

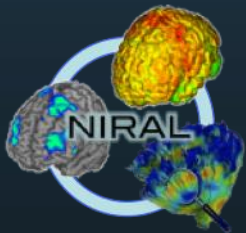


**Slicer**  
**SALT**  
Shape AnaLysis Toolbox

# GROUPS

## User Tutorial

Mahmoud Mostapha, Ilwoo Lyu, Martin Styner, Beatriz Paniagua  
August 2018



THE UNIVERSITY  
*of* NORTH CAROLINA  
*at* CHAPEL HILL



# GROUPS Tool Description

Step 1: Landmark-based Rigid Alignment

Step 2: Group-wise Shape Registration

# Description of GROUPS

Consistent shape correspondence is a prerequisite any group analysis investigating disease patterns and group variability

**Group-wise Registration For Shape Correspondence (GROUPS) tool** is a general framework for establishing correspondence of 3D models that employs group-wise registration in a spherical parametrization space

The inputs are spherical harmonics (SPHARM) point distribution models (PDM) in addition to user-defined geometrical features and landmarks. The output is SPHARM-PDM models with optimized correspondence obtained by minimizing the entropy of the joint distribution of features and landmarks at corresponding point locations

# Description of GROUPS

The GROUPS tool consists of the following detailed steps:

*Inputs: Surface Meshs + Landmarks + Common Sphere*

Step 1 (a) : **Rigid Alignment** uses RigidWrapper CLI

*Output: Rotated Parameterization Spheres*



*Inputs: Surface Mesh + Parameterization Sphere + Common Sphere*

Step 1 (b) : **Surface Remeshing** uses SurfRemesh CLI

*Output: Remeshed Surface*



*Inputs: Surface Meshs + Parameterization Spheres + Surface Features + landmarks*

Step 2 (a) : **Group-Wise Registration** uses Groups CLI

*Output: Aligned SPHARM Coefficients*



*Inputs: Surface Mesh + Parameterization Sphere + Common Sphere + SPHARM Coefficients*

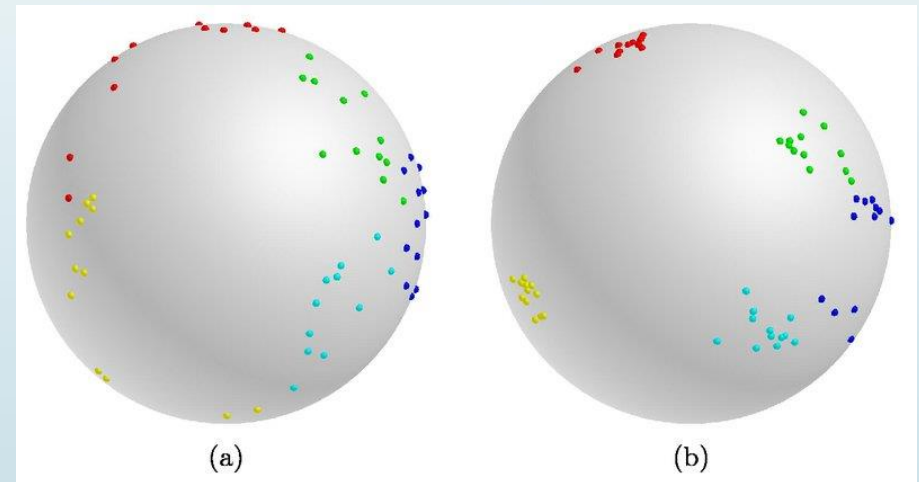
Step 2 (b) : **Surface Remeshing** uses SurfRemesh CLI

*Output: Remeshed Surface*

# Step 1 : Landmark-based Rigid Alignment

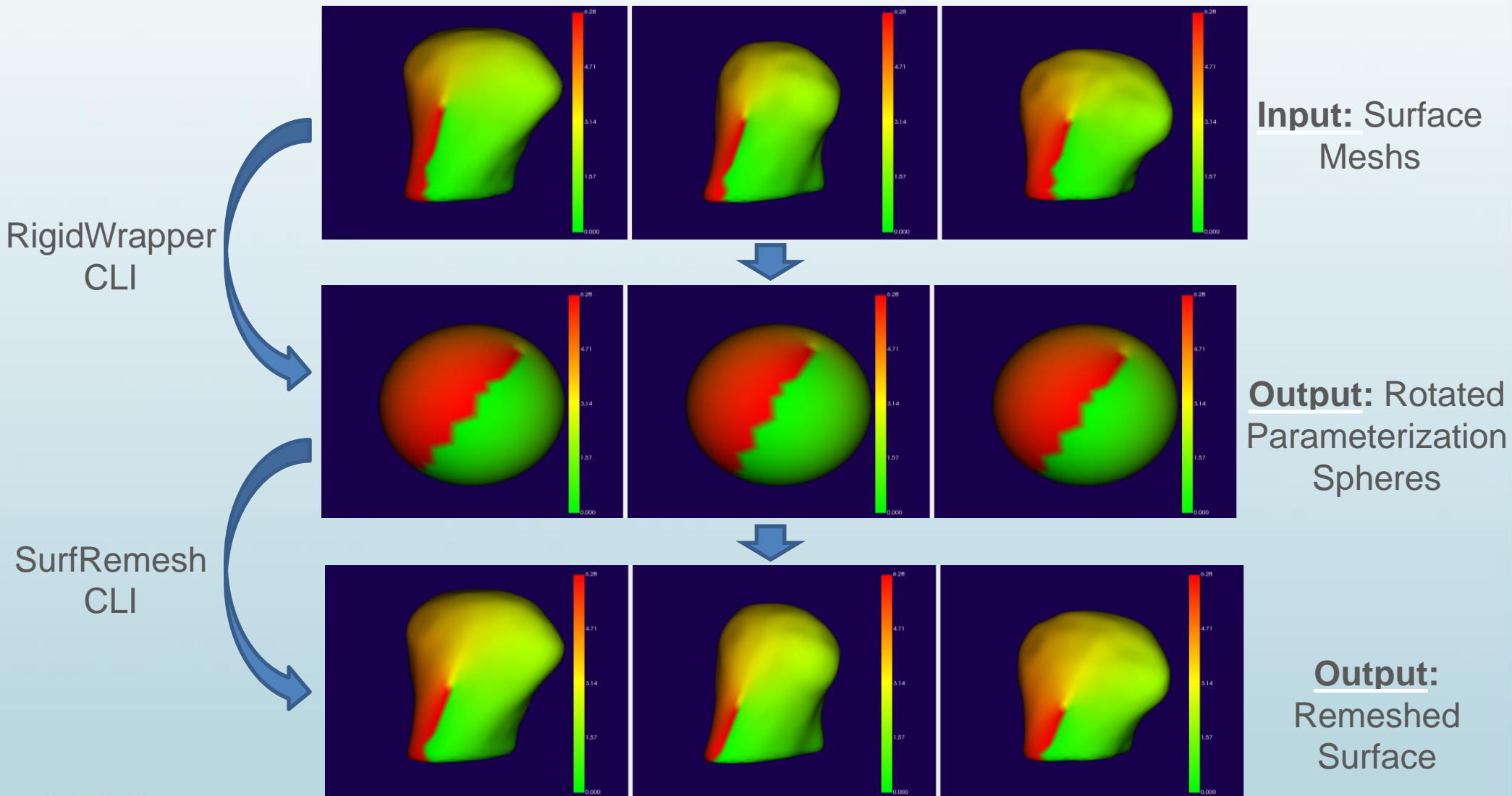
This step will:

- Improve the SPHARM-PDM initial correspondence using a set of user placed landmarks
- Landmarks are defined in terms of 3D Slicer Fiducials (\*.fcsv)
- Minimize the landmark distance errors on the sphere in terms of rigid alignment transformation
- Surfaces are Remeshed using rotated parametrization spheres



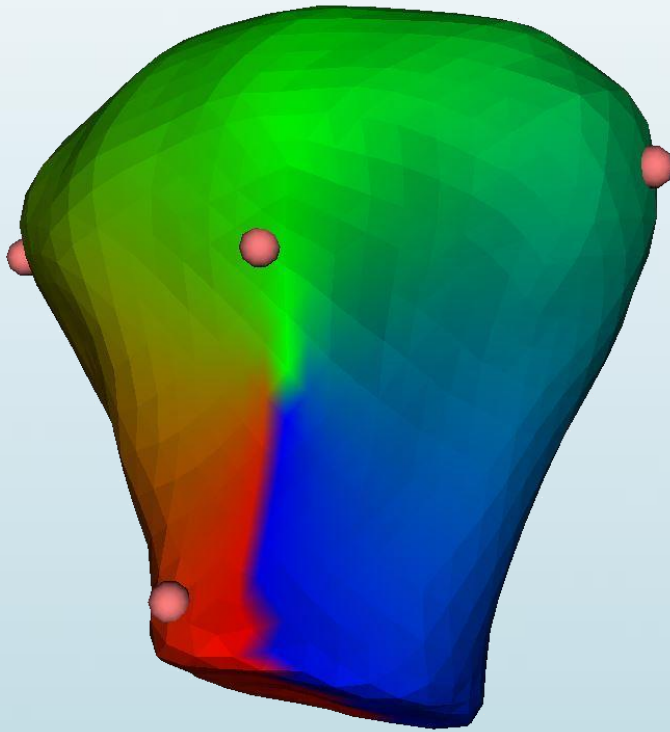
**Figure:** (a) initial landmarks of the SPHARM-PDM surfaces, and (b) aligned landmarks after rigid transformation on the sphere

