Grid Based Registration of Diffusion Tensor Images Using Least Square Support Vector Machines

Esmaeil Davoodi-Bojd[1], Hamid Soltanian-Zadeh[2,3]

[1] Control and Intelligent Processing Center of Excellence, School of Electrical and Computer Engineering, University of Tehran, Tehran 14395-515,Iran
es.davoodi@ece.ut.ac.ir

[2] Control and Intelligent Processing Center of Excellence, School of Electrical and Computer Engineering, University of Tehran, Tehran 14395-515,Iran
hszadeh@ut.ac.ir

[3] Image Analysis Laboratory, Radiology Department, Henry Ford Health System, Detroit, MI 48202, USA
hamids@rad.hfh.edu

Abstract

In this paper, we present a non-rigid image registration method for DTMR images. This method consists of finding control points using a piecewise affine registration and then estimating final transform between two images by minimizing corresponding Least Squares Support Vector Machine (LS-SVM) function of these control points. In our shceme, a fully symmetrically grid points in the reference image is selected and the transformed grid points is computed using the results of piecewise affine registration. These control points are then employed to estimate final transform between images using minimizing related LS-SVM function. In both piecewise affine registration and final transform function estimation, a finite strain (FS) based reorientation strategy is applied to adopt these methods for DTMR images. The main advantage of this method is that in estimating transform function using control points, it considers all control points and thus all transforms of sub-images. Therefore, each point in the reference image is transformed consistently with the whole image points.

Keywords

Diffusion Tensor MRI, Image Registration, Piecewise Affine Image Transform, Diffusion Tensor Reorientation, Least Square Support Vector Machines

1. Introduction

Diffusion Tensor Magnetic Resonance Imaging (DTMRI) is a noninvasive tool for determining white matter connectivity in the brain. DTMRI adds to conventional MRI the capability of measuring the random motion of water molecules, referred to as diffusion. It has been known that the water molecule motion is restricted in the axons due to existence of myelin sheath [1,2]. Therefore, the most important distinctive characteristic of DTMR images is that they have directional information of microtubule living structures. Direction at each voxel is computed mathematically using a 3×3 symmetric positive semi-definite matrix $\boldsymbol{D}$, known as Diffusion Tensor (DT), provided by DTMRI. Consequently, working with and processing of DTMR images are more complicated than other conventional image modalities.

On the other hand, image registration is a useful tool in group normalization, atlas construction, and automatic multi structure segmentation. However, applying conventional image registration methods to the DTMR images can not get acceptable results without considering orientation information of these images. Therefore, it has been proposed to use tensor reorientation strategies when applying spatial transformation to a DTMR image [3].

In [3] two main reorientation methods have been proposed: Finite Strain (FS) method and Preservation of Principal Direction (PPD). FS computes a rotation matrix from the transform function and then applies it to the tensors such that direction of eigenvectors of each tensor rotates consistently with the original spatial transform. On the other hand, PPD considers more information about transform and it is useful for large deformated DT images. Therefore, for common image registration and normalization the FS reorientation strategy is usually sufficient.

In [4] different combinations of channels for image registration were used, including four channels of scalar characteristics of diffusion tensor and one channel of components of diffusion tensor. Although they did not use reorientation strategy in the last channel, but they concluded that using whole diffusion tensor components in image registration get better results than other combinations.

Several non-rigid DTMR image registration procedures have been proposed, recently. Some of them use affine registration as a basis and then by applying it in different parts of the images find a non-linear transform to match the images [5,6]. On the other hand, some methods use multi resolution schemes by increasing the complexity of transform function in each level of registration [7].

One of the important issues of piecewise affine registration is the problem of combining resulting transforms which are belong to different parts of images. In [5] an interpolation method between neighbored transforms was proposed. Unfortunately, this solution only considers limited number of neighbors to estimate transformed border voxels. Thus the resulting transformed regions may be inconsistent with the whole transformed image.

On the other hand, in [8] a scheme for image registration using least square support vector machines (LS-SVM) was presented. The important capability of this method is that it estimates a non-linear transform between two images using some control points from the images. However, this method has been not used in DTMR image field yet.

