Image Registration by Minimization of Mapping Complexity

Andriy Myronenko and Xubo Song

Dept. of Science and Engineering, School of Medicine

Oregon Health and Science University, Portland, OR

myron@csee.ogi.edu, xubosong@csee.ogi.edu

Abstract

The criterion for the correct spatial alignment is a key component in image registration. We formulate the registration problem as one that finds the spatial and intensity mappings of minimal complexity that make images exactly equal. We do not assume any parametric forms of these functions, and estimate them within variational calculus.

We analytically solve for non-stationary intensity mapping, eliminate it from the objective function and arrive with a new similarity measure. We name it the Mapping Complexity (MC) similarity measure, because it achieves the optimum when intensity and spatial mappings are of minimal complexity. Due to its general formulation, the similarity measure works both for complex intensity relationships (e.g. multimodal registration) and for spatially-varying intensity distortions. Our similarity measure can be interpreted as the one that favors one image to lie mostly within a span of the leading eigenvectors of the kernel matrix, where the kernel matrix is constructed from the second image. We introduce a fast algorithm to compute the similarity measure. In particular, we introduce a fast kernel vector product (FKVP) algorithm, which is of general interest in computer vision. We demonstrate the accuracy of the new similarity measure on several mono- and multi-modal examples with complex intensity non-uniformities.

1. Introduction

The criterion for the correct spatial alignment is a key component in image registration [13, 6]. In the simplest case, we can require two images to have equal intensities for the correct alignment. Unfortunately, this is rarely valid due to intensity artifacts, noise or different modalities of the images. In general, two transformations are required to make the images equal: spatial transformation and intensity normalization. To normalize image intensities one often assumes a particular intensity normalization function based on intensity relationship between the images. For instance, if the image intensities are related through a linear trans-

formation, then the normalization function includes only two parameters: scaling and shift in the intensity space. One can analytically solve for these parameters, eliminate them from the objective function, and arrive with a similarity measure that depends only on the spatial transformation. In this example the final similarity measure is the Correlation Coefficient [18]. Notice that the intensity relationship here is spatially stationary, which does not depend on spatial location. Popular similarity measures, including Sum-of-Squared-Differences (SSD), Correlation Coefficient (CC), Correlation Ratio (CR) and Mutual Information (MI), all assume a particular spatially stationary intensity relationship [18].

Real-world images often have spatially-varying intensity relationships. For instance, brain MRI images can be corrupted by slow-varying intensity fields; visual-band images can have illumination non-homogeneity and reflectance artifacts [13]. Unfortunately, the real non-stationary intensity relationship does not allow to simplify for intensity normalization parameters and arrive with similarity measure that dependends only on spatial transformation. One approach to account for non-stationary intensity relationship, used in Statistical Parametric Mapping (SPM), is to simultaneously normalize image intensities and spatially align the images [9, 1]. The intensity normalization was chosen to be a non-linear intensity transformation followed by a convolution filter. The nonlinear intensity transformation function is defined as a linear combination of some basis functions with spatially smooth-varying coefficients (which are defined using another liner combination of basis functions). The convolution filter has to be chosen manually for a specific problem or estimated from the images. In more recent work, Ashburner and Friston [1] proposed a probabilistic framework for joint registration, intensity normalization and segmentation, using alternating optimization of corresponding parameters. This method demonstrates accurate performance on brain MRI images. A particular parametric form of intensity normalization function is somewhat heuristic and may vary among the registration problems.

We formulate the registration problem as one that finds

the spatial and intensity mappings of minimal complexity that make images exactly equal. We do not assume any parametric forms of these functions, and estimate them within variational calculus. We analytically solve for non-stationary intensity normalization function, eliminate it from the objective function and arrive with a new similarity measure. The similarity measure depends only on the spatial transformation, which we estimate using a standard variational approach for image registration. The new similarity measure is based on general non-stationary intensity relationship. We name it the Mapping Complexity (MC) similarity measure, because it achieves the optimum when intensity and spatial mappings are of minimal complexity. Due to its general formulation, the similarity measure works both for complex intensity relationships (e.g. multimodal registration) and for spatially-varying intensity distortions. Our similarity measure can be interpreted as the one that favors one image to lie mostly within a span of the leading eigenvectors of the kernel matrix, where the kernel matrix is constructed out of the second image. We give a fast algorithm to compute the similarity measure. In particular, we introduce a fast kernel vector produce algorithm, which is of general interest in computer vision. We demonstrate the accuracy of MC similarity measure on multiple mono- and multi-modal examples with complex intensity non-uniformities.