# Step 1 : Landmark-based Rigid Alignment

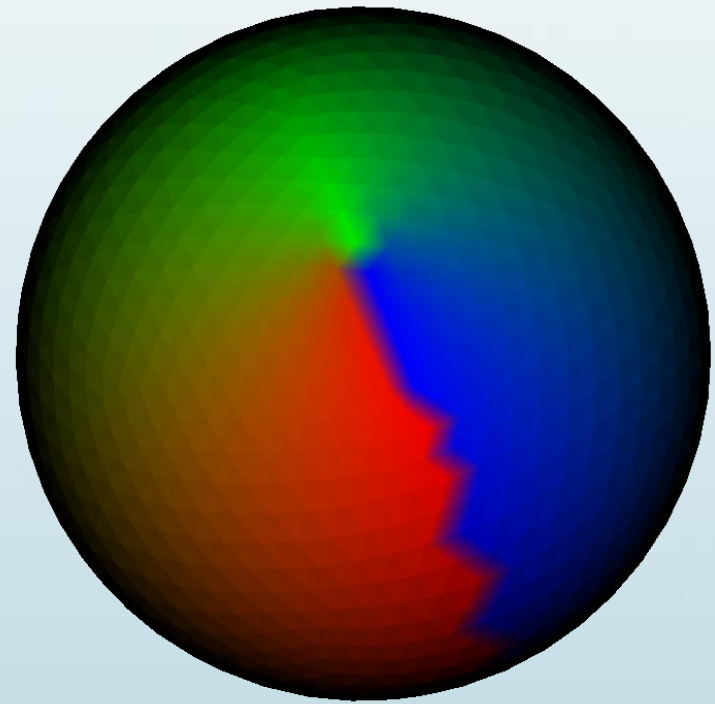


# Step 1 : Landmark-based Rigid Alignment

## Inputs



**Figure:** SPHARM-PDM Meshs  
(\*SPHARM.vtk) with user placed  
fiducals (\*.fcsv) for each subject

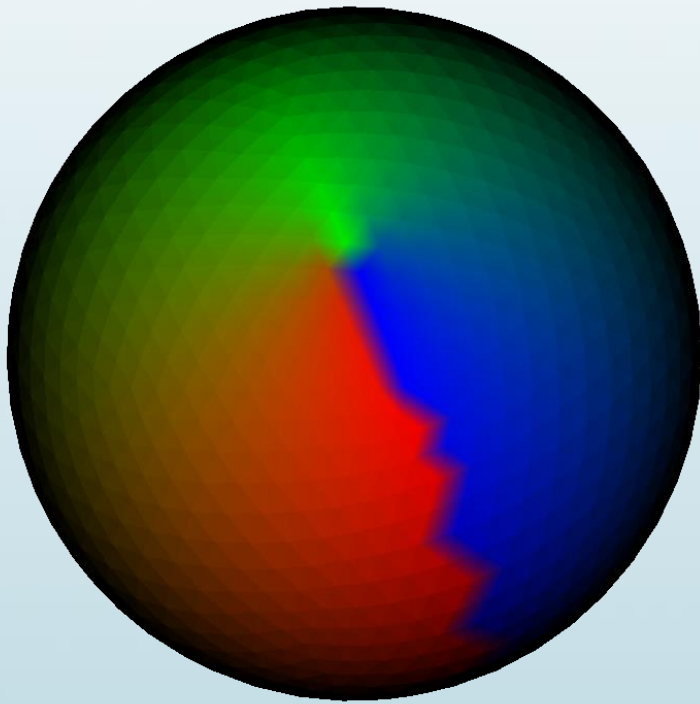


**Figure:** Common parametrization  
sphere obtained from the SPHARM-  
PDM pipeline (\*.surf\_para.vtk)

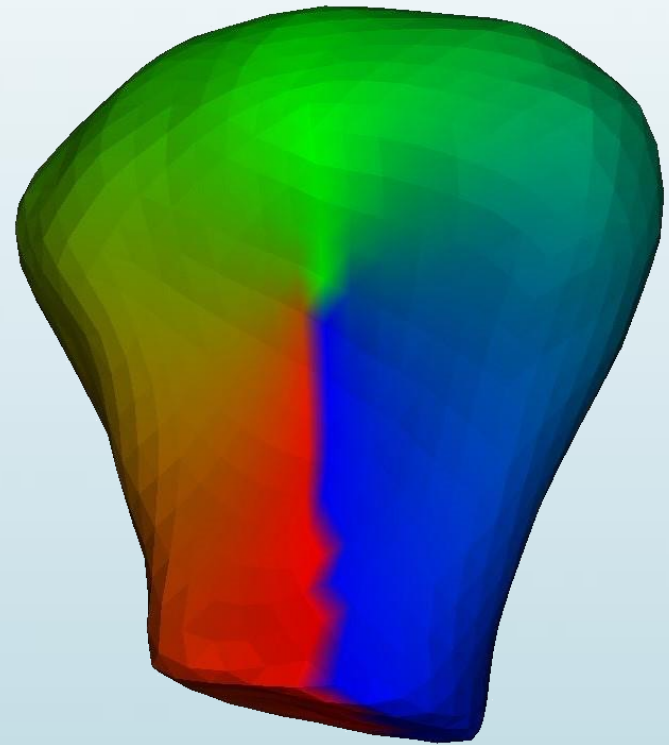


# Step 1 : Landmark-based Rigid Alignment

## Outputs



**Figure:** Rotated parametrization sphere using RigidWrapper CLI  
(\*rotSphere.vtk)



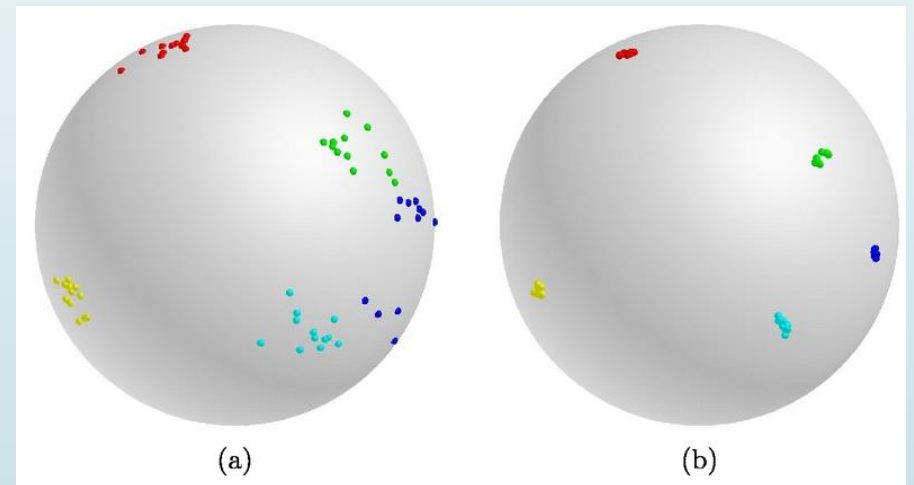
**Figure:** Remeshed Surface using SurfRemesh CLI  
(\* aligned.vtk)



# Step 2: Group-wise Shape Registration

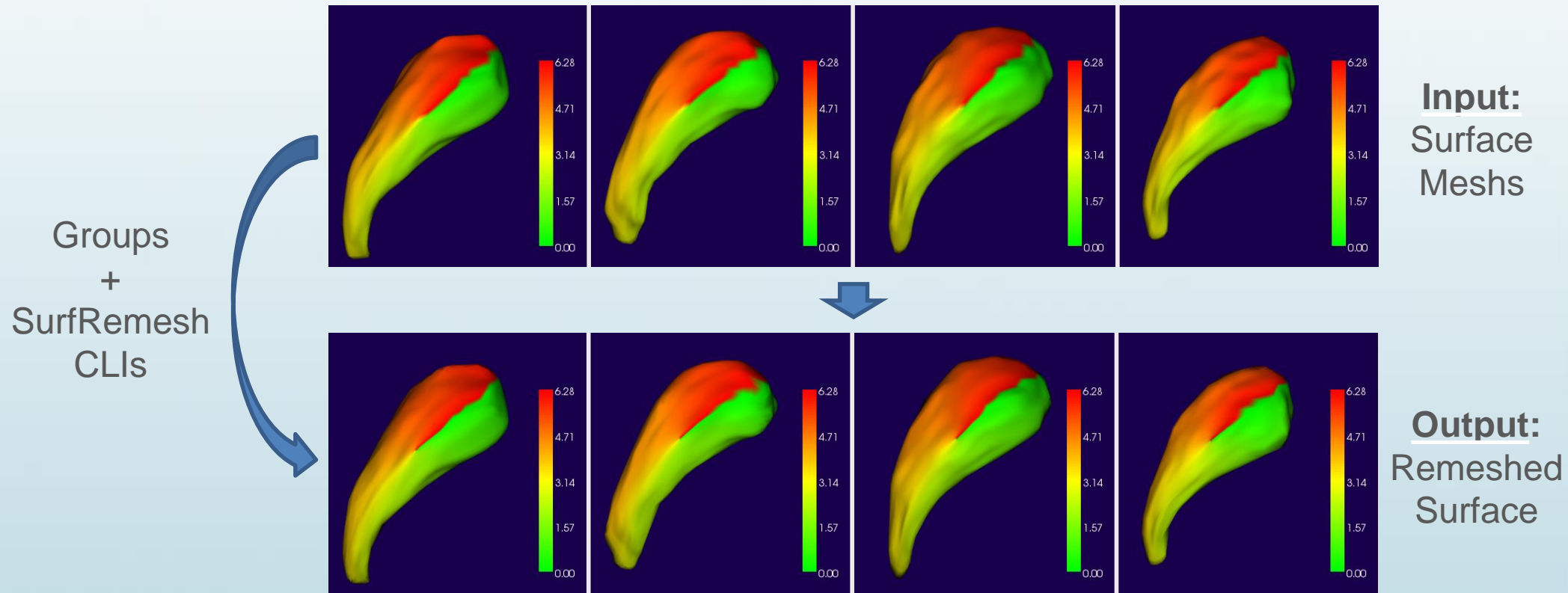
This step will:

- Further improve correspondence using group-wise registration in a spherical parametrization space
- Optimizing landmarks (local) and multidimensional features (global) by minimizing the joint entropy
- Features are pre-computed by the user and saved in SPHARM vtk files as point data arrays
- Surfaces are Remeshed using aligned SPHARM coefficients



**Figure:** (a) landmarks after rigid transformation on the sphere, and (b) final landmark alignment using group-wise shape registration

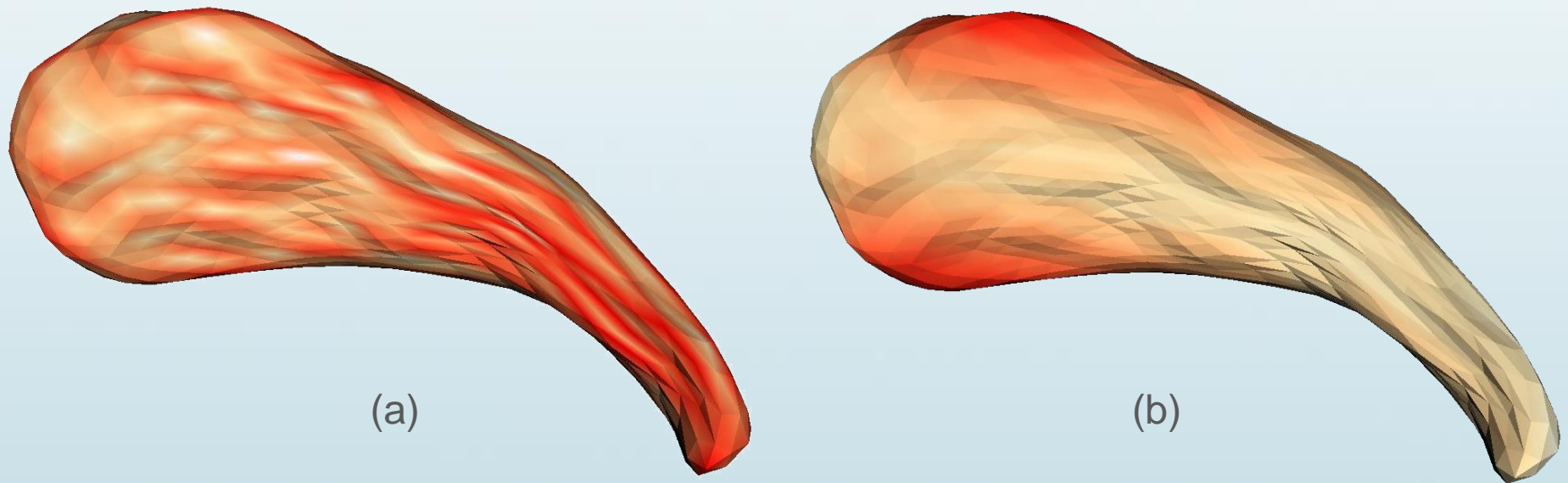
## Step 2: Group-wise Shape Registration



**Figure:** Examples of geometrical features generated by the user and stored in the (\*SPHARM.vtk) Surfaces as point data arrays. (a) Curvedness generated using the SpharmTool in the SPHARM-PDM pipeline

# Step 2 : Group-wise Shape Registration

## Inputs



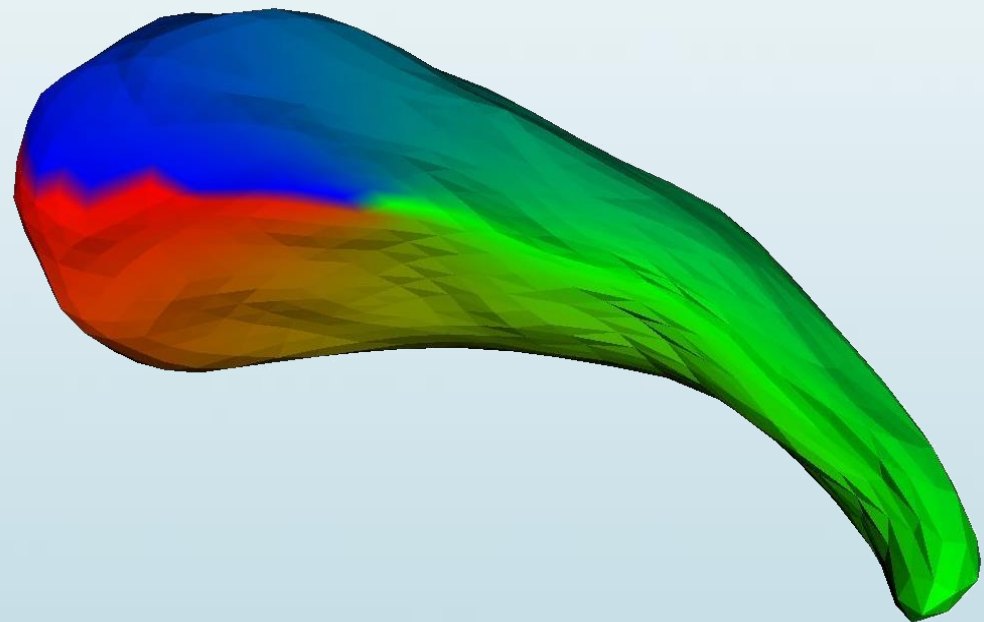
**Figure:** Examples of geometrical features generated by the user and stored in the (\*SPHARM.vtk) Surfaces as point data arrays. (a) Curvedness generated using the SpharmTool in the SPHARM-PDM pipeline. (b) Partial radius (thickness) generated when medial mesh is generated in the SPHARM-PDM pipeline. If landmarks to be used, the user needs to save landmarks as a binary array of the Vertex IDs saved as an array called “Landmarks”

# Step 2 : Group-wise Shape Registration

## Outputs

```
stx_noscale_101247_...l_editLV_CaudL.coeff X
1 0.000000 0.000000 1.000000
2 15
3 0.010421 -0.048431
4 0.011028 -0.024598
5 0.000746 -0.455415
6 0.063409 -0.015192
7 0.019351 0.013289
8 -0.020645 -0.069241
9 0.030415 0.004134
10 0.028381 -0.023158
11 0.031246 -0.044047
12 -0.025440 0.025056
13 -0.002316 -0.011969
14 -0.041846 -0.035913
15 0.004656 -0.001571
16 -0.003190 0.007044
17 -0.061342 0.024651
18 -0.022666 0.016789
19 0.154128 -0.000711
```

