

Mjolnir: Deformable Image Registration using Feature Diffusion

Lotta M. Ellingsen and Jerry L. Prince

Dept. of Electrical & Computer Engineering,
Johns Hopkins University, Baltimore, MD 21218, USA

ABSTRACT

Image registration is the process of aligning separate images into a common reference frame so that they can be compared visually or statistically. In order for this alignment to be accurate and correct it is important to identify the correct anatomical correspondences between different subjects. We propose a new approach for a feature-based, inter-subject deformable image registration method using a novel displacement field interpolation. Among the top deformable registration algorithms in the literature today is the work of Shen et al. called HAMMER. This is a feature-based, hierarchical registration algorithm, which introduces the novel idea of fusing feature and intensity matching. The algorithm presented in this paper is an implementation of that method, where significant improvements of some important aspects have been made. A new approach to the algorithm will be introduced as well as clarification of some key features of the work of Shen et al. which have not been elaborated in previous publications. The new algorithm, which is referred to as Mjolnir (Thor's hammer), was validated on both synthesized and real T1 weighted MR brain images. The results were compared with results generated by HAMMER and show significant improvements in accuracy with reduction in computation time.

Keywords: Deformable registration, Image features, Attribute vector, Magnetic resonance

1. INTRODUCTION

There are two main classes of image registration algorithms: feature-based registration¹ and intensity-based registration.^{2,3} The former registers images by aligning derived anatomical features while the latter uses dense subsets of image intensities. HAMMER¹ combines intensity-based and feature-based registration by introducing two key concepts: attribute vectors and driving voxels. Information about both the image intensity and derived image features are concatenated into an attribute vector, which is defined for every voxel in both the template image and the subject image. Comparison of attribute vectors is analogous to the matching of image intensities. HAMMER's driving voxels are hierarchically selected voxels that represent strong, reliable features that are to be aligned first. This action minimizes the number of sub-optimal correspondences in the beginning of the registration process.

The blending of both intensity-based and feature-based strategies puts HAMMER at the forefront of deformable registration algorithms today and is the main reason why we chose to implement and improve the method ourselves as a basis for future work. In this paper we describe the main steps of HAMMER, which are common to Mjolnir, and discuss the modifications and improvements that we have made. The main technical contribution of our work is a novel, fast interpolation of the displacement field in a deformable image registration process (section 2.4). Other contributions include clarification and improvements of some key features of the HAMMER algorithm, which are missing from the literature but are nonetheless important to the registration community.

Further author information:

J.L.P.: E-mail: prince@jhu.edu, Telephone: 1 410 516 5192, Fax: 1 410 516 5566

L.M.E.: E-mail: lotta@jhu.edu, Telephone: 1 410 516 6820, Fax: 1 410 516 5566

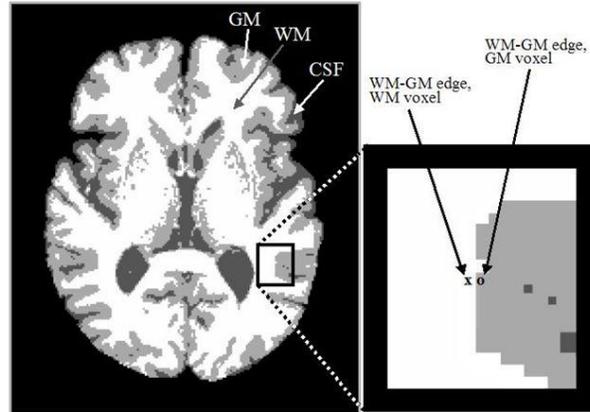


Figure 1. The edge type part of the attribute vector takes one of seven discrete values, one for a non-edge and six for different combinations of edges between WM, GM, and CSF. The left figure shows a hard segmentation of the image. The right figure (a zoomed part of the left figure) shows two voxels, \mathbf{x} and \mathbf{o} , with different edge-types on a WM-GM interface, depending on which side of the interface the voxel lies.