The aim of is this work is to develop a non-rigid image registration method for DTMR images using piecewise affine transform and least square support vector machines. In our scheme we first compute some control points from two images. These control points are used to estimate a transform function between two images by minimizing their LS-SVM function, in combination with reorienting the diffusion tensors during the transform.

2. Methods

In order to find some reliable control points between to images we apply a piecewise affine registration. For this purpose, these two images are divided into $n_x \times n_y \times n_z$ sub-images. Then an affine transform is computed using a registration algorithm for each sub-image. Finally, a grid of points is selected in the reference image and the corresponding transformed points are computed using the resulting sub-transforms.

2.1. Piecewise affine registration

An affine transform, $A_p(.)$, can be expressed by 12 parameters: 3 parameters for rotation ($\underline{q}$), 6 for deformation ($\underline{s}$), and 3 for translation ($\underline{t}$). Transformed point, $\underline{y}$, of a point, $\underline{x}$, can be computed by the following equation:

$$\underline{y} = (Q.S).\underline{x} + T = A_{\underline{p}}(\underline{x}) \qquad (1)$$

where $\boldsymbol{Q}$ is the 3×3 rotation matrix with 3 independent parameters, $\boldsymbol{S}$ is the 3×3 deformation matrix with 3 independent parameters, T is the 3×1 translation vector, and finally, $\underline{p}$ is the whole unknown parameter vector consists of [$\underline{q}$, $\underline{s}$, $\underline{t}$].

In fact, image registration is a minimization problem of a dissimilarity criterion between two images. However, for DTMR images this criterion must handle the orientation of the tensors. [5] has used a dissimilarity function with FS reorientation strategy for affine registration, as shown in (2).

Algorithm1

```
Begin
  • Divide given two DT images, sensed image and reference
    image, into $n_x \times n_y \times n_z$ equal size sub-images.
  • For each corresponding pair of sub-images ($\Omega^s_{ijk}$ and
    $\Omega^r_{ijk}$, $i=1,\dots, n_x$, $j=1,\dots, n_y$, $k=1,\dots, n_z$) do the following
    steps:
      1. Twice each dimension of each sub-image by
         adding zero voxels to their peripheral sides. (This
         is essential in order to let transformed image
         rotate, translate, or scale beyond sub-image size).
      2. Apply evolutionary algorithm with cost function
         given in (3) to find an initial estimation $\underline{p}^0_{ijk}$ of
         unknown parameters.
      3. Apply CG algorithm using initial point, $\underline{p}^0_{ijk}$, to
         minimize dissimilarity function given in (2) and
         find affine transform parameters $\underline{p}_{ijk}$.
end
```

Fig. 1. Piecewise affine registration algorithm of DT images

$$\phi(\underline{p}) = \int_{\Omega} \left\| I_s((Q.S).\underline{x} + T) - Q.I_r(\underline{x}).Q^t \right\|^2 d\underline{x} \qquad (2)$$

where $\boldsymbol{I_s}(.)$ and $\boldsymbol{I_r}(.)$ are sensed and reference images, respectively, and Ω is the region in which the dissimilarity measure is computed. The second term of integral, $Q.\boldsymbol{I_r}(\underline{x}).Q^t$, reorients the diffusion tensor in each voxel.

This function can be minimized by gradient based optimization methods because its gradient expressions can be computed straightly. In [5] the gradient expressions of this function have been computed and we use them in a Conjugate Gradient (CG) algorithm to find unknown affine transform parameters.

One of the basic problems of gradient based optimization methods is that they are easily trapped in local minimums if their object function is too complex. Therefore, we use an Evolutionary Algorithm (EA) to search for optimum parameters by minimizing function $\phi(.)$ before applying CG algorithm.

In EA, the unknown parameters ($\underline{p}$) can be assigned with any values. This may lead the algorithm to too far solutions if the transformed point, i.e. $\underline{y}$, lays out of range of the images (for example $\underline{y}$=[-10,-20,3]^t). Therefore, the cost function for EA should consider these out-range-points. Our proposed cost function for EA is computed by (3).

$$\psi(\underline{p}) = \phi(\underline{p}) + \frac{\sum_{\underline{x}\in\Omega} isOut(A_{\underline{p}}(\underline{x}))}{N_{\Omega}} \qquad (3)$$

where N_{Ω} is the total number of points in Ω, and the function $isOut(.)$ is defined as:

$$\begin{cases} isOut(\underline{y}) = 1, & \underline{y} \notin \Omega \\ isOut(\underline{y}) = 0, & \underline{y} \in \Omega \end{cases} \qquad (4)$$

In summery, (3) adds the percentage of out-range transformed points to the previous cost function. This guarantees that inconsistent solutions are eliminated during the iterations of EA. The summary of our piecewise affine registration algorithm is given in fig. 1.