(a)



(b)

**Figure:** (a) SPHARM deformation coefficients (\*coeff) produced by the Groups CLI. (b) The input surface is then remeshed (\*SPHARM.vtk) using SurfRemesh CLI

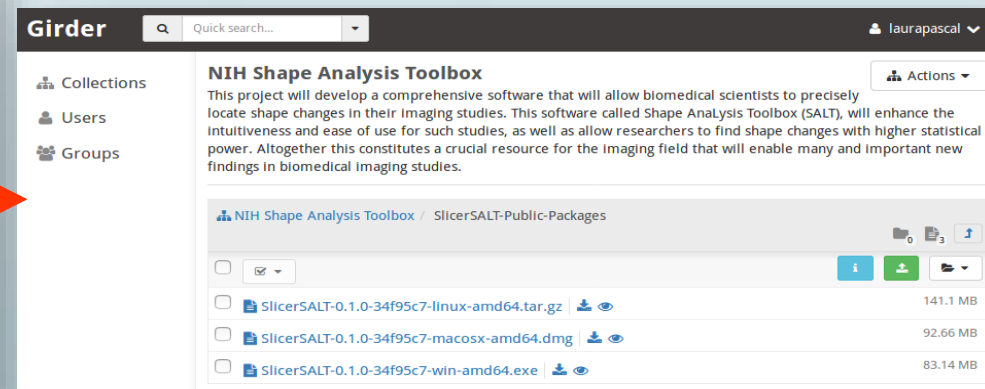
# Installation of GROUPS Tool

GROUPS tool can be used with two open-source software platforms:

- **SlicerSALT:** which is the dissemination vehicle of powerful shape analysis methodology. This software is a light-weight, customized version of 3D Slicer. It contains GROUPS as modules
- **3D Slicer:** which is an open-source and free software platform for medical image informatics, image processing, and three-dimensional visualization. GROUPS can be downloaded as an extension

# GROUPS Installation on SlicerSALT

Download the SlicerSALT packages for your respective operating system from the [SlicerSALT website](#) and install it. GROUPS will be ready to use when



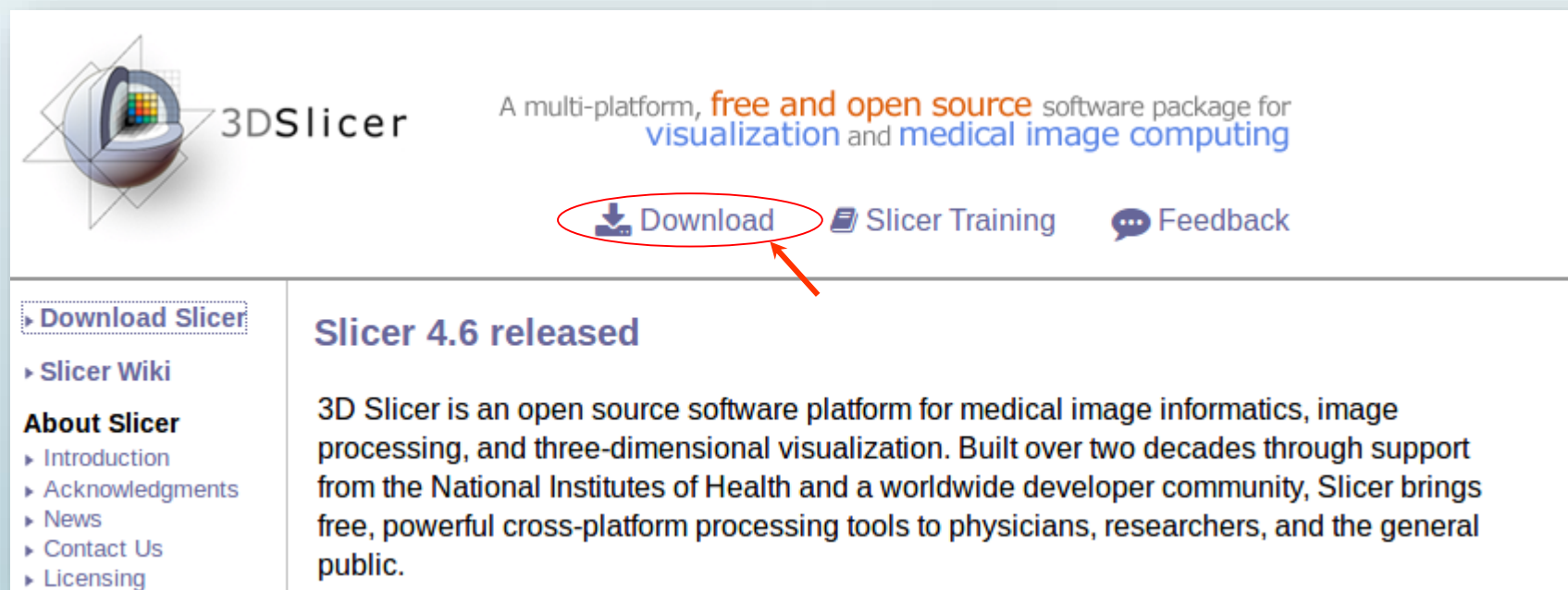
Powered by Girder






# GROUPS Installation on 3D Slicer

1. Download 3D Slicer packages for your respective operating system on the [3D Slicer website](#) and install it



 **3DSlicer**

A multi-platform, **free and open source** software package for **visualization** and **medical image computing**

[Download](#) [Slicer Training](#) [Feedback](#)

► [Download Slicer](#)

► [Slicer Wiki](#)

**About Slicer**

- [Introduction](#)
- [Acknowledgments](#)
- [News](#)
- [Contact Us](#)
- [Licensing](#)

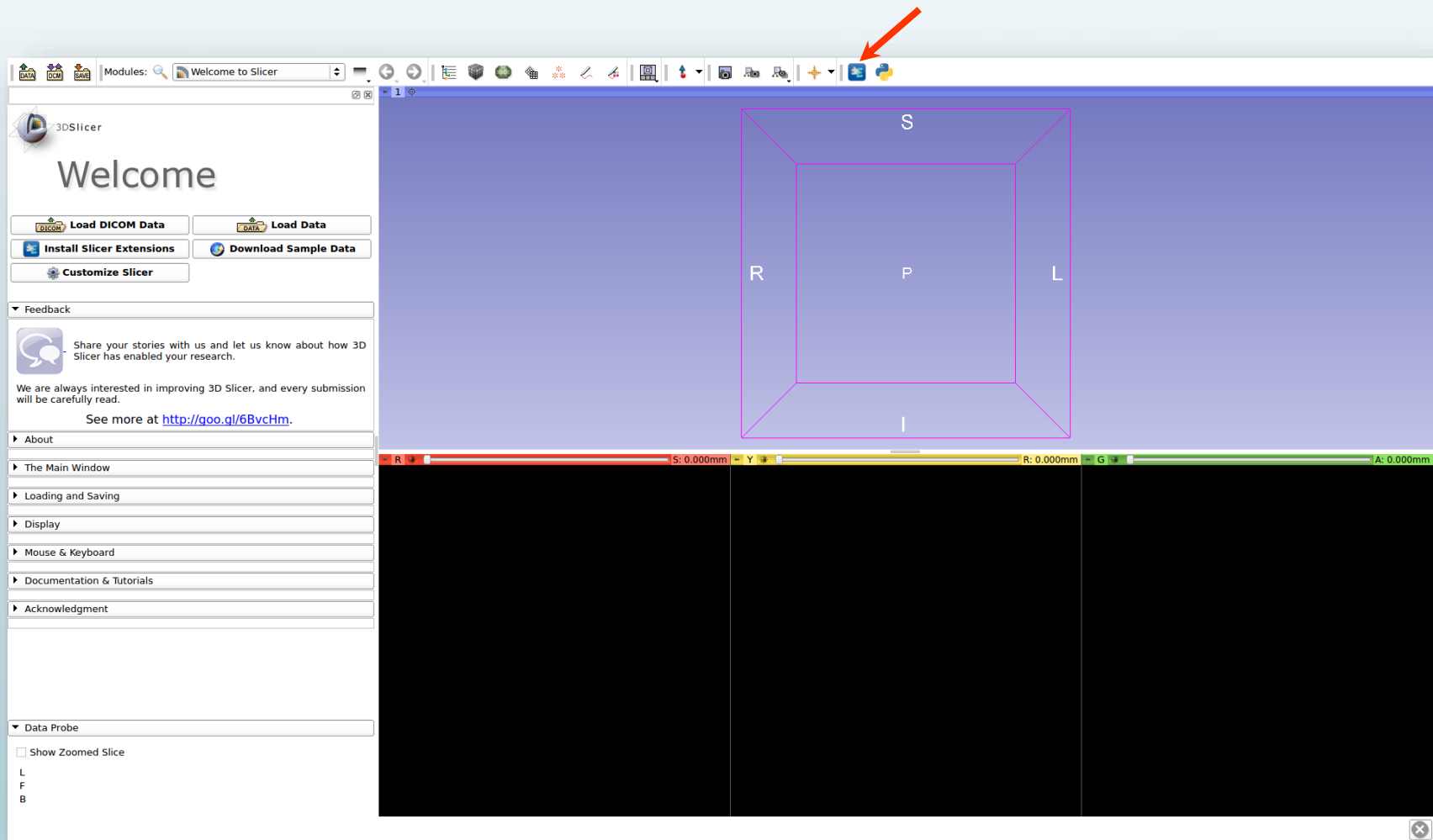
## Slicer 4.6 released

3D Slicer is an open source software platform for medical image informatics, image processing, and three-dimensional visualization. Built over two decades through support from the National Institutes of Health and a worldwide developer community, Slicer brings free, powerful cross-platform processing tools to physicians, researchers, and the general public.



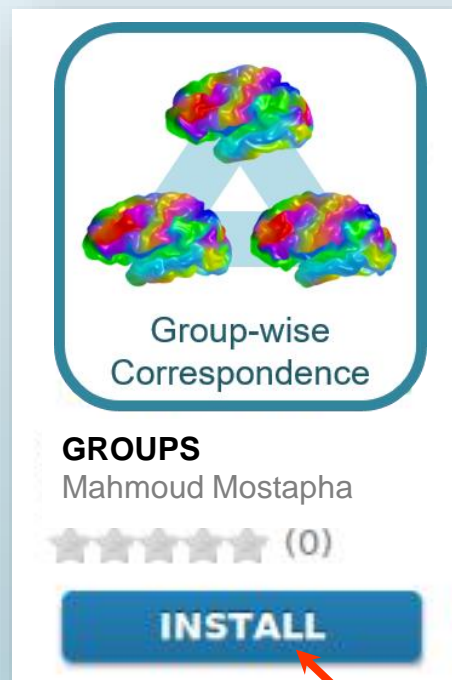
# GROUPS Installation on 3D Slicer

2. In 3D Slicer, open the Extension Manager



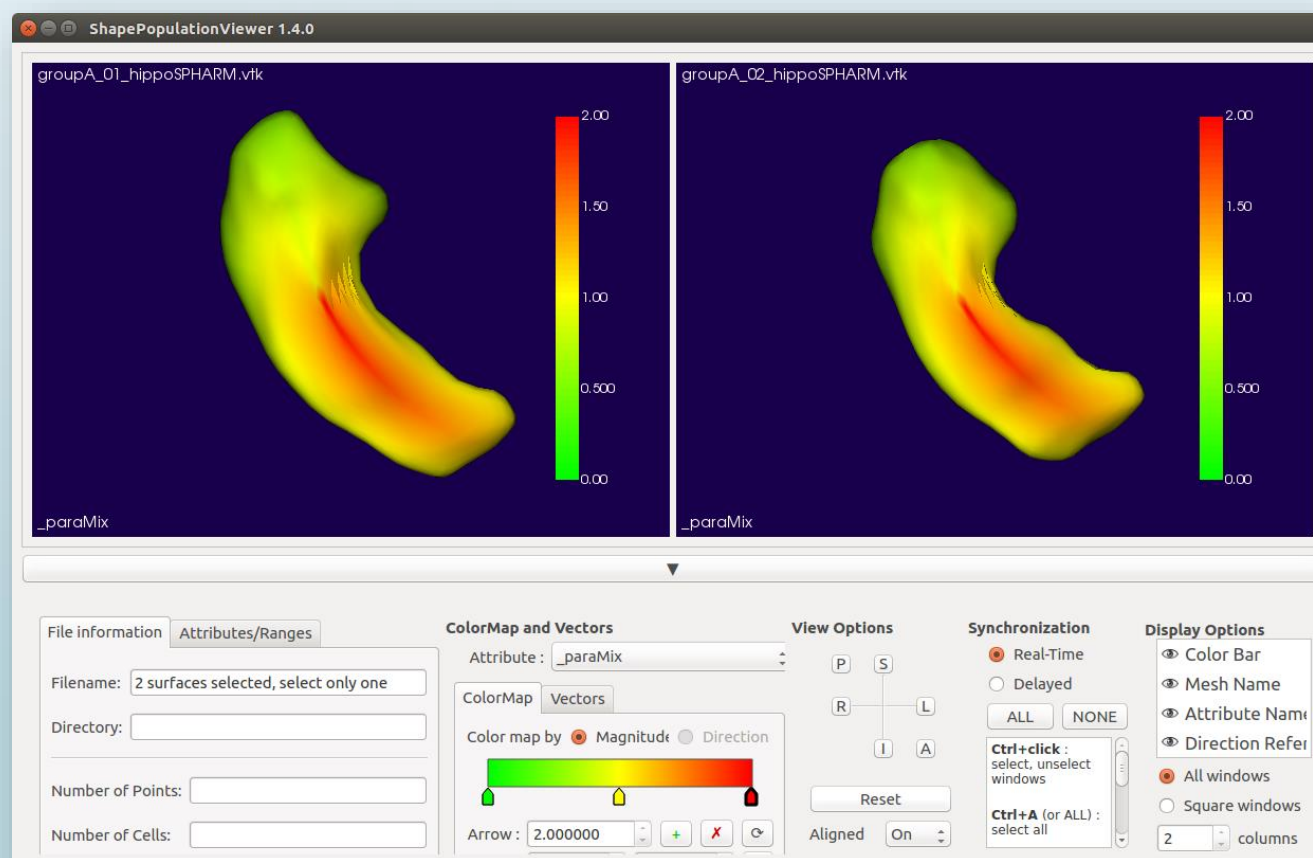
# GROUPS Installation on 3D Slicer

3. In the *Install Extension* tab, select **Shape Analysis** under *Categories*
4. Under **SPHARM-PDM**, select the *Install* button and restart Slicer when prompted