2. METHOD

2.1. Image Features

HAMMER concatenates information about the image intensity and derived local features into an attribute vector, which consists of three parts. The first part is a 1×1 vector that represents the edge type of the voxel in the image based on a hard segmentation of the image into gray matter (GM), white matter (WM) and cerebrospinal fluid (CSF). The edge type takes one of seven discrete values, one for non-edges and six for different combinations of edges between GM, WM, and CSF (see Fig. 1). The second part of the attribute vector is a 1×1 vector representing the intensity of the voxel, normalized between zero and one. The third and last part of the attribute vector is a 9×1 vector of geometric moment invariants (GMIs). Three GMIs are computed in a spherical neighborhood of radius R_G around each voxel for each of the three tissue types based on a hard segmentation of the image. These three GMIs are formulated from the zero-order and second-order 3D regular moments in the following way:

$$\begin{aligned} I_1 &= M_{0,0,0} \\ I_2 &= M_{2,0,0} + M_{0,2,0} + M_{0,0,2} \\ I_3 &= M_{2,0,0} \times M_{0,2,0} + M_{2,0,0} \times M_{0,0,2} + M_{0,2,0} \times M_{0,0,2} - M_{1,0,1}^2 - M_{1,1,0}^2 - M_{0,1,1}^2, \end{aligned}$$

where

$$M_{p,q,r} = \iiint_{(x_1)^2+(x_2)^2+(x_3)^2 < R_G^2} x_1^p x_2^q x_3^r f_t(x_1, x_2, x_3) dx_1 dx_2 dx_3$$

is the $(p+q+r)$ -th 3D regular moment and f_t is the membership function (either WM, GM, or CSF). The GMIs are also normalized between zero and one.

Mjolinir uses a similar attribute vector to that of HAMMER, but with three important differences. First, Mjolinir computes the GMIs using a soft or fuzzy segmentation of the image corresponding to the three main tissue types (see Fig. 2). Using a fuzzy segmentation instead of a hard segmentation reduces the potential for incorrectly labeled edges to be used in the registration process, which could lower its accuracy. The fuzzy segmentation used in Mjolinir was computed using Fuzzy And Noise Tolerant Adaptive Segmentation Method (FANTASM).^{4,5}

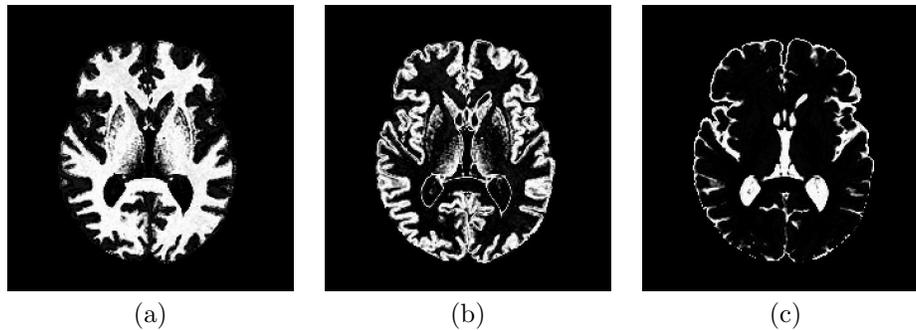


Figure 2. Fuzzy segmentation of the image into three membership functions corresponding to (a) the white matter, (b) gray matter, and (c) cerebrospinal fluid.

The second difference between Mjolnir’s and HAMMER’s attribute vectors concerns segmentation as well. The publicly available HAMMER program⁶ requires a preprocessing step in which the ventricles are identified and classified as a fourth class in the hard segmentation result. This step, which is not mentioned in their publication,¹ is time consuming, and Mjolnir does not require this extra step. It turns out that the GMIs — when computed from the fuzzy segmentation — are distinctive enough to identify the ventricles without requiring the extra segmentation step. The third and final difference between Mjolnir’s and HAMMER’s attribute vectors is that Mjolnir incorporates the actual MR intensities, as suggested in the aforementioned publication, instead of driving the registration entirely from the segmentation result as done in the HAMMER program. In order to make the intensities between images more comparable, Mjolnir applies histogram equalization to each image and normalizes the values between zero and one.

2.2. Driving Voxels

When an attribute vector has been computed for every voxel in both images, distinctive driving voxels (or landmark points) are automatically identified. The driving voxels are selected so that voxels with strong, reliable features are related between the images first. Thus, the initial set of driving voxels contains only points of high curvature such as sulcal fundi and gyral crowns. Such points can be automatically detected using the first GMI, which (as described above) is the zero-th order moment computed in a spherical neighborhood around each voxel. Thus, each voxel in the I_1 -image represents the average intensity of the voxels in a spherical sub-volume around it. If we consider the WM membership function we see that there is a high percentage of white matter tissue around an edge point on the sulcal fundus while there is a low percentage of white matter tissue around an edge point on the gyral crown (see Fig. 3). This information is used both in HAMMER and Mjolnir to select a small initial set of highly distinctive driving voxels. In later iterations of the registration algorithm, the set is augmented by expanding a sphere around each primary driving voxel so that every voxel in the neighborhood is added to the driving voxel set. In the last few iterations, every voxel in the template image is considered as a driving voxel. The number of driving voxels in the subject image, however, is kept constant throughout the process in order to save computation time.