2. Method

The goal of image registration is to spatially align two given images A and B. We define the correct registration as the one that represents A in terms of B with minimal complexity. Mathematically, we want to find a spatial mapping $\mathcal{T}$ and intensity mapping $\mathcal{F}$ of minimal complexity that make images equal, $A = \mathcal{F}(\tilde{B}(\mathcal{T}^{-1}))$, or equivalently[1]

$$\min_{\mathcal{F},\mathcal{T}} \mathcal{K}(\mathcal{F},\mathcal{T}) \text{ s.t. } A(\mathcal{T}) = \mathcal{F}(\tilde{B}) \quad (1)$$

Both $\mathcal{F}$ and $\mathcal{T}$ are the functions of image attributes. Specifically, we define the attributes of the image B as its intensity and spatial features $\tilde{B} = (B(\mathbf{x}), \mathbf{x})$. Eq. 1 can be reinterpreted as the *Occam's razor* principle, or to prefer the simplest mapping among multiple equivalent ones that make the images equal.

We proceed to define the mapping complexity as a norm of the corresponding functions $\mathcal{K}(\mathcal{F},\mathcal{T}) = \mathcal{K}(\mathcal{F}) + \mathcal{K}(\mathcal{T})$, where

$$\mathcal{K}(\mathcal{F}) = \int \sum_{l=0}^{\infty} \frac{\beta^{2l}}{l! 2^l} \left\| D^l \mathcal{F}(\mathbf{z}) \right\|^2 d\mathbf{z} \quad (2)$$

penalizes all order derivatives of the function, weighted by a parameter β [23, 4]. Here D is a derivative operator so

that $D^{2l}\mathcal{F} = \nabla^{2l}\mathcal{F}$ and $D^{2l+1}\mathcal{F} = \nabla(\nabla^{2l}\mathcal{F})$, ∇ is the gradient operator and ∇^2 is the Laplacian operator.

We first solve analytically for $\mathcal{F}$ regardless of $\mathcal{T}$. We rewrite the constrained minimization problem of Eq. 1, using the quadratic penalty method [16], as

$$E(\mathcal{F},\mathcal{T}) = \mathcal{K}(\mathcal{F}) + \mathcal{K}(\mathcal{T}) + \frac{1}{\mu} \int \left(A - \mathcal{F}(\tilde{B}) \right)^2 d\mathbf{x} \quad (3)$$

where $\mu > 0$ is a penalty parameter. Discretizing images in Eq. 3 at N pixel locations, we find the minimum of $\mathcal{F}$ function using standard variational calculus. Solving the corresponding Euler-Lagrange equation, we obtain that $\mathcal{F}$ is in the form [23, 4]:

$$\mathcal{F}(\mathbf{z}) = \sum_{i=1}^{N} w_i G(\tilde{B}, \mathbf{z}); \text{ where } \mathbf{w} = (\mathbf{G} + \mu\mathbf{I})^{-1}\mathbf{a}; \quad (4)$$

where $\mathbf{w} = [w_1, .., w_N]^T$ is a vector of coefficients, G is a Green's function corresponding to the operator in Eq. 2, which is a Gaussian function with width parameter β [23, 21, 4], $\mathbf{a}_{N \times 1}$ is a vector of A image intensities, the kernel matrix $\mathbf{G}_{N \times N}$ is square symmetric (semi) positive definite with elements:

$$g_{ij} = e^{-\left(\frac{(B(\mathbf{x}_i) - B(\mathbf{x}_j))^2}{2\sigma_{int}^2} + \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma_{spat}^2} \right)}; \quad i,j = 1..N; \quad (5)$$