# GROUPS Installation on 3D Slicer

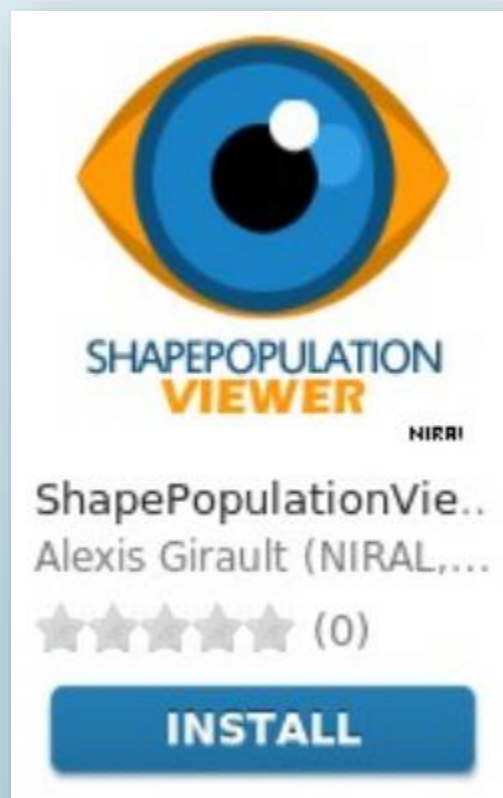
For quality control, we analyze our GROUPS outputs with **Shape Population Viewer** extension. Shape Population Viewer can be installed as a 3D Slicer extension or as an external binary. This module is included as part of the SlicerSALT package



# GROUPS Installation on 3D Slicer

To install **Shape Population Viewer** as a 3D Slicer extension:

- i. Open *Extension Manager*, in the *Install Extensions* tab, select '*Shape Analysis*' under *Categories*
- ii. Select the appropriate *Install* button and restart 3D Slicer when prompted



# GROUPS Installation on 3D Slicer

To install **Shape Population Viewer** as an external binary:

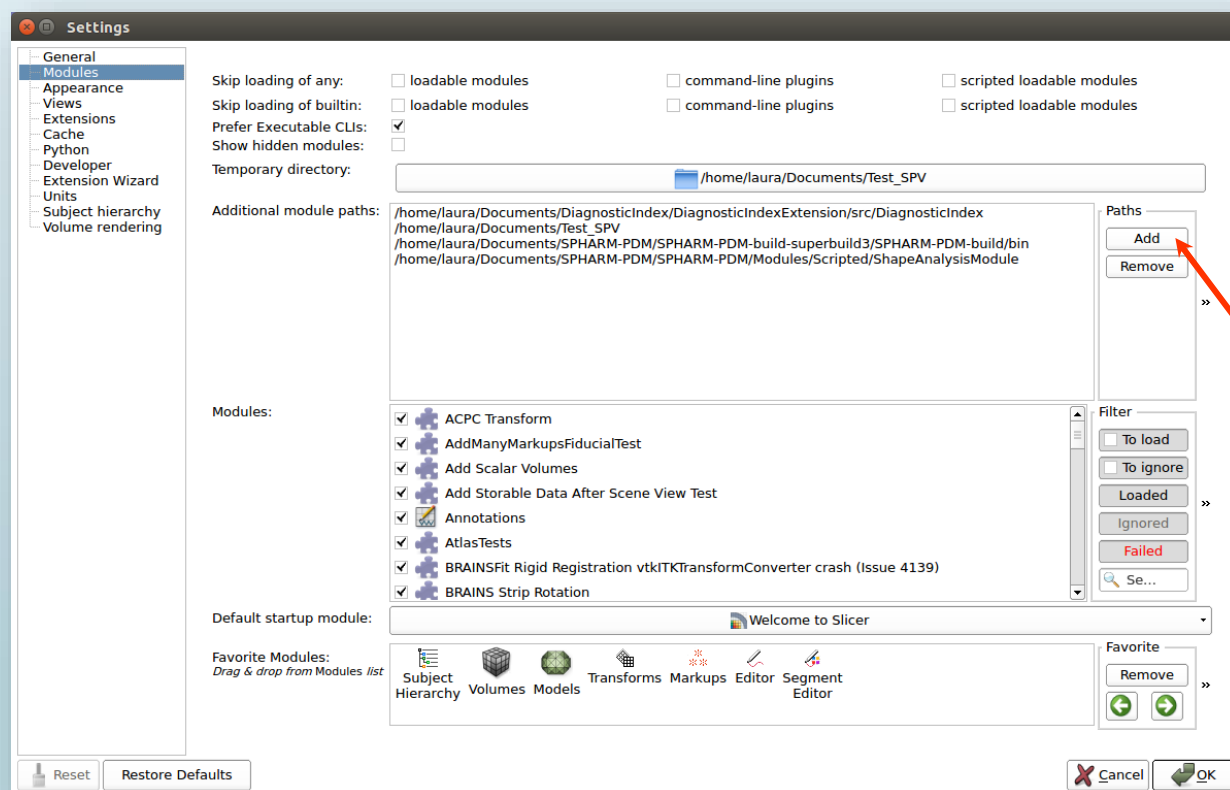
i. Download ShapePopulationViewer package for your respective operating system on [NITRC website](#)

ii. In 3D Slicer, open *Application Settings* in the *Edit* Menu.

On the tab *Modules*, **Add** the folder where ShapePopulationViewer is stored

iii.

iii. Restart 3D Slicer



# Rigid Alignment Use

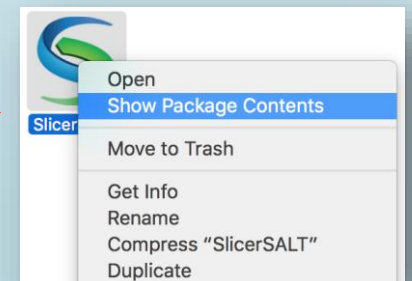
Rigid Alignment tool can be used by two different ways:

- As **command-line tool** through the terminal thanks to SlicerSALT
- As a **module** of SlicerSALT or 3DSlicer

# Rigid Alignment Command-Line Tool

Rigid Alignment method can be run on several cases through a terminal thanks to two files included in the SlicerSALT package:

- RigidAlignment-parameters.ini which allows the user to specify the inputs, outputs and the parameters of the RigidAlignment tool
  - RigidAlignment.py python script which will apply RigidAlignment method on the given input cases with the parameters specified in the RigidAlignment-parameters.ini file
- *RigidAlignment.py and RigidAlignment-parameters.ini files location:*
- *On Linux and Windows: share/Slicer-4.7/CommandLineTool*
  - *On MacOs: Open the SlicerSALT Contents*  
→ *Contents/share/Slicer-4.7/CommandLineTool* →





# Rigid Alignment Command-Line Tool

**Step 1:** Modification of the *RigidAlignment-parameters.ini* file by specifying the **directories** needed for tool CLIs

**Step 2:** Launch Rigid Alignment method with the following command-lines:

➤ On Linux and Windows:

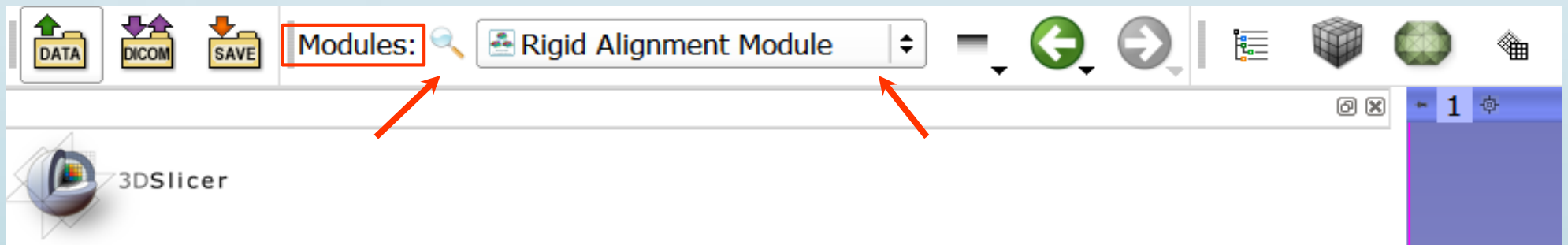
```
$cd path-to-the-SlicerSALT-package  
$./SlicerSALT      --no-main-window      --python-script      share/Slicer-  
4.7/CommandLineTool/RigidAlignment.py      share/Slicer-  
4.7/CommandLineTool/RigidAlignment-parameters.ini
```

➤ On MacOs:

```
$cd path-to-the-SlicerSALT-package/SlicerSALT.app/Contents/MacOS  
$./SlicerSALT      --no-main-window      --python-script      ../share/Slicer-  
4.7/CommandLineTool/RigidAlignment.py      ../share/Slicer-  
4.7/CommandLineTool/RigidAlignment-parameters.ini
```

# Rigid Alignment Module

In 3D Slicer or in SlicerSALT, select ***Rigid Alignment Module*** from the *Modules* drop-down menu (*Category: Shape Analysis*) or on the Search bar



# Setting up Input Directories

## RigidAlignment tab

- For ***Input Models Directory***, select the folder which contains the input surface meshes (\*.vtk)
- For ***Input Fiducial Files Directory***, select the folder where the landmarks (fiducials) files are stored (\*.fcsv)
- For ***Input Common Unit Sphere***, select the folder where the common parametrization sphere is stored (\*.vtk)

▼ RigidAlignment Directories	
Input Models Directory	C:/Program Files/Slicer 4.8.1
Input Fiducial Files Directory	C:/Program Files/Slicer 4.8.1
Input Common Unit Sphere	C:/Program Files/Slicer 4.8.1
Output Spherical Models Directory	C:/Program Files/Slicer 4.8.1
Output Models Directory	C:/Program Files/Slicer 4.8.1

