Published in final edited form as:

Comput Methods Programs Biomed. 2009 August ; 95(2): 146–157. doi:10.1016/j.cmpb.2009.02.006.

STAMPS: software tool for automated MRI post-processing on a supercomputer

Don C. Bigler^{*,a,d,1}, Yaman Aksu^{*,b,d,1}, and Qing X. Yang^{**,c,d}

^aDepartment of Bioengineering, The Pennsylvania State University, Hershey, PA, USA

^bDepartment of Electrical Engineering, The Pennsylvania State University, University Park, PA, USA

^cDepartments of Radiology and Neurosurgery, The Pennsylvania State University, Hershey, PA, USA

^dPenn State Center for NMR Research, H066 NMR Building, 500 University Drive, Hershey, PA 17033, USA

Abstract

This paper describes a Software Tool for Automated MRI Post-processing (STAMP) of multiple types of brain MRIs on a workstation and for parallel processing on a supercomputer (STAMPS). This software tool enables the automation of nonlinear registration for a large image set and for multiple MR image types. The tool uses standard brain MRI post-processing tools (such as SPM, FSL, and HAMMER) for multiple MR image types in a pipeline fashion. It also contains novel MRI post-processing features. The STAMP image outputs can be used to perform brain analysis using Statistical Parametric Mapping (SPM) or single/multi image modality brain analysis using Support Vector Machines (SVMs). Since STAMPS is PBS-based, the supercomputer may be a multi-node computer cluster or one of the latest multi-core computers.

Keywords

nonlinear image registration; PBS; cluster; supercomputer; MRI; image processing

1. Introduction

Magnetic resonance imaging (MRI) has evolved as a superior tool to non-invasively study and monitor neurodegenerative diseases in-vivo; which is critical for developing therapies treating these diseases. This fact has led to the creation of large-scale, multi-center, neuroimaging initiatives such as the Alzheimer's Disease Neuroimaging Initiative (ADNI) [1] and the Open Access Series of Imaging Studies (OASIS) [2]. ADNI is a five year study of 200 elderly controls, 400 subjects with mild cognitive impairment, and 200 with Alzheimer's disease.

© 2009 Elsevier Ireland Ltd. All rights reserved.

*Corresponding authors. **Principal corresponding author, *Email addresses:* dcb210@psu.edu (Don C. Bigler), ya1@psu.edu (Yaman Aksu), qyang@psu.edu (Qing X. Yang).

¹Coauthors.

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Conflict of Interest Statement

None declared.

OASIS contains over 400 MRIs of subjects age 18 to 96, some of which have mild cognitive impairment. Extraction of useful information in a timely manner from such large image sets demands computing resources and organized patient database management. Many MRI post-processing tools (e.g. SPM [3], FSL [4], freesurfer [5], HAMMER [6,7], ITK [8]) are available for this purpose, but not all are designed for supercomputers² that drastically reduce processing time for large image sets. Also the listed tools do not perform all of the post-processing on all MRI types we discuss herein. Therefore, it is desirable to combine key functions of these tools into one image processing “pipeline” such that specific processing results are produced that are otherwise unattainable.

We present a software tool for automated post-processing of multiple types of brain MRIs on a workstation (STAMP) or on a supercomputer (STAMPS). The tool is designed to manage multiple types of image data using a few key attributes (e.g. patient ID, age) which a user can control in a central intuitive GUI independent of their values in image files.³ The tool then processes these images in a pipeline, using linear registration and segmentation tools in FSL and SPM5/VBM5 [9,10], or nonlinear registration tools in HAMMER. The MR image types that can currently be processed by the STAMP pipeline are T_2 and T_2^* maps, T_1 - and T_2 -weighted images, and all types of DTI images. Support for other image types may be added as needed. For some image types (e.g. T_2 , T_2^* maps, and DTI images), custom STAMP tools were developed using ITK for nonlinear registration (HAMMER) and ROI statistical analysis. These registered images may be used to perform voxel-based statistical analysis or to perform “machine learning” experiments (e.g. with Support Vector Machines (SVMs) [11]) which are capable of analyzing image types individually or jointly for biomarker discovery and validation.

