Date: 04/23/2008 Version 03
Authors: Ping Fu and Jennifer Staab, UNC-Chapel Hill

Functional Specifications for 3D Neural Activity Mapping

Introduction:

The most common method for 3D analysis of neural activity, functional MRI, cannot be applied to small animals because of the small size of the animal and the insufficient resolution of the images. As a result, many neurobiologists measure neural activity in small animals by sacrificing the animal, slicing its brain into a number of 2-dimensional slices which are then autoradiograph to create a sequence of 2-D images that represent the animal's neural activity. However, recent advances in image manipulation should enable the 3-D reconstruction of brain activity maps from the sequence of 2-D brain images. There are several published reports of methods that create 3-D maps from 2-D images, and at least one of these uses publicly available software, specifically Image and Turbojet (Nguyen P.T., 2004).

The techniques as outlined in the Nguyen paper cannot be directly used by our client because our client studies the neural activity of animals that have brains that are only a fraction of the size of those used in the Nguyen paper. Our client typically researches neural activity in frogs and fish which have brains that are much smaller than that of the rats used in the Nguyen paper. Therefore, our client must use a digital camera that is mounted on a compound microscope to digitize the 2-D slices that result from slicing up a small animal brain. She uses the 2-D cell images to determine the morphology of the animal brain so that the sequence of images can be aligned using the common features between adjacent cells image slices. Then she uses corresponding silver 'grains' images to show gene expression, a indicator of neural activity, in the 2-D brain slice images. Since the corresponding silver 'grains' images do not typically have common features between adjacent image slices, the alignment from the cells images must be applied to the corresponding silver 'grains' images.

Our goal is to develop a method for creating 3D activity maps from 2D brain images for very small animals like fish and frogs. These 3D activity maps will be created though a process of aligning the 2D images into a cohesive 3D structure and masking out the background noise (e.g. eliminating parts of the image that are not a part of the animal brain). Creating well aligned and properly masked 3D activity maps will allow researchers to use statistical software such as SPM (Statistical Parametric Mapping) to analyze the 3D data. This would enable researchers to study brain-wide neural activity patterns. If successful, the technique could revolutionize the researchers' understanding of auditory processing in frogs and, as a consequence, the evolution of the brain.

The software that was developed for this project must be used with in the Image software environment. Moreover, our software requires certain Image plugging in order to work. Some of these plugging are responsible for alignment (Turbojet, **MultiStackReg) and others are required for proper output of the aligned sequence of images (NIfTi). Specifically, the NIfTi plugin allows the user to import the resulting alignment into statistical software like SPM.

Software Support & Downloads:

Image Home Page: http://www.fil.ion.ucl.ac.uk/spm/

Date: 04/23/2008 Version 03
Authors: Ping Fu and Jennifer Staab, UNC-Chapel Hill

Image plugging:
TurboReg: http://bigwww.epfl.ch/thevenaz/turboreg/
**MultiStackReg: http://www.stanford.edu/~bbusse/work/downloads.html
NIfTi: http://rsb.info.nih.gov/ij/plugins/nifti.html

***Note that we provide an altered version of MultiStackReg called "MultiStackRegFix" that allows for a higher degree of error checking with regards to the alignment and allows for "MultiStackRegFix" to be seamlessly called in a batch mode fashion from the plugin and the macro.*

Documentation on ImageJ Developer's Resources:
ImageJ API Documentation: http://rsb.info.nih.gov/ij/developer/api/index.html
ImageJ Built-In Macro Functions: http://rsb.info.nih.gov/ij/developer/macro/functions.html

SPM software: http://www.fil.ion.ucl.ac.uk/spm/

Primary functions:

To reconstruct 3D images of small animal brains from 2D slice images, and to allow researchers to use this 3D reconstruction to better understand brain activity.

Prioritized requirements:

Below are the prioritized requirements of this project as determined through the initial discussions with our client. Items 1-4 deal with being able to mask and align a cells image sequence and being able to apply the transformation file and masking from the cells alignment to successfully mask and align the corresponding 'grain' image sequence. Items 5-7 deal with being able to provide a method that is faster than their current rate of analysis and a method that enables them to use SPM for statistical analysis of their data. Item 8 would just require that the final product be created in a user friendly format. All 8 of the requirements below were met.