# Setting up Output Directories

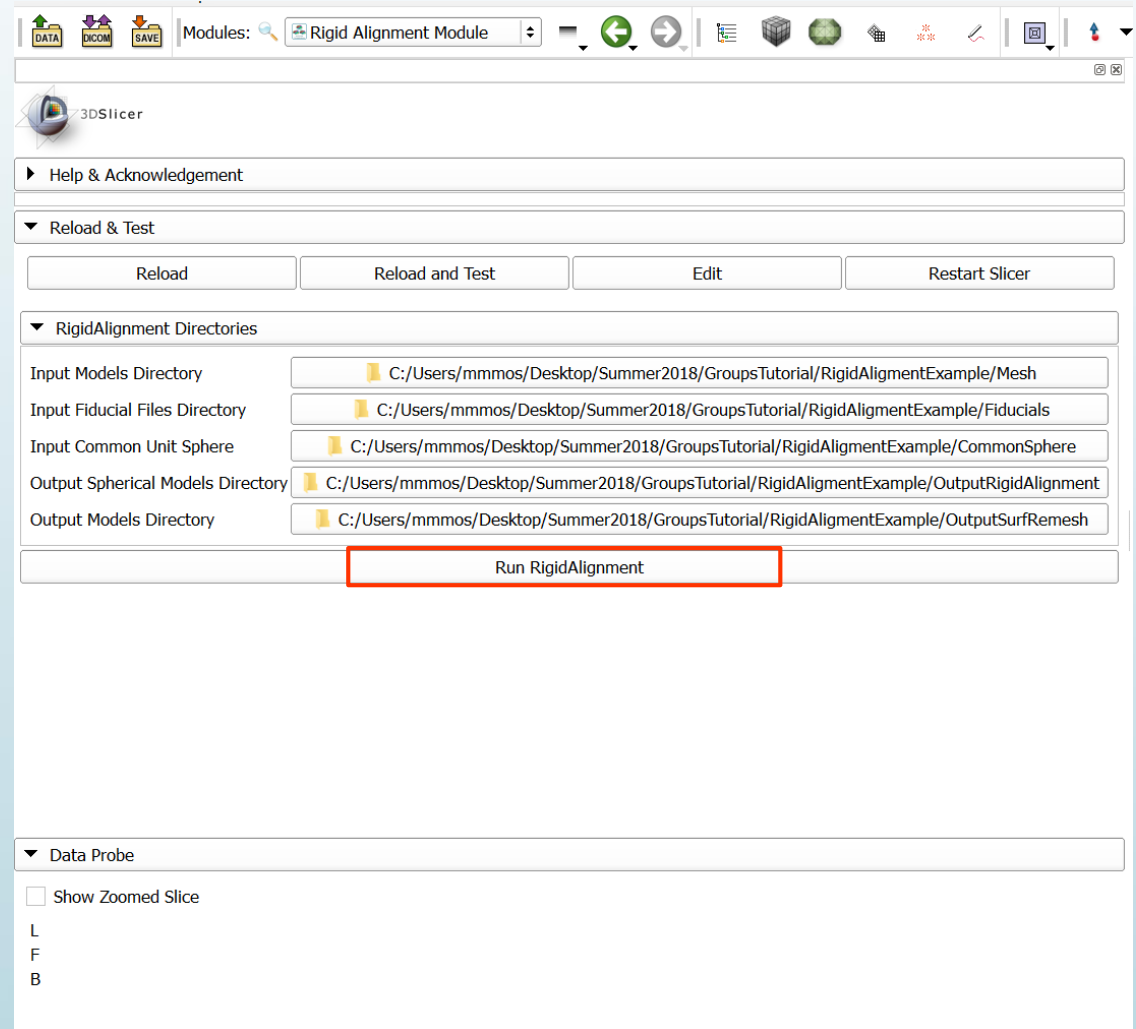
## RigidAlignment tab

- For ***Output Spherical Models Directory***, select the folder where the output of the RigidWrapper CLI will be stored (\*.vtk)
- For ***Output Models Directory***, select the folder where the output of the SurfRemesh CLI will be stored (\*.vtk)

▼ RigidAlignment Directories	
Input Models Directory	C:/Program Files/Slicer 4.8.1
Input Fiducial Files Directory	C:/Program Files/Slicer 4.8.1
Input Common Unit Sphere	C:/Program Files/Slicer 4.8.1
Output Spherical Models Directory	C:/Program Files/Slicer 4.8.1
Output Models Directory	C:/Program Files/Slicer 4.8.1

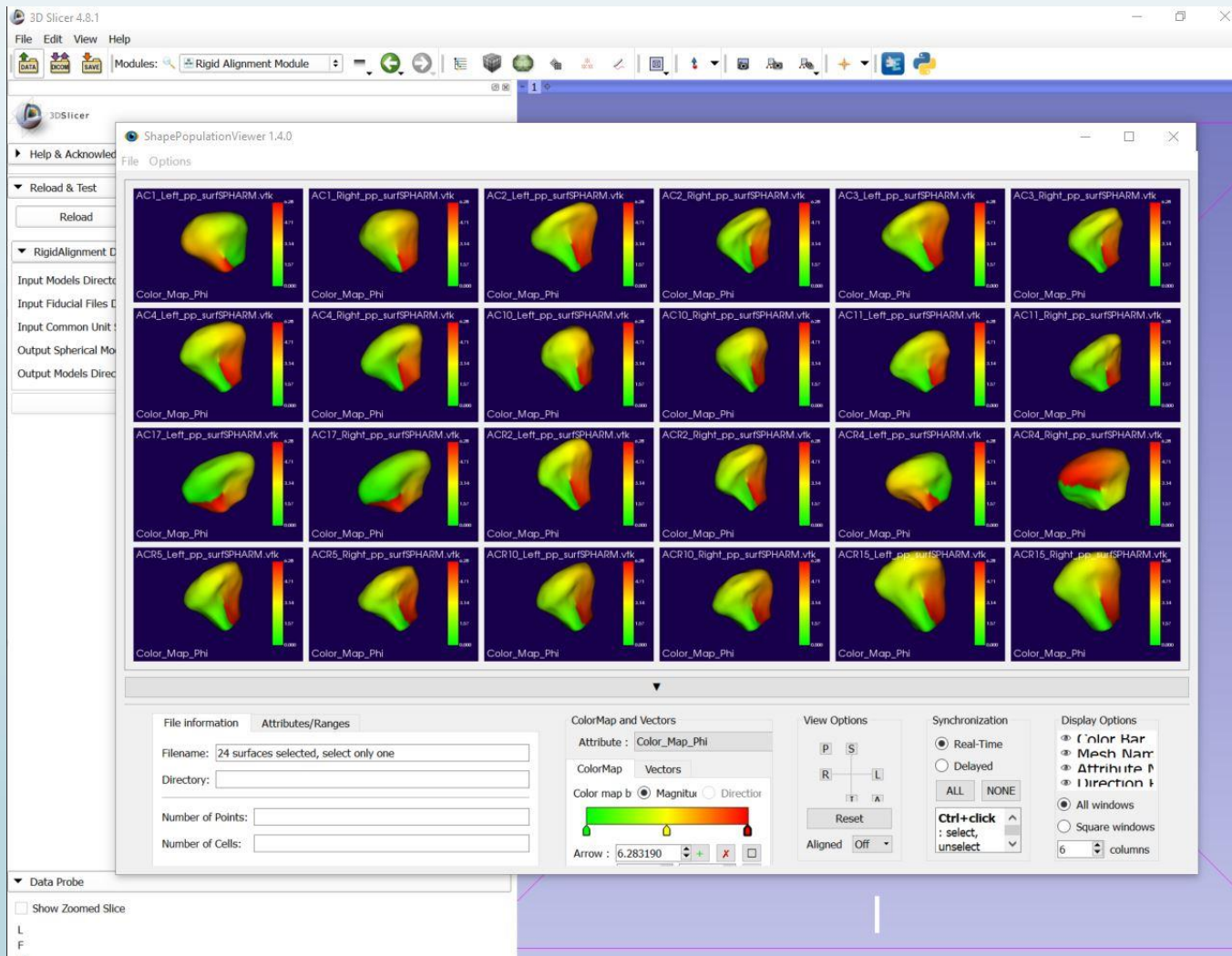
# Running Rigid Alignment Module

Click on the  
***Run RigidAlignment***  
button, to run the CLIs on  
the provided inputs



# Running Rigid Alignment Module

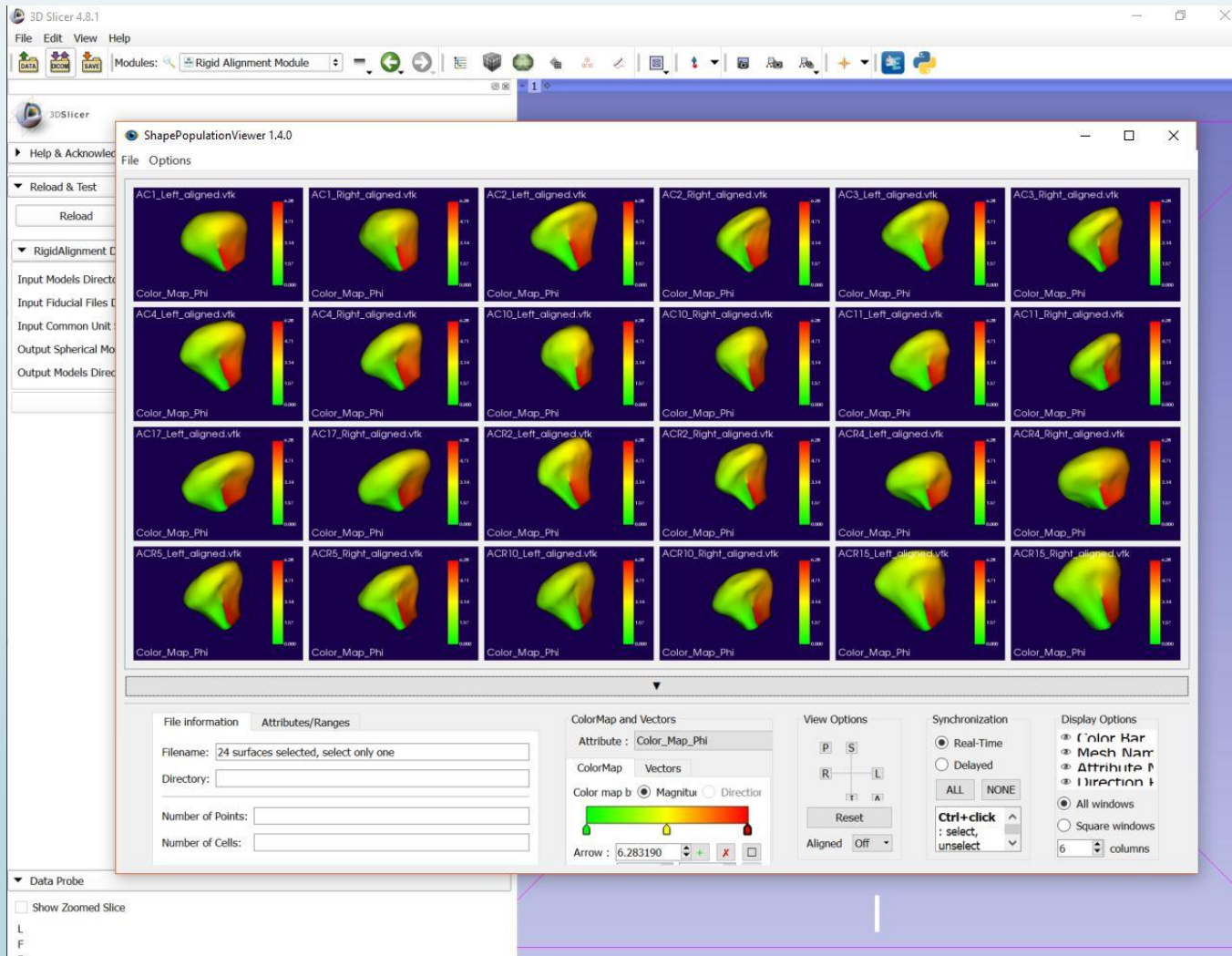
Shape Population Viewer will pop up to preview the input meshes giving the user the chance to inspect the input data before running the tool





# Running Rigid Alignment Module

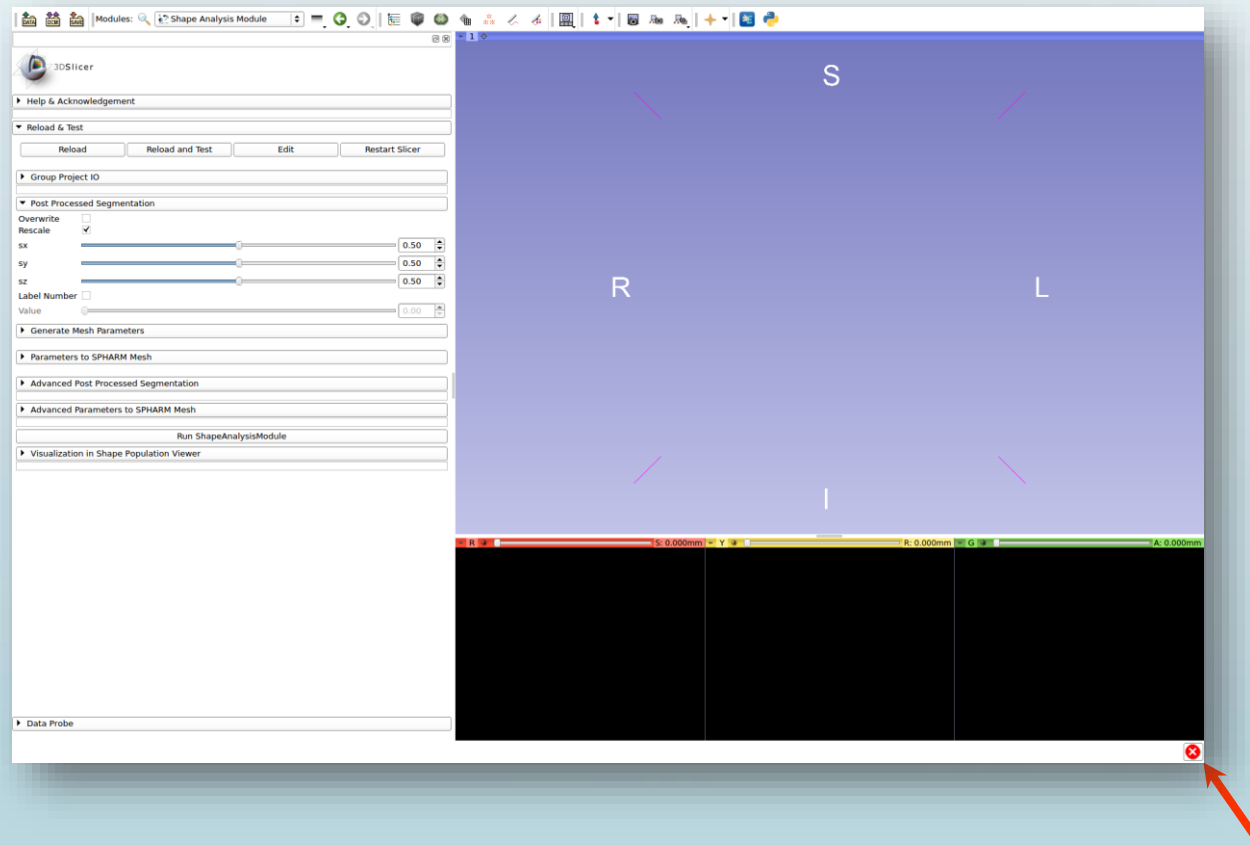
Shape Population Viewer will pop up also after the module finished processing giving the user the chance to check the remeshed surfaces