Since the nonlinear registration task motivated STAMP, here we provide an introduction of this task and the rationale for using it. Due to intra- and inter-person anatomic variations, nonlinear registration of brain MRIs, as opposed to linear (affine) registration, is considered essential for morphological analysis. We use the HAMMER method [6] for several reasons. First, the HAMMER tool [6,7] is a freely available nonlinear brain registration method designed for high accuracy in anatomical correspondence. In this method, which is a hierarchical (coarse-to-fine) matching algorithm, the objective function (in the optimization algorithm) is successively approximated by lower-dimensional smooth functions constructed to have significantly fewer local minima. The method constructs them based on selecting driving anatomical features, represented as distinct “attribute” vectors, for the “matching”. The aim is to highly reduce ambiguity in finding correspondence. Second, HAMMER registered our data well and it was stable and reasonably fast as a rigorous nonlinear registration method (i.e. approximately 2 hours per image). The third column of Fig. 6 illustrates how well HAMMER was able to squeeze oversized ventricles (shown in the second column) into the atlas used for registration.⁴ Third, the HAMMER software generates a tissue (e.g. gray matter) density map named “RAVENS” map [12,13,14,15], with interesting properties for our image analyses. For example, by adjusting the density of a tissue whenever nonlinear registration expands or contracts the brain geometry, the RAVENS map preserves the total amount of tissue in any defined region, which sets it apart from density maps of other methods. Furthermore, a validation [12] supported that SPM voxel-based statistical analysis can be performed on these maps. The RAVENS maps can also be analyzed with machine-learning methods.

²In this paper “supercomputer” means a multi-node computer cluster using PBS or one of the latest multi-core computers using PBS.

³The data management is based on uniquely identifying images, their types, and their relations (Section 3.3).

⁴The subject used in the figures is a 79 year old male ALS subject chosen as a “worst-case scenario” registration example.

Section 2 describes the MRI post-processing. Section 3 describes the system architecture, including samples of program usage. The paper concludes with Section 4, which discusses system requirements and program availability.

2. MRI Post-processing

2.1. Overview

Fig. 1 shows the current input image types for STAMP. These image types (T_1 - and T_2 -weighted image, T_2 map, T_2^* map, DTI image) are each spatially in their own distinct “subject space”, but here it suffices to identify them with a shared “S” tag (i.e. “subject space”) as shown. STAMP outputs many image types. Among these, we first define some key ones below, and use Fig. 1 to group them spatially for clarity:

1. An image spatially normalized by deformable nonlinear registration using the HAMMER tool into an atlas space i.e. the “Hammer space” (“H”), identified as e.g. $I_{T_1,H}$ and $I_{T_2\text{map},H}$.
2. An image created by segmentation into numerous brain regions defined by an atlas, aka labeling. The labeling generates e.g. $I_{T_1,\text{labeled},S}$ from $I_{T_1,S}$.
3. RAVENS map for each of GM ($I_{\text{RAVENS_GM},R}$), WM ($I_{\text{RAVENS_WM},R}$), ventricles (V) ($I_{\text{RAVENS_V},R}$).⁵

The “A1” tag in Fig. 1 indicates some output images are spatially in a standard atlas space we identify as “Atlas1”, distinct from the atlas defined earlier for HAMMER use. By aligning all of its input image types into this standard A1 space (using rigid-body and affine transforms prior to HAMMER) STAMP ensures that the HAMMER registration will put all input image types into the same space. Note that the HAMMER tool supports the HAMMER registration of 8-bit unsigned integer images but not other datatypes such as 16-bit integer or 32-bit floating point (such as our T_2 map, T_2^* map, and DTI images), but STAMP does, which is a novel contribution. The HAMMER tool generates a deformation vector field file during the registration of an image and contains a program that can apply this file to identically deform (i.e. spatially normalize) a second image. We have developed an improved version of this program, based on ITK [8], for STAMP that supports signed and unsigned 8, 16, and 32-bit images and 32 and 64-bit floating point images.

For Atlas1, we chose the Montreal Neurological Institute (MNI)/International Consortium for Brain Mapping (ICBM) 152 atlas supplied by SPM5 and resampled to 1 mm isotropic spacing, to ensure correct alignment of the origin at the anterior commissure (AC). The placement of the origin at the AC is required for successful brain segmentation using SPM5/VBM5. STAMP uses FSL FLIRT [16] for this linear registration and the steps are given in the following subsections.

The atlas for the HAMMER tool (i.e. “Atlas2”) is supplied with this tool. It identifies 106 labeled brain regions, used for labeling. We flipped the atlas in the anterior-posterior direction to put it into the default transverse orientation for Analyze-format images.

2.2. Post-processing the T_1 -weighted images

We now discuss Fig. 2 which describes the details inside the STAMP box in Fig. 1 for the T_1 -weighted input. Since the processing for T_1 -weighted and T_2 -weighted images is identical, we will only discuss T_1 -weighted image as an example. The box places the T_1 -weighted image

⁵While RAVENS images are created by the HAMMER tool they are not in the H space, as the HAMMER tool uses a different nonlinear registration when creating RAVENS, hence our “R” tag to identify the “RAVENS space”.

from S space into A1 space and then into H space. These three images are illustrated by the top row of the 4×3 layout described in Fig. 6.