1. Grayscale alignment of 2D images can be attained from in-order slice images that have features which are clearly distinguishable between adjacent slices (in the case of the Burmeister Lab, cells images should be used).
2. If images are not clear as defined above (such as grains images), alignment can be obtained by applying the same alignment from corresponding clear images which are in the same order and orientations.
3. For a brain region, be able to mask out incomplete information in clear (i.e. cells) images after alignment, and apply the same masking to corresponding unclear (grains) images.
4. For an entire brain, be able to mask out irrelevant information before alignment of clear (i.e. cells) images, and apply the same masking to corresponding unclear (grains) images.
5. 20 small animal brains of approximately 100 slices each should take less than 2 weeks to convert from 2D images to 3D reconstructions, including masking and alignment (how long our clients current process of animal brain analysis takes).
6. Have an output format which can be imported into statistical software such as SPM (Statistical Parametric Mapping).
7. Be able to automate so that one whole brain can be reconstructed without having user to manually input each 2D slice image. (Enable batch mode for around 100 slice images.)

8. Have a user-friendly GUI that is easy to understand for biologists, where the user goes through a wizard that implements batch masking and alignment, for both brain regions and an entire brain.

User Types:
Biology Lab Researchers and Technicians
*Functionality will be the same for all users

Scenario:

Christi is a lab technician working with 2D frog brain slice images. She has two sets of brain slice images of Hector, a particularly dumb frog (hence a very small brain), a set of cells images that do not vary greatly between adjacent slices, and a set of grains images that contain gene expression and typically vary greatly between adjacent slices. She knows the order in which the slice images come in, and she wishes to create a 3D reconstruction of the brain in order to see how the gene expression is distributed. Although the grains images are the ones she is interested in, they are very difficult to align. Therefore, cells images, which are easy to align, must be used to determine how each slice is transformed. After obtaining a 3D reconstruction of the cells images, the same transforms can then be applied to the corresponding grains images to obtain a 3D reconstruction of the grains images.

Christi has 100 cells images of Hector's brain, which are stored as c1.tif…c100.tif, with c1.tif being from the very front of Hector's brain. She opens the wizard and imports the images, masks out the background noise in her images, clicks "align", and gets back aligned images that are transformed (rotated and translated). She saves the aligned images in an SPM readable format, chooses to view the 3D projection of the images, and sees the 3D reconstruction rotated about the x and y-axes. She then chooses to mask and align the corresponding grains images, loads her 100 grains images, g1.tif…g100.tif, and obtains masked and aligned grains images, which she saves in an SPM readable format under the folder "Hector_grains". She then views the 3D projections as before, checks that they are okay, and closes the wizard. She can now import the saved results of the aligned grains to her SPM statistical software and analyze Hector's brain activity before his moment of sacrifice.

Use Cases

There are 4 use cases whose actions are detailed below. The flow diagram on page 6 shows an outline of the wizard windows the user would see as they progress through each of these cases. Note that use case I and II will flow through the entire diagram (starting at the top flowing down the left side and then continuing at the back top and flowing down the right side). Use case III and IV start at the top and only flow through the right side of the diagram. Below are the 4 use cases and the actions the user will go through to complete the alignment based on the case.

I. Aligning an entire brain(in color or grayscale) in (cells and grains) using TurboReg
II. Aligning a brain section in (cells and grains) using TurboReg
III. Aligning an entire brain(in color or grayscale) (grains) using saved transformations
IV. Aligning a brain section (grains) using saved transformation