2.3. Correspondences

Given sets of driving voxels in the subject and template images, it is necessary to find correspondences between the driving voxels, thereby yielding landmark correspondences from which to drive the registration process. Derived correspondences for each template driving voxel are found both in the direction from subject to template and vice versa. Only one displacement field is computed, however (in the domain of the template). For each driving voxel in the subject, \mathbf{x}_s^d , a search is performed to locate a matching driving voxel in the template, \mathbf{x}_t^d . A match is defined when a similarity value s above the threshold t_{vox} is satisfied, where similarity is defined by

$$s(\mathbf{x}_s, \mathbf{x}_t) = \begin{cases} 0, & \text{if } \mathbf{a}_1(\mathbf{x}_s) \neq \mathbf{a}_1(\mathbf{x}_t) \\ c([\mathbf{a}_2(\mathbf{x}_s)\mathbf{a}_3(\mathbf{x}_s)], [\mathbf{a}_2(\mathbf{x}_t)\mathbf{a}_3(\mathbf{x}_t)]), & \text{otherwise} \end{cases} \quad (1)$$

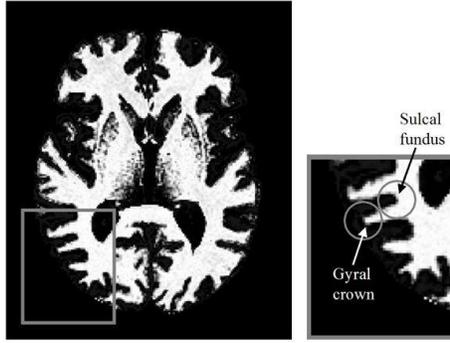


Figure 3. The left figure shows the WM membership function. The right figure demonstrates how a high percentage of white matter tissue surrounds an edge point on the sulcal fundi while a low percentage of white matter surrounds an edge point on the gyral crowns.

where

$$c([\mathbf{a}_2(\mathbf{x}_s)\mathbf{a}_3(\mathbf{x}_s)], [\mathbf{a}_2(\mathbf{x}_t)\mathbf{a}_3(\mathbf{x}_t)]) = (1 - |\mathbf{a}_2(\mathbf{x}_s) - \mathbf{a}_2(\mathbf{x}_t)|) \cdot \prod_{i=1}^9 (1 - |\mathbf{a}_3^i(\mathbf{x}_s) - \mathbf{a}_3^i(\mathbf{x}_t)|)$$

and \mathbf{a}_1 , \mathbf{a}_2 , and \mathbf{a}_3 are the three parts of an attribute vector. If a matching subject voxel is found, the displacement vector \mathbf{u}_s is attached to the template voxel \mathbf{x}_t^d (i.e., in the direction from template to subject). The search region is a spherical neighborhood of radius R_s around the subject voxel. The radius R_s is set to be large in the early iterations, when the number of driving voxels is small, and gradually decreases to a single voxel in the last few iterations. Similarly, the threshold t_{vox} is large in the beginning and gradually decreases as the search region shrinks.

Then for each driving voxel in the template, \mathbf{x}_t^d , a search is performed for candidate voxels in the subject with a similarity value above t_{vox} . The search region is again spherical with decreasing radius as the program progresses. For each candidate voxel found in the subject, the sub-volume around the voxel is tentatively deformed to the template voxel. The deformed sub-volume is then compared voxel-by-voxel with the corresponding sub-volume in the template. The average similarity value of every voxel within the sub-volume is computed and the candidate subject voxel \mathbf{x}_s that gives the highest average similarity value above t_{vol} is used to form the displacement vector $\mathbf{u}_t = \mathbf{x}_s - \mathbf{x}_t^d$ at the coordinate of \mathbf{x}_t^d . If, in addition, the template driving voxel \mathbf{x}_t^d has a previously attached displacement vector \mathbf{u}_s , the two vectors are combined as follows

$$\mathbf{u}_c(\mathbf{x}_t^d) = w_1 \mathbf{u}_t(\mathbf{x}_t^d) + w_2 \mathbf{u}_s(\mathbf{x}_t^d),$$

where w_1 and w_2 are weighting scalars.