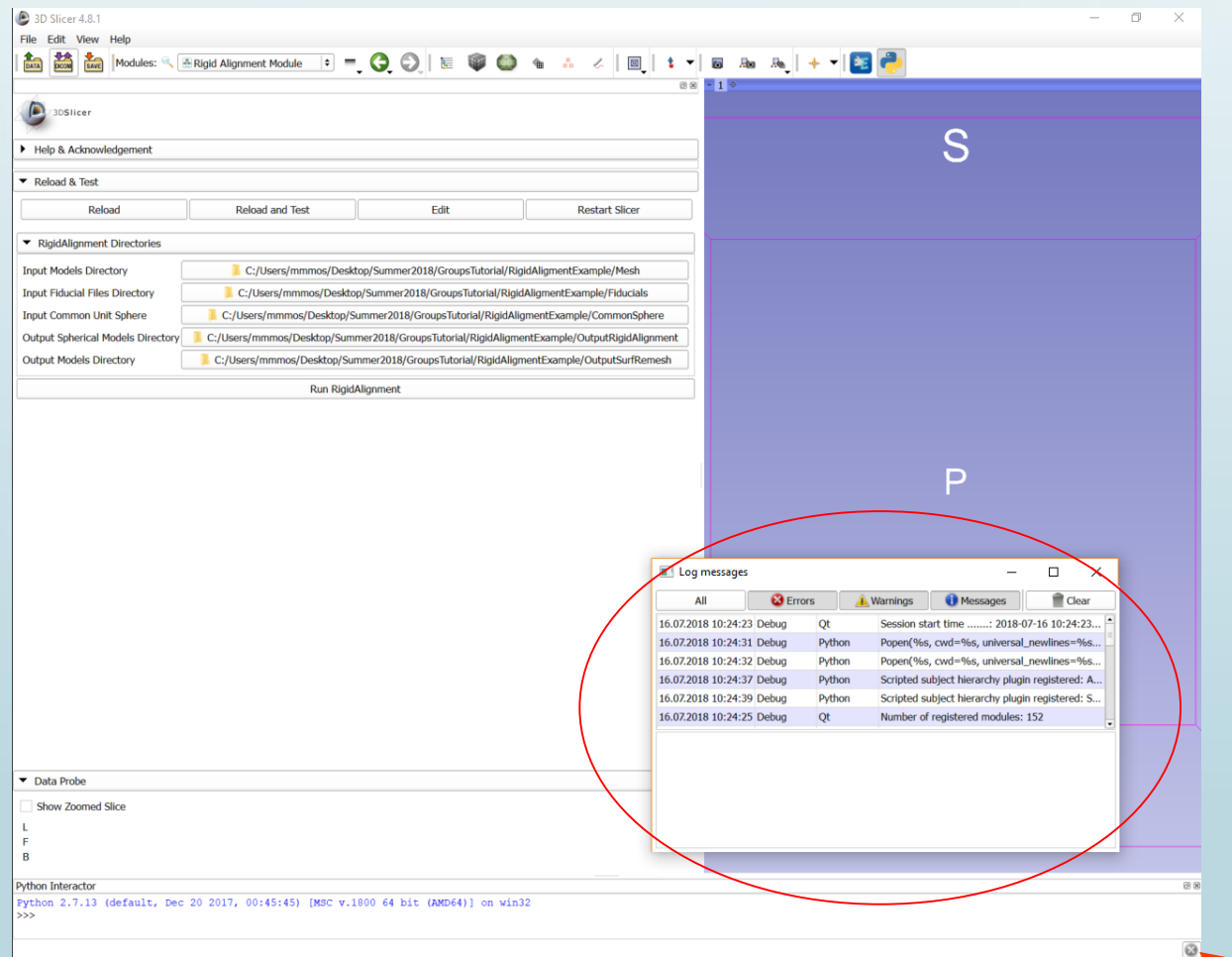
# Debugging

*3D Slicer's Error Log* can also be used for debugging if the module was completed with errors. To open it, click on the red icon at the bottom right.



# Debugging

*3D Slicer's Error Log* can also be used for debugging if the module was completed with errors. To open it, click on the red icon at the bottom right.

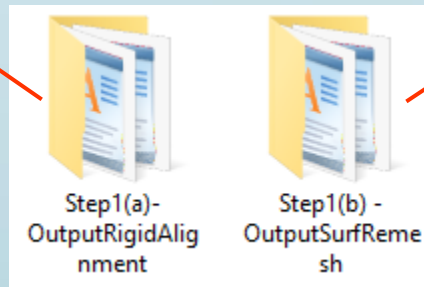


# Output Files

The outputs files for the two sub-steps of Rigid Alignment Module are stored in the two folders specified by the user:

AC4\_Left\_rotSphere.vtk  
AC4\_Right\_rotSphere.vtk  
AC10\_Left\_rotSphere.vtk  
AC10\_Right\_rotSphere.vtk  
AC11\_Left\_rotSphere.vtk  
AC11\_Right\_rotSphere.vtk  
AC17\_Left\_rotSphere.vtk  
AC17\_Right\_rotSphere.vtk

This block contains a list of eight VTK files. Each file name is preceded by a small icon of a document with a blue 'A' in the top left corner. The files are arranged in a vertical list. A red arrow points from the bottom of this list to the 'Step1(a)' folder icon in the central diagram.



AC4\_Left\_aligned.vtk  
AC4\_Right\_aligned.vtk  
AC10\_Left\_aligned.vtk  
AC10\_Right\_aligned.vtk  
AC11\_Left\_aligned.vtk  
AC11\_Right\_aligned.vtk  
AC17\_Left\_aligned.vtk  
AC17\_Right\_aligned.vtk

This block contains a list of eight VTK files. Each file name is preceded by a small icon of a document with a blue 'A' in the top left corner. The files are arranged in a vertical list. A red arrow points from the bottom of this list to the 'Step1(b)' folder icon in the central diagram.

# Group-wise Registration Use

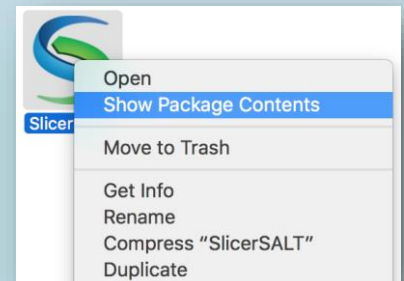
Group-wise Registration tool can be used by two different ways:

- As **command-line tool** through the terminal thanks to SlicerSALT
- As a **module** of SlicerSALT or 3DSlicer

# Group-wise Registration Command-Line Tool

Group-wise Registration method can be run on several cases through a terminal thanks to two files included in the SlicerSALT package:

- GroupWiseRegistration-parameters.ini which allows the user to specify the inputs, outputs and the parameters of the Group-wise Registration tool
  - GroupWiseRegistration.py python script which will apply GroupWiseRegistration method on the given input cases with the parameters specified in the GroupWiseRegistration -parameters.ini file
- *GroupWiseRegistration.py and GroupWiseRegistration -parameters.ini files location:*
- *On Linux and Windows: share/Slicer-4.7/CommandLineTool*
  - *On MacOS: Open the SlicerSALT Contents*  
→ *Contents/share/Slicer-4.7/CommandLineTool* →



# Group-wise Registration Command-Line Tool

**Step 1:** Modification of the *GroupWiseRegistration-parameters.ini* file by specifying the **directories and parameters** needed for tool CLIs

**Step 2:** Launch Group Wise Registration method with the following command-lines:

➤ On Linux and Windows:

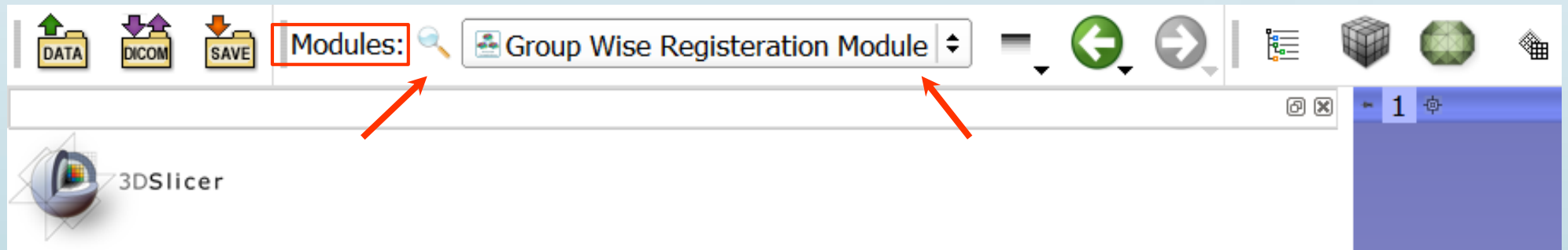
```
$cd path-to-the-SlicerSALT-package  
$./SlicerSALT      --no-main-window      --python-script      share/Slicer-  
4.7/CommandLineTool/GroupWiseRegistration.py      share/Slicer-  
4.7/CommandLineTool/GroupWiseRegistration-parameters.ini
```

➤ On MacOS:

```
$cd path-to-the-SlicerSALT-package/SlicerSALT.app/Contents/MacOS  
$./SlicerSALT      --no-main-window      --python-script      ../share/Slicer-  
4.7/CommandLineTool/GroupWiseRegistration.py      ../share/Slicer-  
4.7/CommandLineTool/GroupWiseRegistration-parameters.ini
```

# Group-wise Registration Module

In 3D Slicer or in SlicerSALT, select **Group-wise Registration Module** from the *Modules* drop-down menu (Category: Shape Analysis) or on the Search bar









# Setting up Input Directories

## Groups tab

- For ***Input Models Directory***, select the folder which contains the input surface meshes (\*.SPHARM.vtk)
- For ***Input Spherical Models Directory***, select the folder where the spherical parametrization files are stored (\*.para.vtk)

▼ GROUPS Directories	
Input Models Directory	 C:/Program Files/Slicer 4.8.1
Input Spherical Models Directory	 C:/Program Files/Slicer 4.8.1
Output Coefficients Directory	 C:/Program Files/Slicer 4.8.1
Output Models Directory	 C:/Program Files/Slicer 4.8.1

# Setting up Output Directories

## Groups tab

- For ***Output Coefficients Directory***, select the folder where the output of the Groups CLI will be stored (\*.Coeff)
- For ***Output Models Directory***, select the folder where the output of the SurfRemesh CLI will be stored (\*.vtk)

▼ GROUPS Directories	
Input Models Directory	C:/Program Files/Slicer 4.8.1
Input Spherical Models Directory	C:/Program Files/Slicer 4.8.1
Output Coefficients Directory	C:/Program Files/Slicer 4.8.1
Output Models Directory	C:/Program Files/Slicer 4.8.1

# Features and Parameters

## Groups Parameters tab

- Once the user specifies the input models directory, the geometrical features/properties stored in the vtk files are dynamically populated into a list where the user can select what features to include (Weight > 0)

The screenshot shows the 'Groups Parameters' tab in the Slicer SALT software. It is divided into two main sections: 'GROUPS Directories' and 'GROUPS Parameters'.

**GROUPS Directories:**

- Input Models Directory:** C:/Users/mmmos/Desktop/Summer2018/GroupsTutorial/Groups/AmyL\_Small\_lowd/Mesh
- Input Spherical Models Directory:** C:/Program Files/Slicer 4.8.1
- Output Coefficients Directory:** C:/Program Files/Slicer 4.8.1
- Output Models Directory:** C:/Program Files/Slicer 4.8.1

**GROUPS Parameters:**

- Select Properties to Use:** A table with two columns: 'Property' and 'Weight'. The first row shows 'C' with a weight of 5. The 'Weight' column has a red circle around it, indicating it is a numeric input field.
- ☐ Enable Use of Landmarks
- Degree of SPHARM Decomposition:** A slider set to 5.00.
- Maximum Number of Iterations:** 5000.

