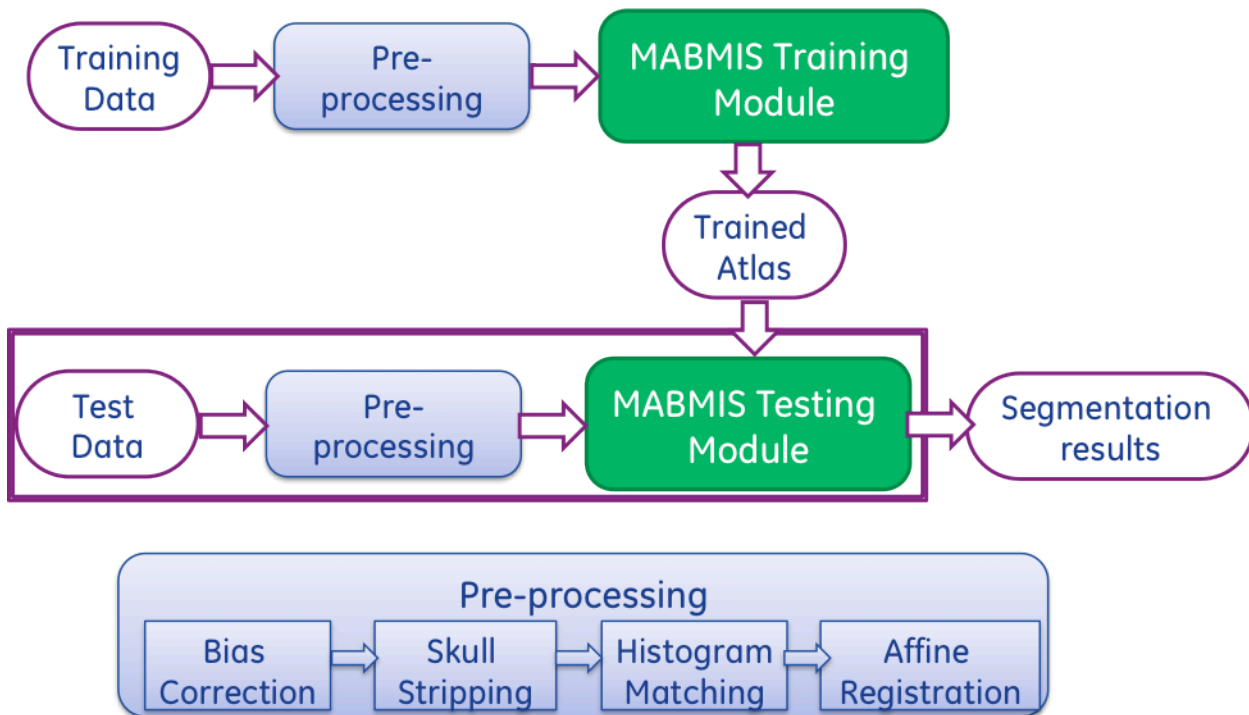


# MABMIS for Slicer Manual

## 1. Introduction

MABMIS is a module for Slicer 4 that implements a multi-atlas based multi-image method for group-wise segmentation [1]. The method utilizes a novel tree-based groupwise registration method for concurrent alignment of both the atlases and the target images, and an iterative groupwise segmentation method for simultaneous consideration of segmentation information propagated from all available images, including the atlases and other newly segmented target images.

## 2. Overview of the Algorithm



**Figure 1 Algorithm overview**

Fig. 1 shows an overview of the algorithm. MABMIS contains two modules: the training module, and the testing module.

In the training module, multiple atlas image pairs are required as the input. Each atlas image pair contains one pre-processed T1-weighted intensity image and one label image on which each interested structure on the T1 image is manually identified and assigned a unique label. The label image is generally generated by an expert. In the training module, the multiple atlas pairs are

processed to construct an atlas tree, where each atlas is one node of the tree. The tree is constructed based on the similarity among the atlases and tree-based groupwise registration.

In the testing module, a group of target images are segmented based on the atlas tree. First, a novel tree-based groupwise registration method is employed to simultaneously register them to the atlas tree, the target images are segmented simultaneously using an iterative groupwise segmentation strategy, which provides improved accuracy and across-image consistency.

Before applying the modules, both atlas images and target images need to be pre-processed. The processing steps include bias correction ( e.g., N4), skull stripping (e.g., skullstripper in slicer), histogram matching. All images are then registered to a common space using affine registration. The pre-processing tools are not included in the module.

Detailed description of the algorithm can be found in [1].