HAMMER uses similar criteria for finding the best matched subject voxel for every template driving voxel. However, there is one important difference. HAMMER does not compare every voxel pair within the spherical neighborhood, but instead subsamples the volume in order to save computation time. This decreases the achievable registration resolution, as demonstrated in Fig. 4. If only every other voxel pair in the sub-volume is used in the comparison, the algorithm does not capture high resolution details in the image.

2.4. Displacement field interpolation

The displacement vectors derived from the driving voxels must be interpolated throughout the entire image in order to generate a complete displacement field. HAMMER uses non-overlapping Gaussian kernels applied to discrete sub-volumes around each driving voxel in order to interpolate the displacement field around that voxel. Thus, the displacement of each template voxel is influenced by a single driving voxel. At the end of each iteration, the displacement field is smoothed by both global and local affine transformations. HAMMER then

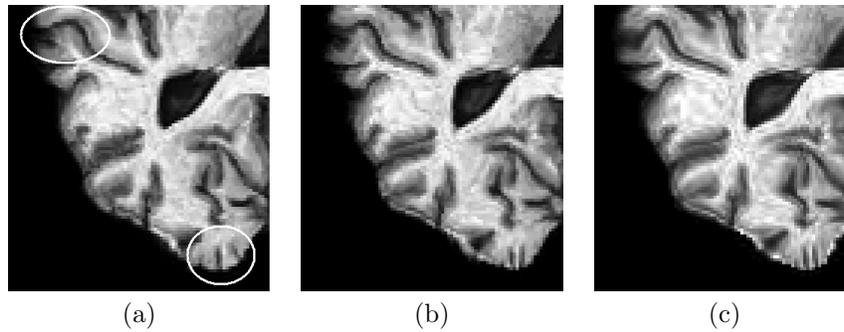


Figure 4. (a) Deformed subject image using every other voxel in the sub-volume comparison. The regions where high resolution details are not captured are circled. (b) Deformed subject image using every voxel in the sub-volume comparison. (c) Corresponding template image.

uses the Laplacian operator both locally and globally to smooth any discontinuities that will be introduced on the boundary of each of the deformed sub-volumes. This combination of many local and global deformations may cause irregular displacements in the registration result.

Mjolnir introduces a novel way to interpolate the displacement field by solving a single partial differential equation (PDE). A fast multi-grid method⁷ is used to find the equilibrium solution of the following equation

$$g\nabla^2\mathbf{v} - p(\mathbf{v} - \mathbf{u}) = \mathbf{0}, \quad (2)$$

Here $\nabla^2 = \partial^2/\partial x^2 + \partial^2/\partial y^2 + \partial^2/\partial z^2$ is the Laplacian operator, \mathbf{v} is the interpolated displacement field, and \mathbf{u} is the initial displacement field computed from the correspondences at the driving voxels. The scalar functions p and g are weighting functions that control the closeness of the displacement field \mathbf{v} to the correspondence field \mathbf{u} and the smoothness of \mathbf{v} , respectively. The scalar function g was chosen to be gradually decreasing, starting with a larger value, while the correspondence field is sparse. This allows the influence of each driving voxel to propagate smoothly to a larger neighborhood in the beginning but become more intense with a smaller influence zone as the density of driving voxels increases. The scalar function p represents the similarity value between the corresponding voxels. A high similarity indicates a more reliable feature match and therefore, the displacement field should follow that correspondence closely. In this way, a smooth displacement field is provided where there are few strong features (e.g., sub-cortical white matter regions) while enabling progress into small concavities where image features are more complex (e.g., on the highly convoluted cortex).

2.5. Multi-resolution scheme

Mjolnir uses three different resolutions. First, the images are downsampled by a factor four and Mjolnir runs until a stop criterion is satisfied — e.g., the displacement field is not changing. Then the resulting displacement field is upsampled and interpolated using trilinear interpolation and the deformed image applied as an input into the next resolution step. This procedure is repeated twice until the final input image is the same size as the original image.