I. Aligning an entire brain using Turbojet

1. The user loads clearly defined, in-order 2D images by specifying where the images are located and the following inputs: start number, end number, increment, contains and scale. “Start number” and “end number” indicate the first and last images in the sequence. “Increment” indicates by what increment (1 or greater) the images are imported into ImageJ. “Scale” indicates how much to reduce the resolution of the imported image sequence (100%= no resolution reduction). “Contains” indicates any common string contained in the file names of the image sequence (like ‘cells’).
2. Imported image sequence is masked by the user to block out the background.
3. User checks the masking of the brain image sequence and corrects any imperfections (failures in masking out the background) using the brush tool in ImageJ. Once imperfections are corrected user uses scroll bar to select most centered image in sequence prior to alignment.
4. Images are aligned using TurboReg and user checks alignment and decides whether to proceed forward or to correct the images using translation and rotation functions in ImageJ to allow for a better alignment of the images. If corrections are made then images are realigned (and the same corrections must be applied to corresponding grains image prior to their alignment).
5. Once the user has a satisfactory alignment they choose to either Save Results or Save Results and view 3D projections. If they decide to Save Results and view the 3D projections they are able to save the 3D projections once they appear.
6. Once the results have been saved the user is asked if he/she would like to align corresponding images (grains images).
7. User decides to align corresponding images and is prompted to select the where the corresponding images are stored and the corresponding image sequence is imported into ImageJ.
8. The user is prompted to apply any changes made to improve the alignment of the cells images (step 5) to the grains image sequence that has just been imported in.
9. Grains images are then masked and aligned and the user is asked if they want to Save Results or Save Results and view 3D projections. If the user decides to view 3D projections then they are able to save these 3D projections once they appear.
10. Once the aligned grains images are saved the user is prompted if they would like to start over or quit.

II. Aligning an brain section using Turbojet

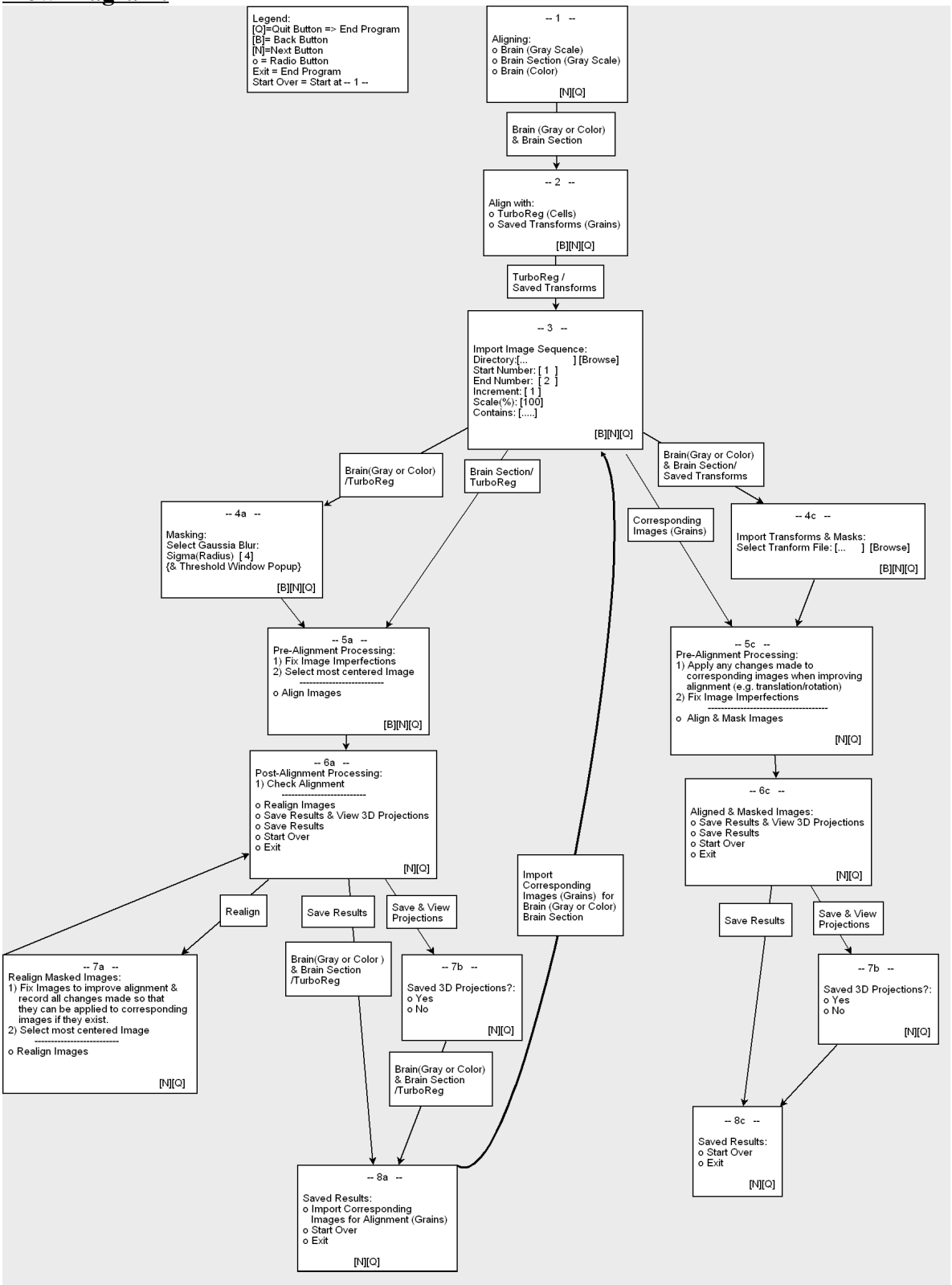
1. The user loads clearly defined, in-order 2D images by specifying where the images are located and the following inputs: start number, end number, increment, contains and scale. “Start number” and “end number” indicate the first and last images in the sequence. “Increment” indicates by what increment (1 or greater) the images are imported into ImageJ. “Scale” indicates how much to reduce the resolution of the imported image sequence (100%= no resolution reduction). “Contains” indicates any common string contained in the file names of the image sequence (like ‘cells’).