The remaining problem is that how to combine the resulting sub-transforms to build final transformed image. The problem arises mainly in the border voxels of sub-images. [5] solves this problem by finding new sub-transforms for those voxels using interpolation between neighboring sub-transforms. Unfortunately, this solution only considers limited number of neighbors to estimate transformed border voxels.

We handle this problem by considering a grid of points in the reference image, for example 8 corner points of a cubic with half size in each dimension, centered on each sub-image (we will discuss about the strategy of selecting these control points on chapter 3.3), and calculate their transformed points using their corresponding sub-transforms. Then, we employ these points as control points to estimate transformed image using LS-SVM.

2.2. Transformation estimation via LS-SVM

Suppose we have two corresponding point sets ***S*** and ***R*** belong to sensed image and reference image, respectively, and each set consists of *N* spatial control points. The aim of transformation estimation is to find a transform which map the sensed image control points, i.e. ***S***, to the corresponding reference image control points, i.e. ***R***. We use a model with linear combination of RBF functions as proposed in [8] and shown in (1).

$$\underline{y}=f(\underline{x})=\sum_{i=1}^{N}\underline{a}_i.\exp\left\{-\left\|\underline{x}-\underline{x}_i\right\|^2/\sigma^2\right\}+\underline{b} \qquad (5)$$

Where $\underline{x}$ is the coordinate of a spatial point ($\underline{x}=[x_1, x_2, x_3]^t$) in the sensed image and $\underline{y}$ is the estimated corresponding point in the reference image, and $\{\underline{x}_i, i=1,\ldots,N\}$ are *N* control points in the sensed image. The coefficients $\underline{a}_i$ and $\underline{b}$ are computed using the following equations which are the solutions of minimizing this LS-SVM problem [8].

$$\underline{a}=(Y-\underline{b}.1).\Omega^{-1} \qquad (6)$$

$$\underline{b}=\frac{Y.\Omega^{-1}.1^T}{1.\Omega^{-1}.1^T} \qquad (7)$$

where ***Y*** is a 3×*N* matrix representing *N* control points in reference image, **1** is a 1×*N* vector consists of *N* ones, and **Ω** is a *N*×*N* matrix whose elements $\mathbf{\Omega}_{ij}$ is computed using the following equation.

$$\begin{cases}\Omega_{ij}=\exp\{-\left\|x_i-x_j\right\|^2/\sigma^2\}, & i\neq j\\ \Omega_{ij}=\exp\{-\left\|x_i-x_j\right\|^2/\sigma^2\}+\gamma^{-1}, & i=j\end{cases} \qquad (8)$$

The parameters σ and γ are tuning parameters. While σ controls the depth of contribution of each control points in its neighborhood, γ determines a tradeoff between the model complexity and training error [8]. The small γ leads to better generalization while the large γ leads to smaller approximation error of control points. In [8] it has been suggested an adaptive scheme to determine these parameters. But generally they can be chosen by prior knowledge or empirically.

2.3. DT reorientation for estimated function

Equation (1) only estimates the transformation function between two images. However, for DTMR images a reorientation strategy must be applied. FS reorientation strategy needs a rotation matrix *Q* for each tensor. For affine transforms this matrix can be computed easily, but for this nonlinear estimated function more calculations must be performed.