3. RESULTS

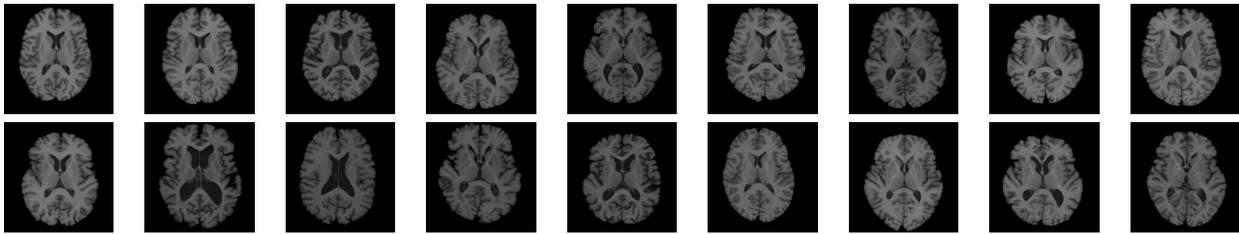
Mjolnir was validated both on synthesized and real T1 weighted MR images of the brain. Table 1 shows the parameters used in the following experiments.

3.1. Average Brain

A total of 18 brain images with significant variability were selected and registered to a 19-th template image using Mjolnir (see Fig. 5). The average of the 18 deformed images was computed and compared with results from the same experiment performed using HAMMER. The accuracy of Mjolnir is reflected in the sharpness of

Table 1. Parameters Used in the Experiments.

Parameters	Low Resolution	Middle Resolution	High Resolution
R_G : Radius of spherical neighborhood in GMI computations	3	3	7
R_S : Starting radius of spherical search neighborhood	7	6	5
t_{vox} : Starting similarity threshold for voxel similarity	0.8	0.8	0.8
t_{vol} : Starting similarity threshold for average volume similarity	0.6	0.6	0.6
$w1$: Weight in displacement field	0.7	0.7	0.7
$w2$: Weight in displacement field	0.3	0.3	0.3
g : Weight in PDE	0.63 to 0.3	0.63 to 0.3	0.63 to 0.3

**Figure 5.** A total of 18 different subjects selected for validation study.

the average image (see Fig. 6). Mjolnir's results are visually compatible with results generated by HAMMER; however, some evidence of strong artificial edges are visible around the thalamus in the HAMMER result (see Fig. 7). In addition, the shape of the putamen is slightly different and the size is smaller in the HAMMER result than in the template image (see Fig. 7).

3.2. Validation

A brain image was deformed to create an artificial brain image with a known displacement field (see Fig. 8). The two registration programs, Mjolnir and HAMMER, were used to register the initial brain to the artificial brain and their displacement field results compared with the known displacement field. The results showed significant improvements in Mjolnir in terms of maximum absolute error and slight improvements in terms of average absolute error (see Fig. 9). The two programs were run on the same machine (Pentium Xeon 3.0 GHz, 4GB Ram), including all required preprocessing steps. Mjolnir showed improvements in terms of speed, finishing in 170 minutes while HAMMER finished in 202 minutes (see Fig. 9). Both images were $256 \times 256 \times 198$ voxels in size.

4. CONCLUSIONS

One of the top deformable registration algorithms (HAMMER) was implemented and improved. The new algorithm (Mjolnir) was validated on both synthesized and real T1 weighted MR brain images. The maximum error in Mjolnir's registration of a synthesized brain to a real brain was 3.8407 voxels compared to 5.3766 voxels in HAMMER. The average absolute error was 0.4949 voxels in Mjolnir versus 0.5832 voxels in HAMMER. Only 1.06% of the voxels had absolute error above 2 voxels in Mjolnir compared to 2.15% in HAMMER. Mjolnir is slightly faster than HAMMER, finishing in 170 minutes while HAMMER finished in 202 minutes, including all required preprocessing steps.