### **3. Installation**

#### **3.1. Required Software Packages**

The following is needed:

- Insight Toolkit (Ver. 4.0 or higher)
- CMake (Ver. 2.8.9 or higher)
- 3D Slicer (Ver. 4.0 or higher)

#### **3.2. Download & Installation**

The source code can be downloaded from github:

<https://github.com/BrainMeasurement/MultiAtlas>

There are two ways to install the pre-compiled module. (The pre-compiled module were build on slicer 4.3).

(1) Using slicer extension manager

The pre-compiled version can be installed to slicer using slicer extension manager. (For the usage of extension manager, refer to: <http://www.slicer.org/slicerWiki/index.php/Documentation/Nightly/SlicerApplication/ExtensionsManager>)

(2) Manual installation

One can also download the pre-compiled module from NITRC site ([http://www.nitrc.org/projects/mabmis\\_slicer4/](http://www.nitrc.org/projects/mabmis_slicer4/)), then manually install it by following the instruction:

[http://www.slicer.org/slicerWiki/index.php/Documentation/Nightly/SlicerApplication/ExtensionsManager#How\\_to\\_manually\\_install\\_an\\_extension\\_.3F](http://www.slicer.org/slicerWiki/index.php/Documentation/Nightly/SlicerApplication/ExtensionsManager#How_to_manually_install_an_extension_.3F)

## 4. User Guide

The module has two parts: training and testing. The training module requires as input a set of atlases, where each atlas contains an intensity image and a pre-segmented label images. The module then generates a trained atlas tree that can be used for multi-structure segmentation in the testing phase. The testing module requires two inputs. The first one is the trained atlas tree, and the second one is the set of images to be segmented. As a result, the testing module outputs multi-structure segmentation results on these images.

The example input xml files are provided in the file “*TestData.zip*”.

### 4.1. Training

#### 4.1.1 GUI

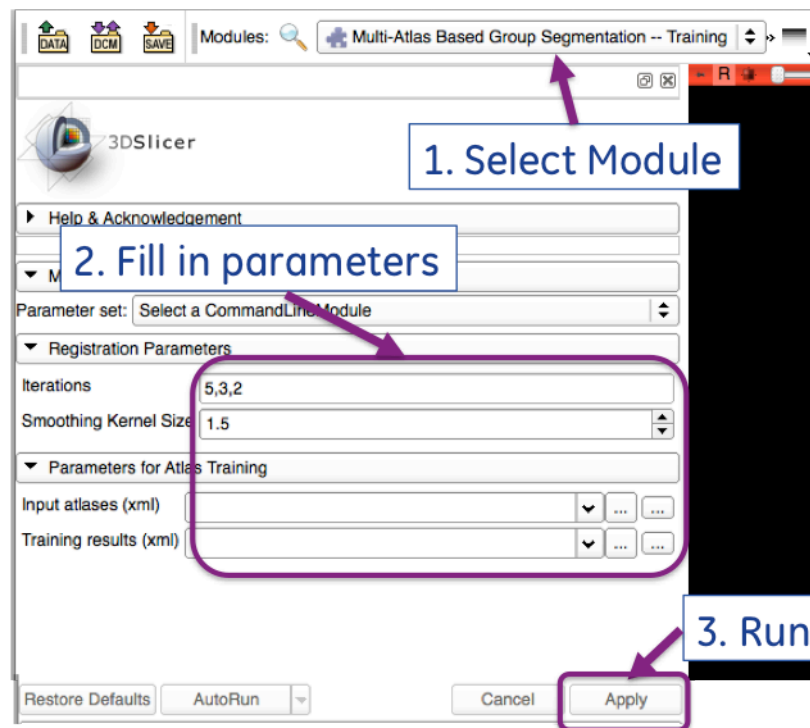


Figure 2 The training module

#### 4.1.2 Command line

```
IGR3D_MABMIS_Training [-i xx,xx,xx] [-s sigma] -- trainingXML TrainingData.xml -- atlasTreeXML TrainedAtlas.xml
```

- [-i xx,xx,xx]:(OPTIONAL) number of iterations used in image registration. It is three numbers separated by comma. Default value is 5,3,2.
- [-s sigma] : the size of the smoothing kernel that is used for smoothing deformation fields. Default value is 1.5.
- *TrainingData.xml* : an xml file that specifies information about the training images, including the number of training data, filenames of T1-weighted image and the segmented label images. An example can be found in the test folder.
- *TrainedAtlas.xml*: an xml file for output, that stores information about the trained atlas tree which can be used to segment