Automatic thresholding of the T_1 -weighted image is performed first. As performed in FSL BET [17], a lower intensity threshold, t_2 , is chosen such that 2% of the voxel intensities fall below it. Then, an upper threshold, t_{98} , is chosen such that 98% of the voxel intensities fall

below it. Then, a rough brain/non-brain lower threshold is set to $t = \frac{t_{98} - t_2}{10}$, and all voxel intensities below t are set to zero. In our experience this equation works well for images with adequate signal to noise ratio.

As shown, $I_{T_1,A1}$ is created from $I_{T_1,thresh,S}$ by linear registration to Atlas1. Next, $I_{T_1,A1}$ is tissue-segmented into GM, WM, and cerebrospinal fluid (CSF). Then, as required by HAMMER, if 2D slices (of this 3D image) are not 256×256 , zero-padding is used to make them 256×256 . Next, using HAMMER, CSF voxels in the ventricles are set to 50 in order to create $I_{T_1,segGWCv,A1}$, which is a segmented image with 150, 250, 10, and 50 for the GM, WM, CSF, and V tissues respectively.⁶ For the tissue segmentation tool, the STAMP user can select between SPM5/VBM5, FSL, or SPM5/VBM5 with cortical thickness estimation. The selected tool also performs bias field correction, which we identify (after ensuring 256×256 slices) as $I_{T_1,biascorr,A1}$. If SPM5/VBM5 is selected, STAMP generates a brain-only (and biascorrected) T_1 -weighted image ($I_{T_1,brainonly,A1}$) by masking this $I_{T_1,biascorr,A1}$ with the brain-only $I_{T_1,segGWCv,A1}$. If instead FSL is selected, FSL BET generates the brain-only T_1 -weighted image and FSL FAST [18] performs the tissue segmentation and bias correction. Finally, if SPM5/VBM5 with cortical thickness estimation is selected, STAMP uses the GM-only image to generate an image of estimated cortical GM thickness distribution ($I_{T_1,thick,A1}$) [19].

Having generated from the T_1 -weighted image both its brain-only version in Atlas1 space ($I_{T_1,brainonly,A1}$) - or estimated cortical thickness image in Atlas1 space ($I_{T_1,thick,A1}$) - and its tissue-segmented image in Atlas1 space ($I_{T_1,segGWCv,A1}$), the HAMMER tool can now be used. As shown, the HAMMER tool uses one or both of these images depending on the task. It uses both to generate $I_{T_1,H}$ but labeling and RAVENS generation do not use $I_{T_1,brainonly,A1}$ as input. By definition, the voxel intensities of the RAVENS image set (i.e. the set of three types of RAVENS images) sum to the patient's segmented brain volume prior to the generation of the RAVENS image. The sum is on the order of 1×10^6 and varies across patients. To remove patient brain volume as a bias from RAVENS images for subsequent voxel-based analysis, STAMP provides an option for normalizing the RAVENS image set such that each patient would have the same sum.

After registering an image by HAMMER to generate e.g. $I_{T_1,H}$, STAMP has options to create additional output images from it. The image can be cropped (to remove non-brain background) to create e.g. $I_{T_1,c,H}$ from $I_{T_1,H}$, which is then intensity-normalized to create e.g. $I_{T_1,cn,H}$. Custom command-line tools built using ITK crop the image using the dimensions of a template image. For the intensity normalization, the image is thresholded by $\pm 3\sigma$ of its voxel intensities, then multiplied by $32760/(imaximin)$, where $imax$ is the maximum intensity and $imin$ is the minimum intensity after thresholding, and then saved as a 16-bit signed integer image.

⁶In our experience, this step may require manual relabeling for subjects with ventricles that are morphologically very different from the atlas.

2.3. Post-processing the DTI images, T_2 maps, and T_2^* maps

2.3.1. Introduction—There are commonalities among the STAMP post-processing of DTI images, T_2 maps, and T_2^* maps. First, placing one of these three into A1 space is performed differently than in the earlier T_1 -weighted case, as it is more important for these images to be aligned with the T_1 -weighted⁷ image in A1 space rather than with only Atlas1 itself. How to achieve this alignment is different for DTI images than it is for the maps. However, a commonality of the two cases is that ultimately a linear (affine) transformation T_{S-A1} (from S space to A1 space) is created via a registration to a T_1 -weighted image in A1 space (using FSL FLIRT) and applied to the S-space image (i.e. $I_{DTI_*,S}$ or $I_{T_2map,S}$ or $I_{T_2^*map,S}$) to place it into A1 space (thereby respectively generating $I_{DTI_*,A1}$ or $I_{T_2map,A1}$ or $I_{T_2^*map,A1}$, each of which is both in A1 space and aligned with the T_1 -weighted image).