Suppose for each voxel the nonlinear function in (5) can be expressed by an affine transform, i.e.

$$\underline{y}=f(\underline{x})\equiv F_{\underline{x}}.\underline{x}+T \qquad (9)$$

where $F_{\underline{x}}=Q_{\underline{x}}.\ S_{\underline{x}}$ is the linear transformation matrix. Differentiating both sides of (9) subject to $\underline{x}$ gives,

$$J_f(\underline{x})=F_{\underline{x}} \qquad (10)$$

where $J_f(\underline{x})$ is the Jacobian matrix of function *f(.)* and can be computed using the following equation:

$$J_f(\underline{x})= -\frac{2}{\sigma^2}\times\sum_{i=1}^{N}\underline{a}_i.(\underline{x}-\underline{x}_i).\exp\left\{-\left\|\underline{x}-\underline{x}_i\right\|^2/\sigma^2\right\} \qquad (11)$$

Now the rotation matrix can be estimated from computed $F_{\underline{x}}$ and the following relation proposed by [3].

$$Q_{\underline{x}}=(F_{\underline{x}}.F_{\underline{x}}^t)^{-0.5}.F_{\underline{x}} \qquad (12)$$

And finally, the transformed image $I_t(.)$ of the reference image $I_r(.)$ can be expressed as,

$$I_t(f(\underline{x}))=Q_{\underline{x}}.I_r(\underline{x}).Q_{\underline{x}}^t \qquad (13)$$

which consists of two transforms: spatial transform and tensor reorientation transform.

3. Results and Discussion

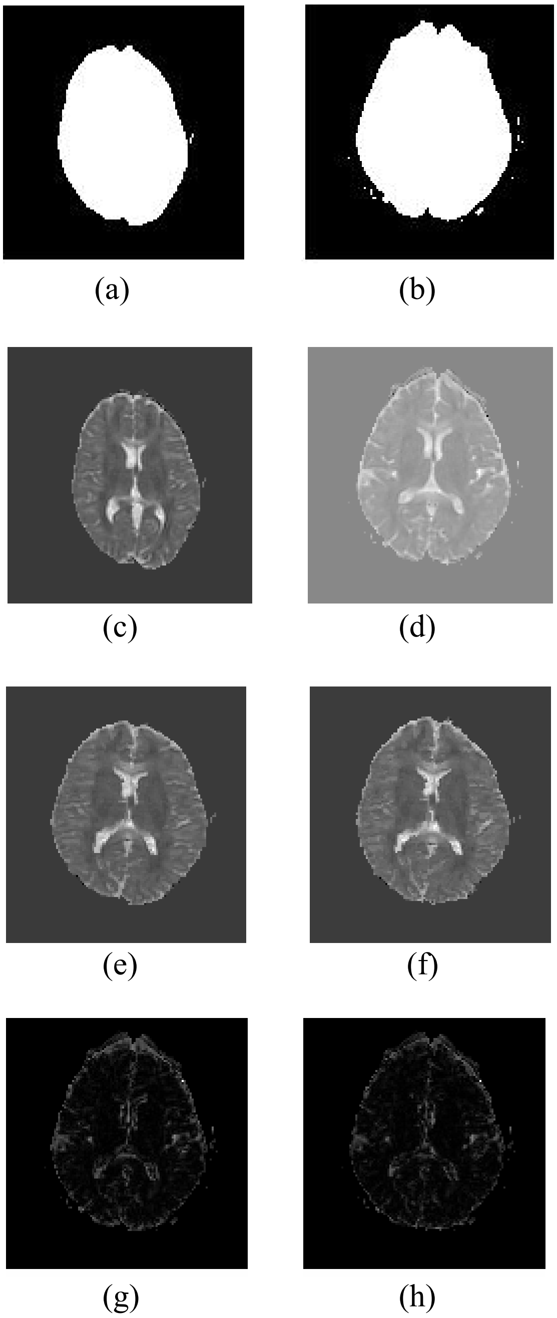
3.1. Data

In this work we used two DTMR image data-set acquired in Henry Ford Hospital using a 1.5-T GE SIGNA EXCUTE system from two healthy volunteers. The resolution of each voxel in these data-sets was 0.9375 × 0.9375 × 3 mm and the image size was 256 × 256 × 40 voxels. For each slice 6 gradients in 6 directions was applied to get 6 diffusion-weighted images and one T2-weighted image was acquired for each slice.

In order to reduce the computation time, we downsampled each image slice to 128 ×128 voxels. This does not affect the generality of the problem.