# Features and Parameters

## Groups Parameters tab

▼ GROUPS Parameters

Select Properties to Use

	Property	Weight
5	C	1

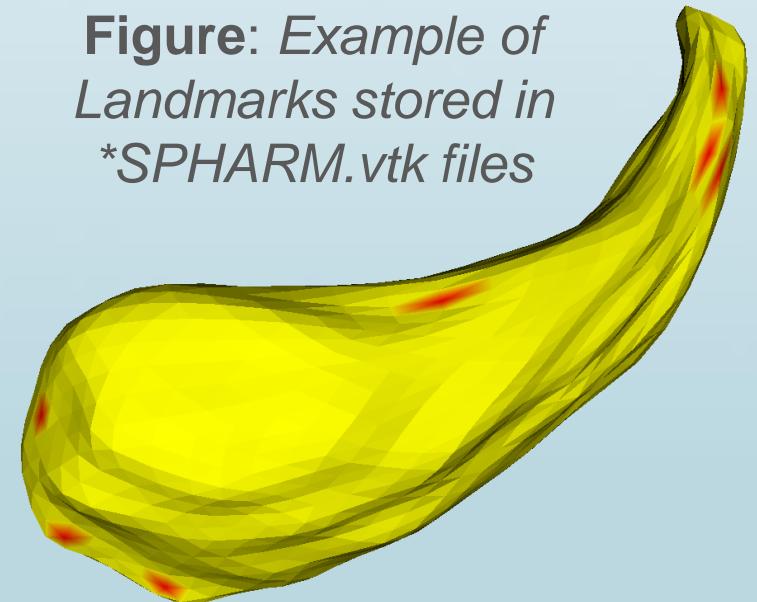
☒ Enable Use of Landmarks

Degree of SPHARM Decomposition  5.00

Maximum Number of Iterations

- Enable Use of Landmarks: Option for the user to select if landmarks will be included in improving the correspondence
- If enabled, the user need to store the landmarks as a point data array called "Landmarks" indicating Vertex IDs to be selected (Value > 0)

**Figure:** *Example of Landmarks stored in \*SPHARM.vtk files*



# Features and Parameters

## Groups Parameters tab

- Degree of SPHARM Decomposition: Degree value represents the degree of the spherical harmonic decomposition used to represent the computed deformation field
- Changing this value results in different levels of detail of the deformation field that will be used to transform the input SPHARM mesh

▼ GROUPS Parameters

Select Properties to Use	Property	Weight
5	C	1

☒ Enable Use of Landmarks

Degree of SPHARM Decomposition  5.00

Maximum Number of Iterations 5000

# Features and Parameters

## Groups Parameters tab

- Maximum Number of Iterations: Number of iterations before the energy minimization optimization stops
- A higher number of iterations usually needed with increasing the number of subjects in the dataset, number of properties selected for the optimization procedure, or with higher deformation field SPHARM degree

▼ GROUPS Parameters

Select Properties to Use

	Property	Weight
5	C	1

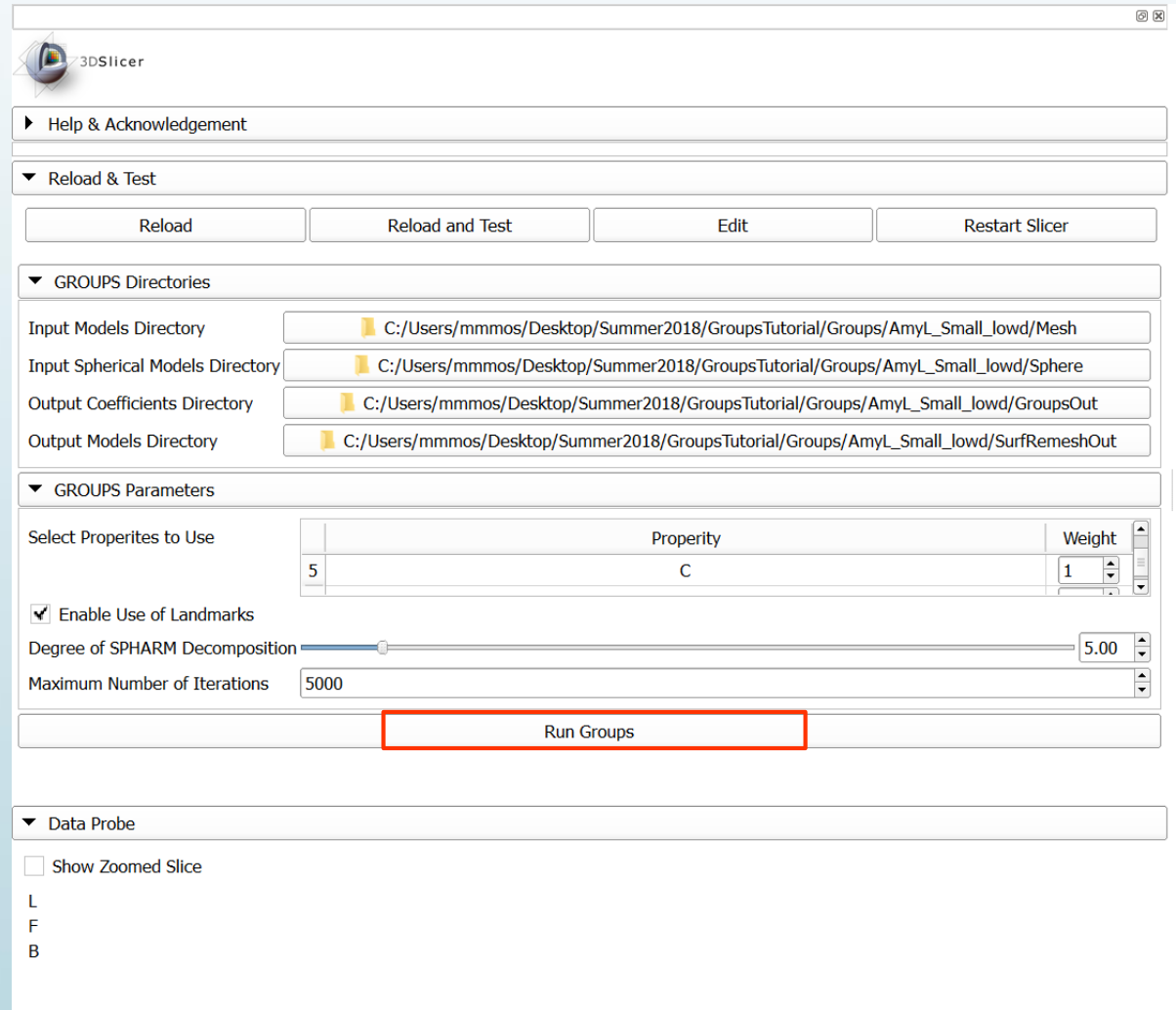
☒ Enable Use of Landmarks

Degree of SPHARM Decomposition  5.00

Maximum Number of Iterations

# Running Group-wise Registration Module

Click on the  
***Run Groups*** button,  
to run the CLIs on the  
provided inputs



3DSlicer

► Help & Acknowledgement

▼ Reload & Test

Reload Reload and Test Edit Restart Slicer

▼ GROUPS Directories

Input Models Directory C:/Users/mmmos/Desktop/Summer2018/GroupsTutorial/Groups/AmyL\_Small\_lowd/Mesh

Input Spherical Models Directory C:/Users/mmmos/Desktop/Summer2018/GroupsTutorial/Groups/AmyL\_Small\_lowd/Sphere

Output Coefficients Directory C:/Users/mmmos/Desktop/Summer2018/GroupsTutorial/Groups/AmyL\_Small\_lowd/GroupsOut

Output Models Directory C:/Users/mmmos/Desktop/Summer2018/GroupsTutorial/Groups/AmyL\_Small\_lowd/SurfRemeshOut