Second, the STAMP post-processing for the above three types contains a large final identical segment. This segment is shown in Fig. 3. Applying T_{S-A1} (as described above) is its first step. The rest of this segment performs two tasks.

The first task is the creation of the HAMMER-labeled image in S space. This task is performed when T_{S-A1} is inverted to obtain the transformation T_{A1-S} (from A1 space to S space) which is then applied to the previously generated $I_{T_1,labeled,A1}$ to place it into S space (thereby generating $I_{*,labeled,S}$). Since STAMP relies on FSL FLIRT to perform these operations, the final image resolution, spacing, and slices of $I_{*,labeled,S}$ will match $I_{*,S}$.

The second task is HAMMER registration which is performed in two steps: 1) The A1-space image (generated above, e.g. $I_{DTI_FA,A1}$) is easily masked with $I_{T_1,segGWCv,A1}$ to generate the brain-only image in A1-space e.g. $I_{DTI_FA,brainonly,A1}$. 2) The resulting (A1-space and brain-only) image (e.g. $I_{DTI_FA,brainonly,A1}$) goes under HAMMER registration, which is performed differently than in the earlier T_1 -weighted case. The deformation field file generated earlier by the HAMMER registration of the T_1 -weighted image in A1-space is now applied to $I_{DTI_FA,brainonly,A1}$ to achieve the HAMMER registration, thereby generating $I_{DTI_FA,H}$. This operation is made possible by the aforementioned custom ITK command-line programs developed for STAMP.

2.3.2. Post-processing the DTI images—We now discuss Fig. 4 which describes the details inside the STAMP box in Fig. 1 when the input is a DTI image set. The approach taken here is to generate T_{S-A1} (Sec. 2.3.1) as the linear (affine) transformation that would align the low-resolution DTI B0 image ($I_{DTIB0,S}$) (i.e. the input image) to the higher-quality⁸ T_1 -weighted image $I_{T_1,brainonly,A1}$ (i.e. the reference image)⁹. The DTI B0 image, which is a T_2 -weighted image, is chosen here because it is the most suitable DTI image type for linear registration to a T_1 - or T_2 -weighted image. The other DTI images (input to STAMP) highlight the white matter fibers and are therefore not suitable for linear registration to a T_2 - or T_1 -weighted image. Also, by using the DTI B0 image STAMP is not restricted to processing a particular DTI image type, such as FA.

After linear registration to the higher-quality T_1 -weighted image, the rest of the images in the DTI image set, one at a time, are input along with T_{S-A1} into STAMP (as shown in Fig. 3) to

⁷For the DTI images, T_2 maps, and T_2^* maps STAMP will automatically choose the T_2 -weighted image in A1 space for alignment if it is available, otherwise it will use the T_1 -weighted image in A1 space.

⁸Higher quality means better tissue contrast and higher resolution.

⁹Since the skull signal intensity in a typical DTI B0 image is usually on the level of the background noise or removed during DTI processing (prior to STAMP), the proper reference image to be used here is the skull-stripped T_1 -weighted image i.e.

$I_{T_1,brainonly,A1}$.

generate outputs as described by Sec. 2.3.1. For example, if the DTI image is DTI FA ($I_{DTI_FA,S}$), the output images will include $I_{DTI_FA,A1}$ (which is both in A1 space and aligned with the T_1 -weighted image), the labeled image $I_{DTI_FA,labeled,S}$, and the image $I_{DTI_FA,H}$ upon HAMMER registration.

2.3.3. Post-processing the T_2 maps and T_2^* maps—Processing of T_2 (or T_2^*) maps presents some special problems that alter the registration to A1 space as described for DTI images. Fig. 5 shows the flow diagram for these image types. As acquired in our laboratory, T_2 maps include only a part of the brain. In order to properly register these partial brain volume images to $I_{T1,A1}$ via FSL FLIRT linear registration, we use the higher-quality whole-brain image (i.e. $I_{T1,A1}$) as the “input image” instead of the “reference image”. This generates a linear (affine) transformation, which STAMP inverts to obtain T_{S-A1} (Sec. 2.3.1). Then, the T_2 (or T_2^*) map of partial brain volume in S space (i.e. $I_{T2map,S}$ or $I_{T2^*map,S}$), is input along with T_{S-A1} into Fig. 3 to generate image outputs as described by Sec. 2.3.1. Sometimes the above “reverse transform” procedure is not sufficient to successfully register the whole brain image $I_{T1,A1}$ to the partial brain image $I_{T2map,S}$ (or $I_{T2^*map,S}$). In this case, STAMP provides an option that will create a weighting image from $I_{T1,segGWCV,A1}$ that doubles the weighting of the ventricles over the other brain tissues during registration and can be assigned as an input to FSL FLIRT. This option enables successful registration of $I_{T1,A1}$ to $I_{T2map,S}$ (or to $I_{T2^*map,S}$).