2. Images are aligned using TurboReg and then the aligned images are masked to block out areas that are not common for all the aligned images in the sequence. The user checks alignment and decides whether to proceed forward or to correct the images using translation and rotation functions in ImageJ to allow for a better alignment of the images. If corrections are made then images are realigned (and the same corrections must be applied to corresponding grains image prior to their alignment).
3. Once the user has a satisfactory alignment they choose to either Save Results or Save Results and view 3D projections. If they decide to Save Results and view the 3D projections they are able to save the 3D projections once they appear.
4. Once the results have been saved the user is asked if they would like to align corresponding images (grains images).
5. User decides to align corresponding images and is prompted to select the where the corresponding images are stored and the corresponding image sequence is imported into ImageJ.
6. The user is prompted to apply any changes made to improve the alignment of the cells images (step 3) to the grains image sequence that has just been imported in.
7. Grains images are then masked and aligned and the user is asked if they want to Save Results or Save Results and view 3D projections. If the user decides to view 3D projections then they are able to save these 3D projections once they appear.
8. Once the aligned grains images are saved the user is prompted if they would like to start over or quit.

III. & IV. Aligning an entire brain OR brain section (grains) using saved transformations

1. The user loads, in-order 2D corresponding images (grains) by specifying where the images are located and the following inputs: start number, end number, increment, contains and scale. Start number and end number indicates the first and last images in the sequence. Increment indicates by what increment (1 or greater) the images are imported into ImageJ. Scale indicates how much to reduce the resolution of the imported image sequence (100%= no resolution reduction). Contains indicates any common string contained in the file names of the image sequence (like ‘cells’).
2. The user is prompted to apply any changes that would improve the alignment of the cells images to the grains image sequence that was imported in.
3. The imported grains image sequence is masked using masks indicated by the user.
4. Grains images are aligned and the user is asked if they want to Save Results or Save Results and view 3D projections. If the user decides to view 3D projections then they are able to save these 3D projections once they appear.
5. Once the aligned grains images are saved the user is prompted if they would like to start over or quit.

Flow Diagram:



References:

1. Nguyen P.T., Holschneider D. P., Maarek J.M.I., Yang J., Mandelkern M.A. (2004). Statistical Parametric Mapping Applied to an Autoradiographic Study of Cerebral Activity during Treadmill Walking in Rats. *NeuroImage , 23* (1), 252-259.