▼ GROUPS Parameters

Select Properties to Use	Property	Weight
5	C	1

☒ Enable Use of Landmarks

Degree of SPHARM Decomposition 5.00

Maximum Number of Iterations 5000

Run Groups

▼ Data Probe

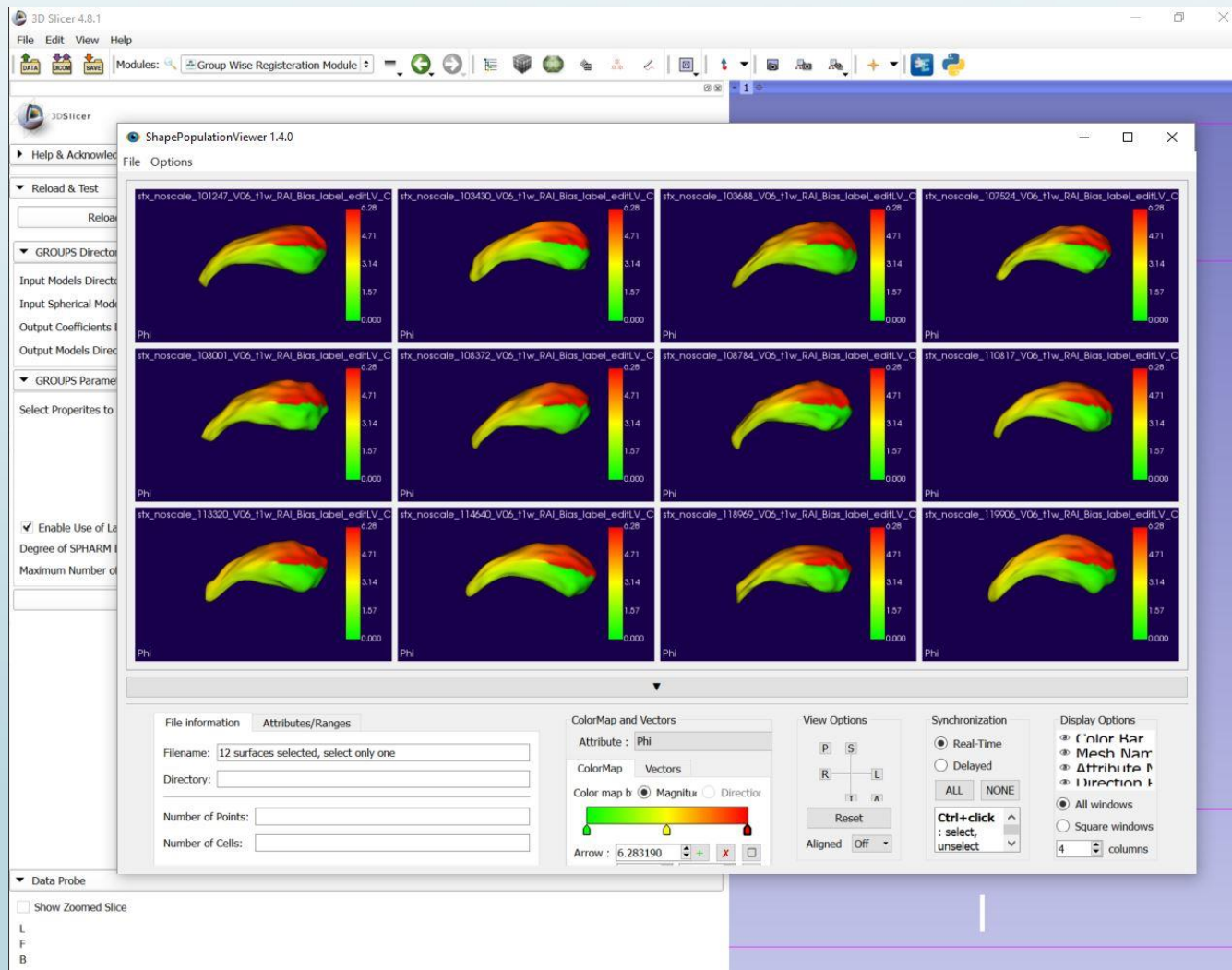
☐ Show Zoomed Slice

L  
F  
B



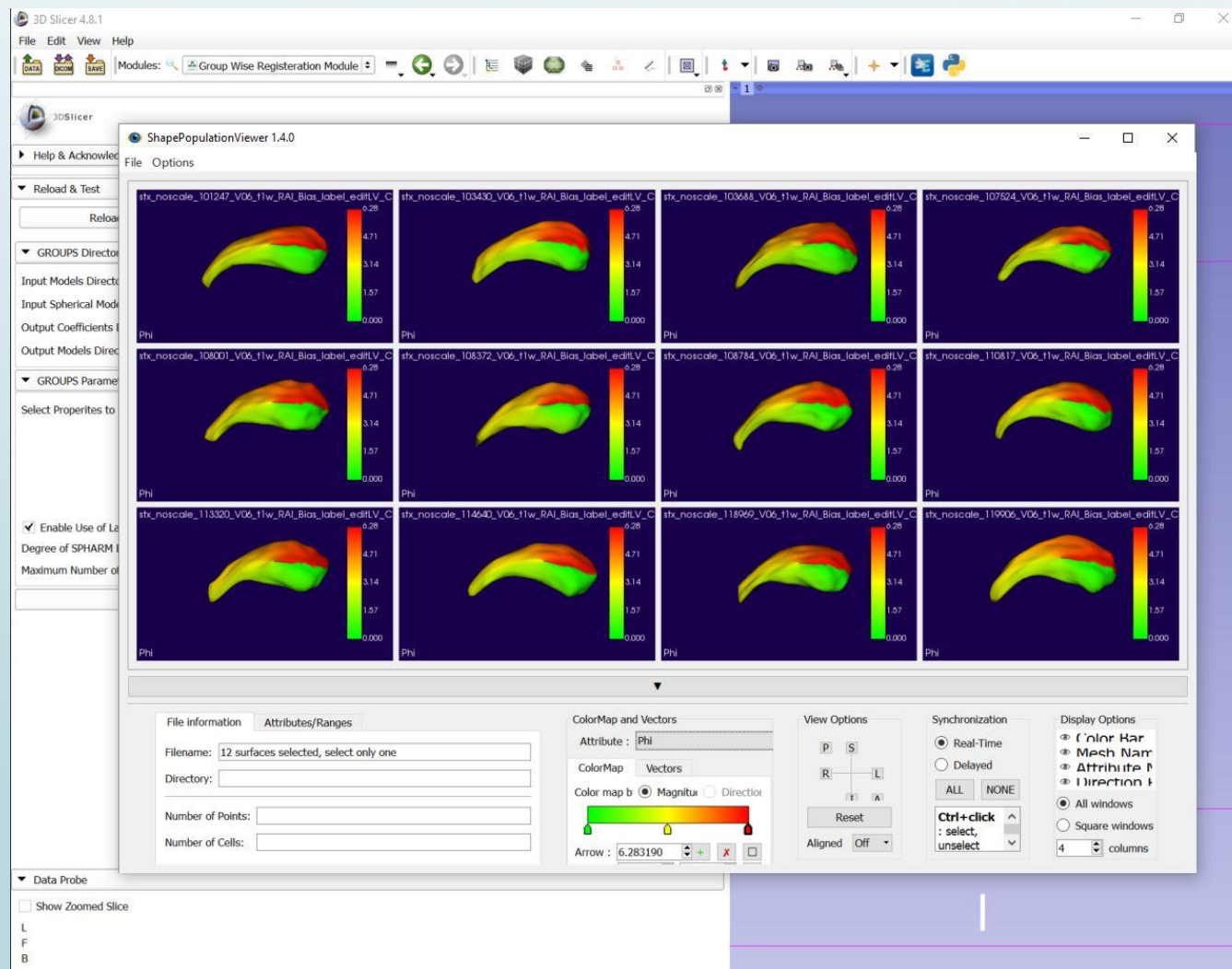
# Running Group-wise Registration Module

Shape Population Viewer will be used again for quality control of the input meshes before running the tool, in particular, features planned to be included should be inspected by the user carefully



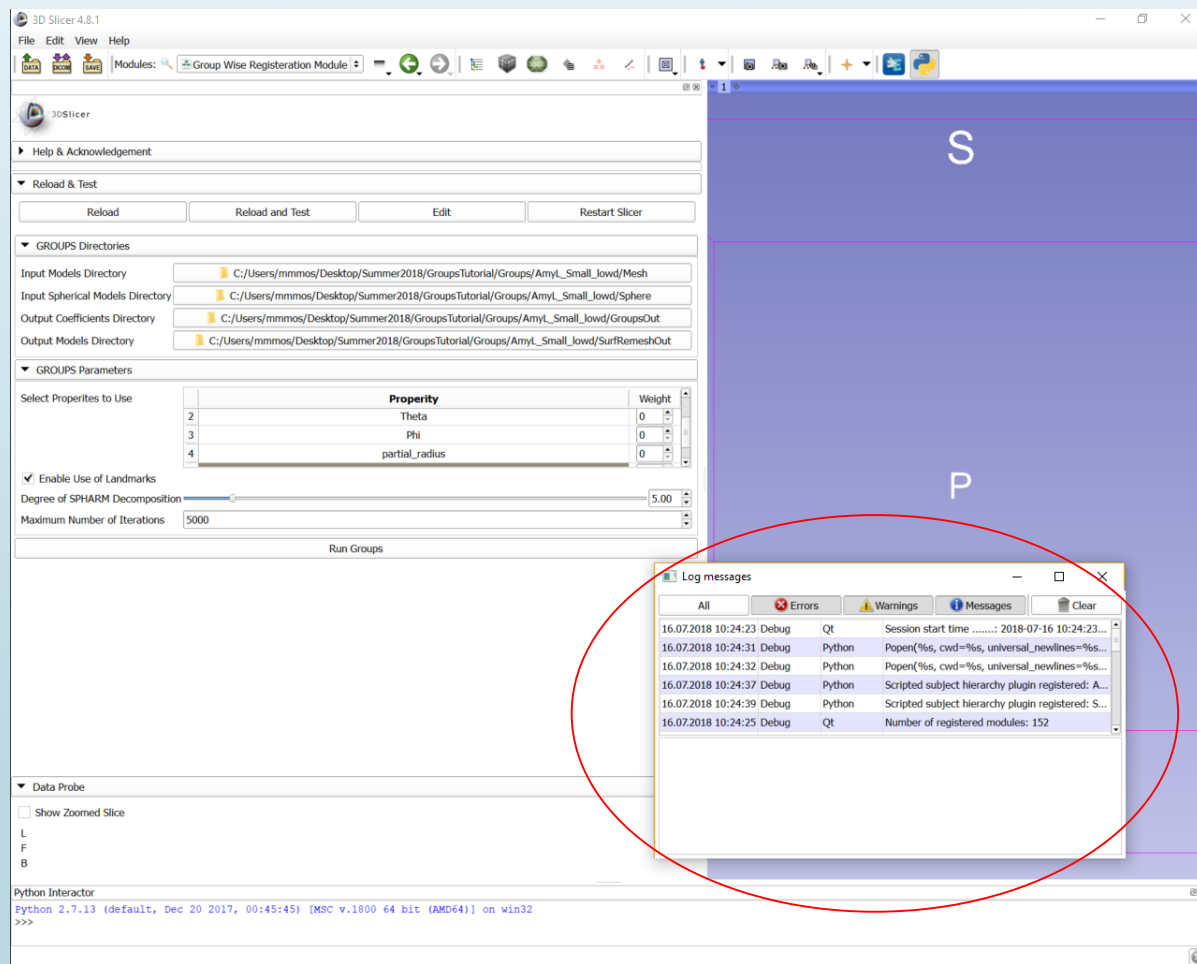
# Running Group-wise Registration Module

Also, Shape Population Viewer will be used to inspect the final correspondence established by the Group-wise Registration tool



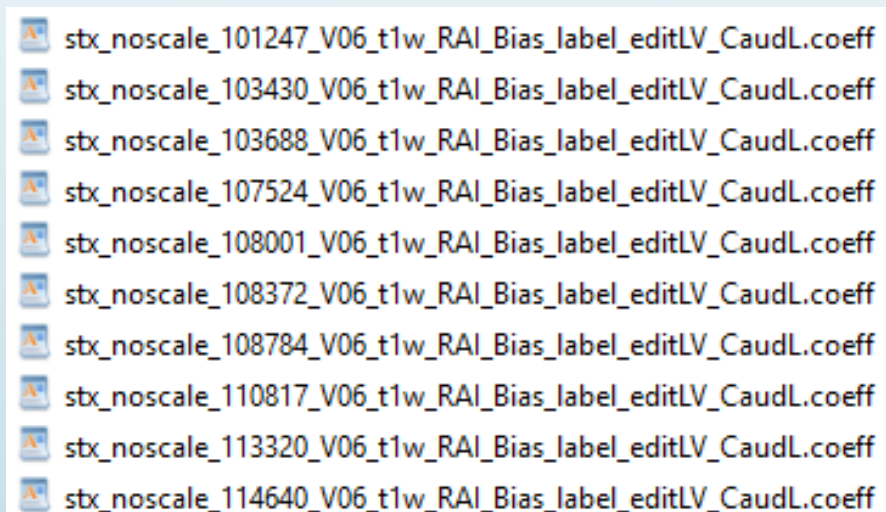
# Debugging

*3D Slicer's Error Log* can also be used for debugging if the module was completed with errors. To open it, click on the red icon at the bottom right.



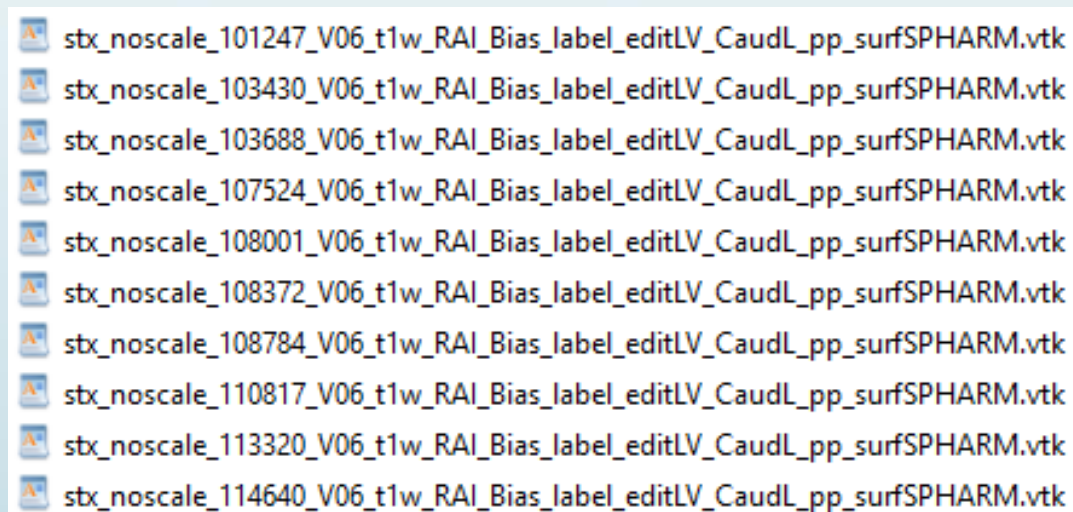
# Output Files

The outputs files for the two sub-steps of Group-wise Registration Module are stored in the two folders specified by the user:



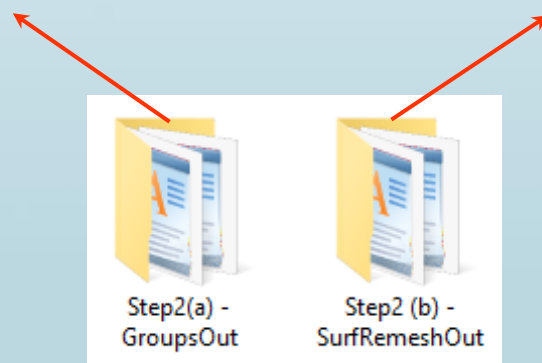
A list of 9 files with .coeff extensions, each preceded by a small icon of a document with a blue 'A' in the top left corner. The files are:

- stx\_noscale\_101247\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff
- stx\_noscale\_103430\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff
- stx\_noscale\_103688\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff
- stx\_noscale\_107524\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff
- stx\_noscale\_108001\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff
- stx\_noscale\_108372\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff
- stx\_noscale\_108784\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff
- stx\_noscale\_110817\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff
- stx\_noscale\_113320\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff
- stx\_noscale\_114640\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL.coeff



A list of 9 files with .vtk extensions, each preceded by a small icon of a document with a blue 'A' in the top left corner. The files are:

- stx\_noscale\_101247\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk
- stx\_noscale\_103430\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk
- stx\_noscale\_103688\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk
- stx\_noscale\_107524\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk
- stx\_noscale\_108001\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk
- stx\_noscale\_108372\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk
- stx\_noscale\_108784\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk
- stx\_noscale\_110817\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk
- stx\_noscale\_113320\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk
- stx\_noscale\_114640\_V06\_t1w\_RAI\_Bias\_label\_editLV\_CaudL\_pp\_surfSPHARM.vtk



# Acknowledgements - Resources - Questions

- The GROUPS developers gratefully acknowledge funding for this project provided by NIH NIBIB R01EB021391 (Shape Analysis Toolbox for Medical Image Computing Projects), as well as the Slicer community.
- Github repository:
  - ♦ [Groups](#)
  - ♦ [SlicerSALT](#)
  - ♦ [3D Slicer](#)
- Forums:
  - ♦ [SlicerSALT](#)
  - ♦ [3D Slicer](#)
- Papers:
  - [Robust estimation of group-wise cortical correspondence with an application to macaque and human neuroimaging studies](#)
  - [Group-wise shape correspondence of variable and complex objects](#)
- For other remarks or questions, please email:
  - [mahmoudm@cs.unc.edu](mailto:mahmoudm@cs.unc.edu)