2.4. Summary statistics

STAMP also generates summary statistics for all of the supported image types using brain regions identified in Atlas2. Custom command-line tools were developed using ITK that take as input an image in A1 or S space ($I_{*,A1}$ or $I_{*,S}$), its labeled version in the same space ($I_{*,labeled,A1}$ or $I_{*,labeled,S}$)¹⁰, the brain regions’ text descriptions and intensities (labels) in the labeled image, and outputs a statistics text file (i.e. a spreadsheet in comma separated value (CSV) format) containing statistics for all brain regions. A CSV file will be generated for each input image type. The chosen statistic varies depending on the input image type. If the input image is $I_{T1,segGWCV,A1}$, the file will contain the brain region volumes as well as the whole brain volume (both in mm^3). If the input is $I_{*,brainonly,A1}$, the file will contain, for each brain

region, the voxel frequency (n), the image intensity mean ($\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$), an image intensity

variance ($s^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2$),¹¹ the image intensity median, first quartile, third quartile, maximum, and minimum. Lastly, if the input image is $I_{T1,thick,A1}$, the file will contain the average thickness for each brain region. Since STAMP is designed to handle large image sets, it also generates summary statistics (CSV) files that will combine the individual statistics based on brain region and image type. One summary file is generated for each brain region and contains all statistics for each input image of a given type.

2.5. Illustration of registration results

Fig. 6 shows three orthogonal views of (a) T_1 -weighted, (b) DTI FA, (c) T_2^* map, and (d) T_2 map images in S space, A1 space, and H space for a 79 year old male subject. For the T_1 -weighted image, the rigid-body transform from S space to A1 space merely rotates and translates the image. In this case, each image in A1 space has the same shape and size as the image in S space. For the other three image types shown in the Figure, the A1 image is a result

¹⁰The labeling is performed in A1 space for T_1 - and T_2 -weighted images and in S space for all the others.

¹¹Note the division is by n , not by $n - 1$.

of alignment to the A1 T1-weighted via an affine (rather than rigid-body) transform wherein scaling and shearing in addition to rotation and translation are applied. Each image in A1 space no longer has the same shape and size as the image in S space. The nonlinear registration to H space alters the shape and size of A1-space images by nonlinear deformations.

Fig. 7(a–c) shows the images of (a) DTI FA in H space, (b) T_2^* map in H space, and (c) T_2 map in H space overlaid on the T_1 -weighted image in H-space of the same subject in Fig. 6. The lookup table used for the overlays maps is blue to yellow for the minimum to maximum intensity. This figure demonstrates the excellent registration between the images after nonlinear registration to the HAMMER atlas. For the DTI FA image in Fig. 7(a), the low FA values in the cortex and mid-brain GM structures show up as blue and overlap well with the dark grayscale intensities of GM in the T_1 -weighted image. As expected, the high FA values in the white matter appear as red and yellow over the light grayscale intensities of white matter in the T_1 -weighted image. Although not quite as obvious as the DTI FA image, the T_2^* and T_2 maps in Fig. 7(b) and Fig. 7(c) are also well aligned with the T_1 -weighted image. As GM/WM contrast is small in the T_2^* and T_2 maps and their values are significantly lower than that of CSF, the overlays appear mostly blue on the T_1 -weighted image overlay. However, examining the ventricles in yellow and the T_2^* and T_2 of CSF shows that they co-registered well.

Fig. 8 illustrates the qualities of both nonlinear (HAMMER) and linear registration (FSL FLIRT) in STAMP using T_1 -weighted images acquired from two subjects with vastly different brain anatomy. The subjects used for the overlay are the same 79 year old male and a 37 year old male. The same overlay color lookup table is used in Fig. 7. In this extreme case, even after considerable shrinkage of the ventricles, as shown by the outlined regions in A1 and H space for the T_1 -weighted image in Fig. 6, registration errors still exist in the areas of the ventricles outlined in green in the axial and coronal views of the nonlinearly registered example in Fig. 7. Also, as outlined in the coronal view, some misregistration exists in the area of the cortex. However, as seen in the figure, the registration error is much larger in the linearly registered case.

3. Architecture

3.1. Overview

We have developed a software library named *L* (i.e. *Library*), on which STAMP is based. *L* is in the tcl/tk language and contains a wide range of functions. Functions exist for controlling the invocation of external standard MRI tools used by STAMP (e.g. FSL, SPM5, HAMMER) and tools we developed for STAMP. Functions also exist for constructing and managing our in-memory-only database for uniquely identifying images and their relations during runtime.