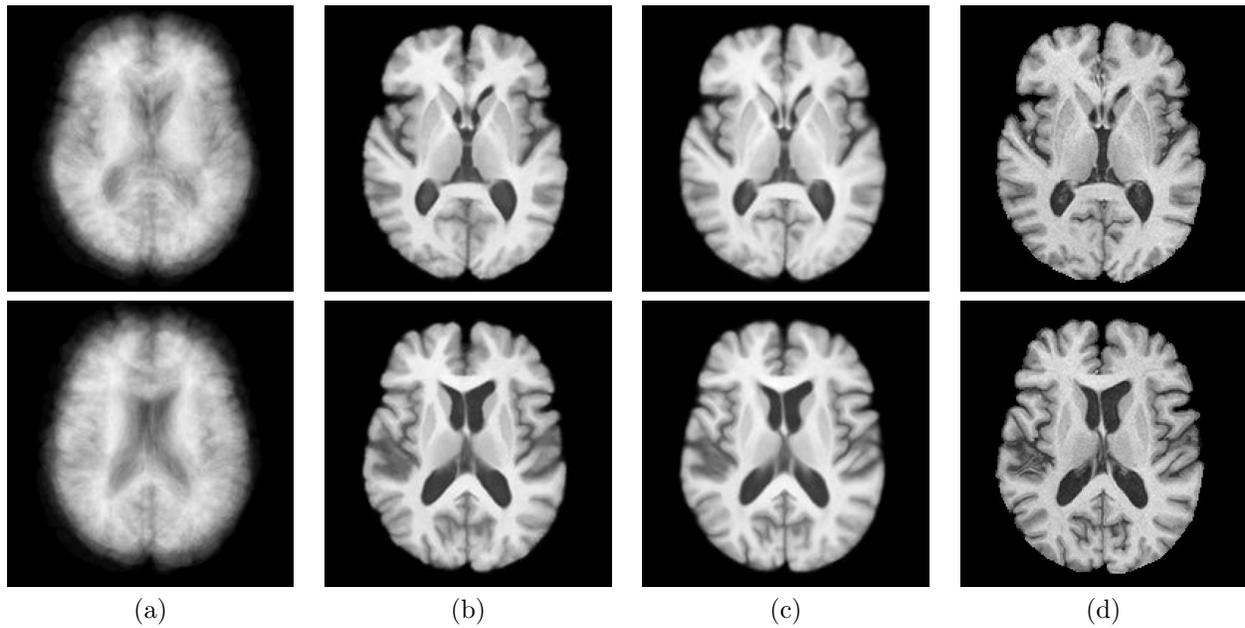


Figure 6. Average of 18 brains registered to the 19th brain (a) before registration (b) after Mjolnir registration (c) after HAMMER registration. (d) Template image. Two different slices are shown.

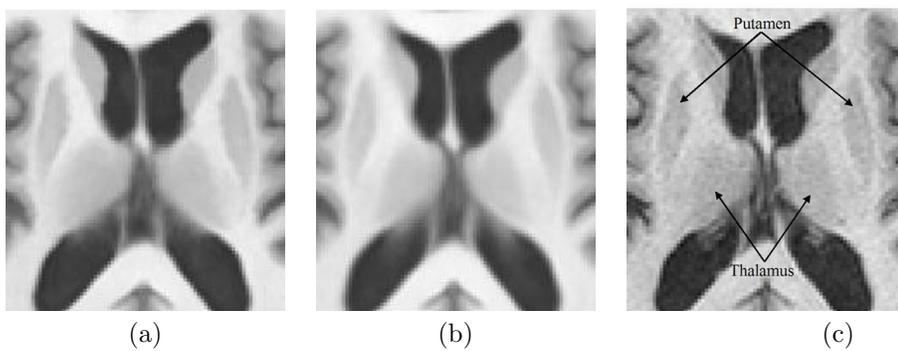


Figure 7. Average of 18 brains registered to the 19th brain (a) after Mjolnir registration (b) after HAMMER registration. (c) Template image. A sharp, artificial edge that is not evident in the template image has been introduced around the thalamus in the HAMMER result. The shape and size of the putamen is also different than the template.

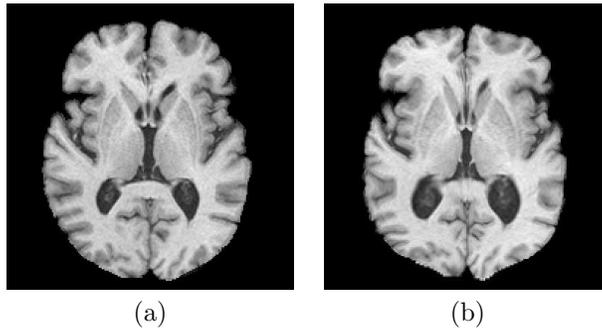


Figure 8. (a) Real MR brain image. (b) Artificial brain image created from image (a) by using a known displacement field.