3.2. Pre-processing

For each slice, we extracted brain using their corresponding 6 diffusion weighted-images and used this



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Fig. 2. (a) and (b), computed masks for eliminating background noise correspond to reference image and subject image, respectively. (c) and (d), correspond to first component of diffusion tensor of reference image and subject image, respectively. (e) is the result of affine transform of reference image, computed by the affine registration. (f) is the final result of estimating transform function from control points computed using piecewise affine registration. And finally, (g) and (h) are the absolute differences between subject image and images (e) and (f), respectively. All of these images correspond to slice 20 of image data set.

as a mask to cancel background noise. To building this mask we performed fallowing steps:

- The image mask for each slice was calculated by averaging corresponding voxels of the 6 diffusion-weighted images.
- Assuming a threshold, voxels greater than this threshold were assigned to 1 and other voxels were assigned to 0. We selected this threshold equal to 250 for this data set (The range of voxel values were between 0 and 15035). Two samples of this mask are illustrated in fig. 2 (a, b) .

In the next step we calculated diffusion tensors for each voxel using a method suggested by [9]. This method is fast because it gives an analytical solution for computing tensors for whole image, rather than for every data point. In fig.2 (c, d) the first diffusion tensor component for reference and subject images are illustrated.

After calculating diffusion tensors, the resulting images were aligned together using an affine registration algorithm, before matching them by our piecewise affine algorithm. This gives us an initial affine transform and thus reduces complexity of the next step of matching process. The algorithm we used was the algorithm in fig. 1 with $n_x=n_y=n_z=1$. The result of pre-matching of two images is illustrated in fig. 2 (e).

3.3. Main registration procedure

The main registration process consists of finding control points using piecewise affine registration and then estimating final transform between two images by minimizing LS-SVM functin of these control points.

In the piecewise affine registration, i.e. algorithm 1, we chose $n_x=n_y=16$ *and* $n_z=5$. Therefore, by extracting 8 control points from each sub-image, 16×16×5×8=10240 pairs of control points were computed. It may appear that by increasing the number of sub-images and consequently by increasing the number of control points, the resulting transform function is more precise, but the size of sub-image should be large enough so that its voxels have enough needed information for registration process. For example registration of 2×2×2 sub-images can not lead to good results.

Another important issue about control points is that how some points from each sub-image are selected as control points. At first glance, it can be said that this can be done without any restriction about the number or the position of these points. However, since we have used piecewise

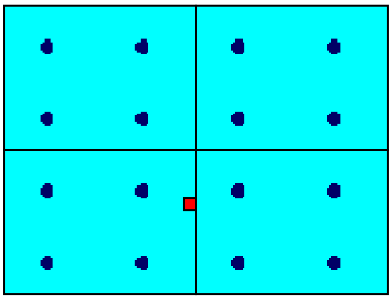


Fig. 3. Illustration of a fully symmetrically grid points. Blue points are the control points and the red point is a border point which we want to estimate its transformed point from the control points.

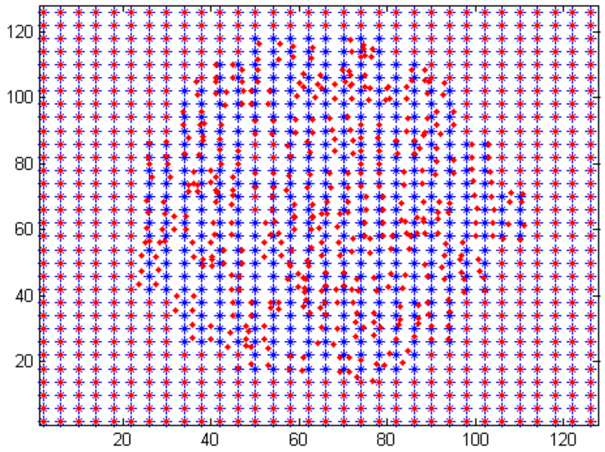


Fig. 4. A typical grid (blue stars), correspond to slice 20 of reference image, and its transformed grid (red points), computed from the result of the piecewise affine registration.

affine registration to match the images, two key points should be considered:

- Since for each pair of sub-images an affine transform is computed (in the registration process), at least four non-planar spatial points should be selected from each sub-image to have the complete effect of this affine transform.
- The control points should be located fully symmetrically in the reference sub-image space (see fig. 3). This consideration is the result of the fact that in the LS-SVM method, in order to estimate transformed point of a point in the reference image, the distances between that point and all control points is computed (see (5)). Therefore, the points on borders (for example the red point in fig. 3) should see equal distances from symmetric control points on the neighbored sub-images so that they see equal effects of neighbored transforms.