## 4.2. Testing

### 4.1.1 GUI

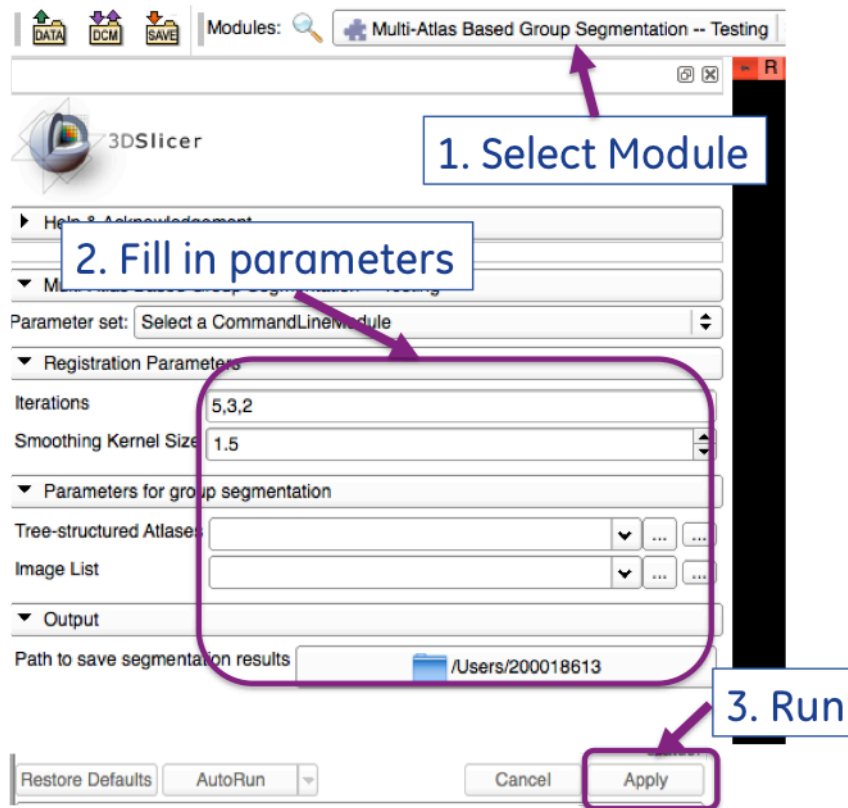


Figure 3 The testing module

### 4.1.2 Command line

```
IGR3D_MABMIS_Testing [-i xx,xx,xx] [-s sigma] --atlasTreeXML TrainedAtlas.xml --imageListXML TestImageList.xml --outputfolder OutputFolder
```

- [-i *xx,xx,xx*]:(OPTIONAL) number of iterations used in image registration. It is three numbers separated by comma. Default value is 5,3,2.
- [-s *sigma*] : the size of the smoothing kernel that is used for smoothing deformation fields. Default value is 1.5.
- *TrainedAtlas.xml*: an xml file that contains the trained atlas resulted from the training module.
- *TestImageList.xml*: an xml file that contains a list of test images to be segmented. An example can be found in the test folder
- *OutputFolder*: The location to save the segmentation results.

The segmentation results will be saved in the same location as the test images,, and are named as "xxxx\_seg.hdr", where "xxxx" is the test image file name.

## 5. Developer's guide

Here we provide the detailed descriptions on the ITK classes related to training, testing and preprocessing, respectively. Once the source code is modified, new executable files can be generated by following the instructions in 3.2.2.

### 5.1. Training

- main function in IGR3D\_MABMIS\_Training.cxx: This perform the training for multi-atlas multi-structure segmentation and builds an atlas tree using both training images and simulated images.
- itk::MABMISSimulateData: This class generate the simulated images from the atlas images based on PCA on the deformation fields between an atlas image that is selected as the atlas tree root and all other atlas images.
- itk::MABMISTreeOperation: This class contains basis operations on building an atlas tree.
- itk::MABMISDeformationFieldFilter: This class includes operations on the deformation fields, including reading, writing, and inverting deformation field, warping image using a given deformation field, composing an deformation field by concatenating two deformation fields, etc.

### 5.2. Testing

- IGR3D\_MABMIS\_Testing.cxx: This performs segmentations on a list of test images based on the pre-trained atlas tree. It first construct a new tree by connecting the test images to their most similar images in the atlas tree. Then, the deformations between each test image to all images in the original atlas tree are computed and are used to segment them using a weighted label fusion approach.

This file also includes functions used in the segmentation process, including

- GaussianWeightedLabelFusion: weighted label fusion
- PairwiseRegistrationOnTreeViaRoot: register any two images through the atlas tree root.

## **6. References**

[1] H. Jia, P.-T. Yap, D. Shen, "Iterative multi-atlas-based multi-image segmentation with tree-based algorithm", *NeuroImage* 59 (2012) 422-430.