padding3D() ITK.smoothing3d.exe |
| DBM050_batch.m A batch for multiple 3-D images registration to one reference image (template ICBM_152). Results of the registration are stored in separate folders for each image. | Framework_highdim3D.m The core script for performing deformable registration of one floating image to a reference image. |
| DBM060Jacobian.m Go through the calculated deformation vector fields and calculate partial derivatives, their Jacobian and logarithm of the Jacobian. | io.load_metainage() io.save_metainage() c_jacobian3D() |
| DBM070StatAncova.m Calculate voxel-wise T statistics from detJ or detJlog with age effect on detJ removed by linear regression. | STAT.ttest2() |
| DBM080StatDoMaps.m Locate clusters of significant voxels in the results of ANCOVA. | STAT.pdf() STAT.icdf() stat_fdr() stat_clusterfiltering_easy() |
| Framework_highdim3D.m The core script for performing deformable registration of one floating image to a reference image. | preprocessImages3Dsimple() jointDistribution3D() ITK.upsampling.exe ITK.spatialmodel.exe |