Consequently, we selected 8 corner points of a cubic with half size in each dimension, centered on each sub-image to build our grid in the reference image. In fig. 4, a typical grid and its transformed grid are shown.

After calculating control point grids, we estimated final transformed image, which is shown in fig.2 (f), using the method introduced in chapter 2.2. In this figure the absolute difference between subject image and final transformed image is also shown. It can be seen that this image is very similar to the subject image and there is not any inconsistent region thanks to the capability of LS-SVM method to estimate a smooth transform function.

4. Conclusion

In this paper, we presented a non-rigid image registration method for DTMR images. This method consists of finding control points using a piecewise affine registration and then estimating final transform between two images by minimizing LS-SVM function of these control points.

In piecewise affine registration the images are divided into equal size sub-images. Then, an evolutionary algorithm is applied to find an initial solution for affine transform parameters in each pair of sub-images. After that, a CG algorithm is used to refine this solution.

In the final step, a fully symmetrically grid points in the reference image is selected and the transformed grid points is computed using the results of piecewise affine registration. These control points are then employed to estimate final transform between images using minimizing related LS-SVM function. In both piecewise affine registration and final transform function estimation, an FS based reorientation strategy is applied to adopt these methods for DTMR images.

One of the advantages of this method is that in estimating transform function using control points, it considers all control points and thus all transforms of sub-images. Therefore, each point in the reference image is transformed consistently with the whole image points.

Another advantage of this procedure is that it can be applied with different sub-image sizes and consequently with different grid sizes. This can help us to estimate a fine transform function between images from coarse to fine resolutions.

References

[1] D. Goldberg-Zimring, A.U. Mewes, M. Maddah, and S.K. Warfield, “Diffusion tensor magnetic resonance imaging in multiple sclerosis,” American Society of Neuroimaging, vol. 15, pp. 68S - 81S, Oct. 2005.

[2] C.F. Westin, S.E. Maier, H. Mamata, A. Nabavi, F.A. Jolesz, and R. Kikinis, “Processing and visualization for diffusion tensor MRI,” Med. Imag. Analysis, vol. 6, pp. 93-108, 2002.

[3] D.C. Alexander, C. Pierpaoli, P.J. Basser, and J.C. Gee, “Spatial transformations of diffusion tensor magnetic resonance images,” IEEE Trans. Med. Imag., vol. 20, no. 11, pp. 1131-1139 , Nov. 2001.

[4] H.J. Park, M. Kubicki, M.E. Shenton, A. Guimond, R.W. McCarley, S.E. Maier, R. Kikinis, F.A. Jolesz, and C.F. Westin, “Spatial normalization of diffusion tensor MRI using multiple channels,” Neuroimage, vol. 20, no. 4, pp. 1995-2009, 2003.

[5] H. Zhang, P.A. Yushkevich, D.C. Alexander, and J.C. Gee, “Deformable registration of diffusion tensor MR images with explicit orientation optimization,” Med. Imag. Analysis, vol. 10, pp. 764-785, 2006.

[6] H. Zhang, P.A. Yushkevich, and J.C. Gee, “Registration of diffusion tensor images,” Proceedings of the 2004 Com. Soc. Con. on Com. Vis. And Pat. Rec, vol. 1, pp. 842-847, 2004.

[7] Y. Cao, M.I. Miller, S. Mori, R.L. Winslow, and L. Younes, “Diffeomorphic matching of diffusion tensor images,” Conference on Computer Vision and Pattern Recognition. Workshop, pp. 67-74, 2006.

[8] D.Q. Peng, J. Liu, J.W. Tian, and S. Zheng, “Transformation model estimation of image registration via least square support vector machines,” Pattern Recognition Letters, Vol. 27, Issue 12, pp. 1397-1404, Sep. 2006.

[9] P.J. Besser, and D.K. Jones, “Diffusion-tensor MRI: theory, experimental design and data analysis –a technical review,” NMR in Biomedicine, vol.15, pp. 456-467, 2002.