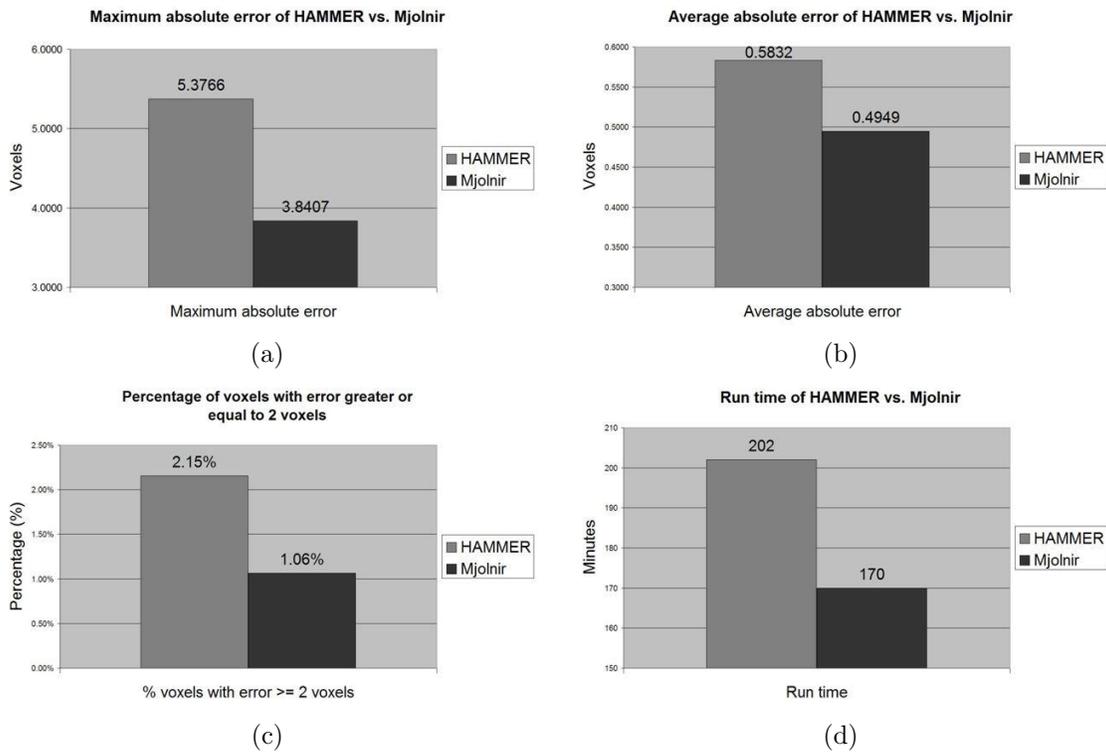


Figure 9. (a) Maximum error of HAMMER vs. Mjolnir. (b) Average absolute error of HAMMER vs. Mjolnir. (c) Percentage of voxels with error greater or equal to 2 voxels. (d) Runtime of HAMMER vs. Mjolnir on a Pentium Xeon 3.0 GHz, 4GB Ram.

5. ACKNOWLEDGMENTS

This work was supported in part by NIH/NINDS Grant R01NS37747.

REFERENCES

1. D. Shen and C. Davatzikos, "HAMMER: Hierarchical attribute matching mechanism for elastic registration," *IEEE Trans. Med. Imag.* **21(11)**, pp. 1421–1439, 2002.
2. G. K. Rohde, A. Aldroubi, and B. M. Dawant, "The adaptive bases algorithm for intensity-based nonrigid image registration," *IEEE Trans. Med. Imag.* **22(11)**, pp. 1470–1479, 2003.
3. D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes, "Nonrigid registration using free-form deformations: Application to breast MR images," *IEEE Trans. Med. Imag.* **18(8)**, pp. 712–721, 1999.
4. D. Pham, "Robust fuzzy segmentation of magnetic resonance images," in *Proceedings of the Fourteenth IEEE Symposium on Computer-Based Medical Systems (CBMS2001)*, pp. 127–131, IEEE Press, 2001.
5. D. Tosun, M. E. Rettmann, X. Han, X. Tao, C. Xu, S. M. Resnick, D. L. Pham, and J. L. Prince, "Cortical surface segmentation and mapping," *NeuroImage Special Issue on IPAM Mathematics in Brain Imaging* **23(S1)**, pp. S108–S118, 2004.
6. HAMMER software: <http://www.rad.upenn.edu/sbia/rsoftware.html>.
7. W. Briggs, van E. Henson, and S. McCormick, *A Multigrid Tutorial, 2nd Edition*, SIAM, Philadelphia, PA, 2000.