3.2. Parallel Processing

To simultaneously process multiple input images, by distributing this task to multiple processors of a supercomputer, we developed a parallel processing version of STAMP, named STAMPS. Since it typically takes over 2 hours (for FSL, SPM5, HAMMER combined) to process just one input image to generate output images in H or R space, the distributed computing provided by STAMPS to drastically reduce total processing time for the image set is very significant. STAMPS invokes the PBS batch queueing system. It automatically creates PBS job scripts and submits them to PBS, using automatic decision making based on its monitoring of node availability and of the processing status of individual input images. The following pseudo-code demonstrates this process:

```
find total available free nodes using 'pbsnodes -a';
foreach image in data-base do
```

```

get current node;
create job script for this image for current node, wall time, and 1 processor;
if not managed by PBS scheduler then
    tell script to wait for previous job on current node;
end
write job script to file;
execute qsub with job script file as argument;
increment to next node;
if reached last node or user defined number of nodes then
    begin at first node again;
end
end

```

3.3. Data File I/O and Data Object Management

During runtime, STAMP populates a database-like object instance, named *db*, storing key attributes (e.g. demographical and scan identification) for identifying images during the processing flow. This instance is not stored in a file and resides in memory only. The Analyze file format (i.e. pair of files *filename.img*, and *filename.hdr*), currently required for input image files, is limited as it cannot store some of these attributes. Thus, for the program to construct and use the *db*, it must read such attributes from other input data sources. For the program's startup task of reading all input data sources to initially populate the *db*, the user can select one among distinct modes that allow reading and interpreting different sources. The selected mode also serves as a program operation mode, as there exists a Basic Mode that forgoes various STAMP processing capabilities for the sake of enabling the user to input only image files (and no other input sources). More specifically, since the limited Analyze format by default lacks provisions for identifying relationships among different MR image types (for a given person), this mode cannot and does not properly construct the *db* which is essential for full STAMP processing capabilities. Before using the Enhanced Mode, for each input image (*filename.img*) the STAMP user is required to create a text file (*filename.csv*) in comma separated value (CSV) format with values of a small set of (8) STAMP-required attributes: Patient ID, Patient Age, Patient Sex, Patient Group¹², Study ID, Image Type, Sequence Name, Acquisition Date. The CSV file creation task is convenient and quick using the CSV Writer GUI in STAMP, as the user can specify an attribute value and assign it to a (user-selected) group of image files in one step. This is described by Fig. 9. The LONI Mode and the OASIS Mode are STAMP operation modes designed to interpret the customized data bundles of these two data providers.¹³

3.4. System Monitoring and Log Files

STAMP and STAMPS create log text files of many types. For example, there are log files for each top-level UNIX shell and tcl script (Sec. 3.5) and separate log files for the post-processing on the separate supercomputer "nodes". The logs indicate in detail the invocation of *L* functions

¹²Categorical attribute, e.g. AD, Control.

¹³These bundles contain customized supplementary files alongside image files, serving a purpose akin to that of CSV files of the Enhanced Mode.

and the tools (e.g. FSL, SPM5, HAMMER), with their input parameter values and date and time stamps.

3.5. Samples of Program Runs

To run STAMPS, the user sets a few L UNIX environment variables, prepares a text file of post-processing configuration parameters (*pipeline.config*) and a text file for the “operation mode” (Sec. 3.3) e.g. *enhanced.config*, and passes their directory to a UNIX shell script: *pbs pipeline all.sh /home/foo/config_files* Likewise, to run STAMP, the user invokes a similar UNIX shell script. In turn, the shell script executes a tcl script for invoking various L functions.

4. System Requirements and Availability

STAMP and STAMPS are implemented as described and functioning. Regarding their use by others, we believe they are currently insufficiently documented. To receive updates on their availability, contact us.

STAMP and STAMPS can currently run on Linux only. Their architecture is readily suitable for Windows and Mac as it is written in tcl/tk. Moreover, since STAMPS uses PBS, the use of STAMPS on PBS-based non-Linux supercomputers is a worthwhile consideration. However, we note that not all tools in STAMP from external sources, such as HAMMER, are available for all major operating systems.

5. Discussion

The previous sections described STAMP and STAMPS. In this section we would like to briefly discuss directions for future work.

5.1. Longitudinal RAVENS Analysis

One of the STAMP-required attributes is Acquisition Date. Currently STAMP or STAMPS is not capable of using this field to process repeated scans of the same subject for longitudinal studies in a meaningful way. An example of where this might be advantageous is a longitudinal RAVENS analysis. As described previously, by adjusting the density of a tissue whenever nonlinear registration expands or contracts the brain geometry, the RAVENS map preserves the total amount of tissue in any defined region. Therefore, the target atlas used during nonlinear registration determines whether a given region expands or contracts during RAVENS generation. Then, for longitudinal RAVENS analysis, ideally the target atlas (Atlas2) for the repeated scans should be the baseline scan. In order to perform subsequent group analysis using SPM or SVM, the baseline scan for each subject would first be registered to a standard atlas using HAMMER (thereby generating a HAMMER deformation field in H space). Afterwards to generate RAVENS maps, all of the images for a given subject, including the baseline scan, are fed through STAMP using the baseline image as the atlas. Finally, the RAVENS maps are registered to the atlas for group analysis by applying the HAMMER deformation field generated for a subject baseline image to all of the longitudinal RAVENS maps generated for that subject. At the moment these lengthy operations must be performed manually in STAMP and STAMPS, but could be incorporated into future versions with minimal work.

5.2. Custom Population-based Atlas Creation

Within the field of whole-brain voxel-based group analysis, it is common to create population based atlases to reduce registration errors due to using an atlas that is potentially very different anatomically from the subject population. In the case of STAMP and STAMPS, its reliance on HAMMER, which uses a segmented atlas, complicates the process of creating a population based atlas. However, a simple technique for creating such an atlas can be performed using

SPM5/VBM5. SPM5/VBM5 provides a “labeling” function that will output a hard-labeled segmented brain image using the GM, WM, and CSF probability images as input. To create a custom population-based atlas, SPM5/VBM5 is first used to create registered GM, WM, and CSF probability images for all of the subjects in the patient population. The registration atlas could be the atlas supplied with SPM5 or any other atlas. Then the registered probability images are averaged across the group, such that one average registered GM, WM, and CSF probability image exists. Next SPM5/VBM5 is used to create the hard-labeled, average, registered image. Finally, since HAMMER expects the segmented labels to be 150, 250, 10, and 50 for the GM, WM, CSF, and V tissues respectively, HAMMER must be used to relabel the image as shown in Fig. 2 and discussed in Sec. 2.2. Fig. 10 shows examples of the average GM, WM, and CSF probability images and the final labeled atlas. Currently, STAMP or STAMPS does not automate this task. Again, future versions could provide this functionality with minimal work.

5.3. Novel Registration Techniques

Finally, STAMP and STAMPS relies on HAMMER and FSL for the various registration tasks. Future versions might provide options for using other novel registration techniques not currently supported. For example, demons non-linear registration [20] or perhaps using a DTI based [21] registration technique to drive the registration. Incorporating new registration techniques would require more work and testing than the two previously mentioned ideas, but is a relatively simple task given the pipeline nature of STAMP and STAMPS.

Acknowledgments

This work was supported in part by the NIH (RO1 EB00454 and 1R01AG027771-01A2), The Zimmerman Family Love Fund, The Paul and Harriett Campbell ALS Research Fund, and The ALS Association, Greater Philadelphia Chapter.

References

1. Alzheimer's Disease Neuroimaging Initiative (ADNI). <http://www.adni-info.org>
2. Marcus DS, Wang TH, Parker J, Csernansky JG, Morris JC, Buckner RL. Open Access Series of Imaging Studies (OASIS). Cross-sectional MRI Data in Young, Middle Aged, Nondemented, and Demented Older Adults. *Journal of Cognitive Neuroscience* 2007;19:1498–1507. [PubMed: 17714011]
3. Statistical Parametric Mapping (SPM). <http://www.fil.ion.ucl.ac.uk/spm>
4. FMRIB Software Library. <http://www.fmrib.ox.ac.uk/fsl>
5. freesurfer software. <http://surfer.nmr.mgh.harvard.edu>
6. Shen D, Davatzikos C. HAMMER: hierarchical attribute matching mechanism for elastic registration. *IEEE Transactions on Medical Imaging* 2002;21:1421–1439. [PubMed: 12575879]
7. HAMMER software. <http://www.rad.upenn.edu/sbia/software/index.html>
8. NLM Insight Segmentation and Registration Toolkit (ITK). <http://www.itk.org>
9. SPM5. <http://www.fil.ion.ucl.ac.uk/spm/software/spm5>
10. VBM5. <http://dbm.neuro.uni-jena.de/vbm/vbm5-for-spm5>
11. Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines*. Cambridge UK: Cambridge University Press; 2000.
12. Davatzikos C, Genc A, Xu D, Resnick SM. Voxel-Based Morphometry Using the RAVENS Maps: Methods and Validation Using Simulated Longitudinal Atrophy. *NeuroImage* 2001;14:1361–1369. [PubMed: 11707092]
13. Goldszal AF, Davatzikos C, Pham DL, Yan MXH, Bryan RN, Resnick SM. An Image-Processing System for Qualitative and Quantitative Volumetric Analysis of Brain Images. *Journal of Computer Assisted Tomography* 1998;22:827–837. [PubMed: 9754125]
14. Davatzikos C. Mapping image data to stereotaxic spaces: Applications to brain mapping. *Human Brain Mapping* 1998;6:334–338. [PubMed: 9788069]

15. Shen D, Davatzikos C. Very High-Resolution Morphometry Using Mass-Preserving Deformations and HAMMER Elastic Registration. *NeuroImage* 2003;18:28–41. [PubMed: 12507441]
16. Jenkinson M, Smith SM. A global optimisation method for robust affine registration of brain images. *Medical Image Analysis* 2001;5:143–156. [PubMed: 11516708]
17. Smith SM. Fast robust automated brain extraction. *Human Brain Mapping* 2002;17:143–155. [PubMed: 12391568]
18. Zhang Y, Brady M, Smith S. Segmentation of brain MR images through a hidden Markov random field model and the expectation maximization algorithm. *IEEE Transactions on Medical Imaging* 2001;20:45–57. [PubMed: 11293691]
19. Bigler, DC.; Yang, QX.; Smith, MB. A Simple Method for the Automatic Measurement of Cortical Thickness. ISMRM 14th Scientific Meeting & Exhibition; Seattle, Washington, USA. 2006.
20. Thirion JP. Image matching as a diffusion process: an analogy with Maxwell's demons. *Medical Image Analysis* 1998;2:243–260. [PubMed: 9873902]
21. Hui Z, Avants BB, Yushkevich PA, Woo JH, Wang S, McCluskey LF, Elman LB, Melhem ER, Gee JC. High-Dimensional Spatial Normalization of Diffusion Tensor Images Improves the Detection of White Matter Differences: An Example Study Using Amyotrophic Lateral Sclerosis. *IEEE Transactions on Medical Imaging* 2007;26:1585–1597. [PubMed: 18041273]

Fig. 1.

Overall flow diagram showing how STAMP takes the various input image types and generates the labeled image in subject space, normalized image in HAMMER space, and the RAVENS image.

Fig. 2.
Post-processing in STAMP for T_1 -weighted images.

Fig. 3.
The final identical segment of the STAMP post-processing for DTI images, T_2 maps, and T_2^* maps.

Fig. 4.
Post-processing in STAMP for DTI images.

Fig. 5.
Post-processing in STAMP for T_2 maps and T_2^* maps.

Fig. 6.

This 4×3 layout (of orthogonal views) illustrates STAMP input image types for a 79 year old male subject in Subject (S) space and their output version generated by STAMP in Atlas1 (A1) space and HAMMER (H) space. The input images types, shown along the first column (top to bottom), are (a) T_1 -weighted, (b) DTI FA, (c) T_2^* map, and (d) T_2 map. The second and third columns show these image types in A1 space and H space, respectively. The ventricles of the T_1 -weighted image are outlined in both A1 space and H space to demonstrate the dramatic size alteration that is needed to match this subject with the HAMMER atlas.

Fig. 7.

For the same subject shown in Fig. 6, the (a) DTI FA H space image, (b) T_2^* map H space image, and (c) T_2 map H space image are overlaid on the T_1 -weighted image in H-space. The color lookup table used for all the maps consist of a gradient from blue to yellow, with red in the middle.

Fig. 8.

Coronal (top) and axial (bottom) views of the 79 year old male T_1 -weighted image with enlarged ventricles shown in Fig. 6 and Fig. 7 is overlaid on a 37 year old male T_1 -weighted image without enlarged ventricles. The overlays are shown for both nonlinear (HAMMER) and linear (FSL FLIRT) registration to the HAMMER atlas. The circled regions highlight areas of poor registration performance. The color lookup table used for the overlay is a gradient from blue to yellow, with red in the middle.

[illegible]

CSV Writer in STAMP, used with STAMP's Enhanced Mode. Into the middle browser, the user accumulates content for multiple CSV files to be created (where each item (line) in that browser will accumulate content for one CSV file) and then creates all files via a single button press. The accumulation step is convenient and quick, as each Select button and each Apply button (for a particular attribute e.g. *Patient ID*) can be used (when the attribute value is entered next to it) to, respectively, automatically highlight (Select) any and all existing items in the browser holding a matching attribute value (i.e. any and all files to be assigned that value) and insert (Apply) a value to all highlighted items at once, thereby enabling the user to assign the same e.g. *Patient ID* to many images at once. This way, the user does not have to specify the content for the CSV files laboriously one file at a time.

Fig. 10.

Coronal views of average registered GM, WM, and CSF probability images and final labeled atlas for 46 subjects between 37 and 81 years of age.