where we have absorbed the Gaussian normalization constant in μ and have intentionally assign different weights in intensity σ_{int} and spatial σ_{spat} directions[2]. Setting $\mu = 0$, we obtain the solution to the exact constraint $A = \mathcal{F}(\tilde{B})$, but it is often usefull to relax this equality by treating μ as a weight parmeter. Finally, substituting Eq. 4 into Eq. 3, we obtain

$$\min_{\mathcal{T}} E(\mathcal{T}) = \mathcal{K}(\mathcal{T}) + \mathbf{w}^T \mathbf{G}\mathbf{w} + \frac{1}{\mu} \|\mathbf{a} - \mathbf{G}\mathbf{w}\|^2 =$$

$$\mathcal{K}(\mathcal{T}) + \mathbf{a}^T(\mathbf{G} + \mu\mathbf{I})^{-1}\mathbf{a}, \text{ where } \mathbf{a} = A(\mathcal{T}). \quad (6)$$

where the quadratic term $\mathbf{w}^T \mathbf{G}\mathbf{w}$ is obtained from $\mathcal{K}(\mathcal{F})$ after substituting the kernel form (Eq. 4) [4]. To summarize, we have analytically found the minimizer for the $\mathcal{F}$ mapping and eliminated it. The final objective function depends only on the spatial transformation $\mathcal{T}$. The last term here can be seen as a new similarity measure.

Analysis: To analyze the performance of the new similarity measure, we decompose the Gaussian kernel matrix $\mathbf{G}$

---

[1]We assume that spatial transformation $\mathcal{T}$ is bijective.

[2]The actual width of the Gaussian is a single parameter β, which comes from the Eq. 2. However, non-equal weights σ_{int} and σ_{spat} help to balance the influence of intensity and spatial information. More precisely, non-equal weights are obtained either by rescaling the image intensities and spatial coordinates in advance or by explicitly defining different weights in the norm definition (Eq. 2).

in terms of its eigenvalues $\Lambda = \mathrm{d}[\lambda_1, .., \lambda_N]$, $\lambda_1 \geq \lambda_n \geq \lambda_N \geq 0$ and eigenvectors $\mathbf{Q} = [\mathbf{q}_1, ..\mathbf{q}_n] : \mathbf{G} = \mathbf{Q}\Lambda\mathbf{Q}^T$. We can rewrite the similarity measure as

$$E(\mathcal{T}) = \mathbf{a}^T(\mathbf{G} + \mu\mathbf{I})^{-1}\mathbf{a} = \mathbf{a}^T(\mathbf{Q}\Lambda\mathbf{Q}^T + \mu\mathbf{I})^{-1}\mathbf{a} =$$

$$\mathbf{a}^T\mathbf{Q}(\Lambda + \mu\mathbf{I})^{-1}\mathbf{Q}^T\mathbf{a} = \sum_{n=1}^{N} \frac{1}{\lambda_n + \mu}(\mathbf{q}_n^T\mathbf{a})^2 \quad (7)$$

Thus, by minimizing Eq. 7, we are looking for the transformed image A that mostly lies within a subspace spanned by the leading eigenvectors[3] of the kernel matrix $\mathbf{G}$ (formed out of the image B). The projections of $\mathbf{a}$ onto the other eigenvectors (corresponding to small eigenvalues) are more heavily penalized.

3. Implementation

Spatial transformation: Recall that $\mathcal{K}(\mathcal{T})$ measures the complexity of the spatial transformation $\mathcal{T}$ and may or may be in the form of Eq. 2. We follow the standard variational approach for the transformation estimation [14]. We set $\mathcal{T}(\mathbf{x}) = \mathbf{x} + \mathbf{u}(\mathbf{x})$[4] and define $\mathcal{K}(\mathcal{T}) = w \int \|\triangle\mathbf{u}(\mathbf{x})\|^2 \, d\mathbf{x}$, where w is a weight parameter. Thus, the objective function takes the form:

$$E(\mathbf{u}) = \mathbf{a}^T(\mathbf{G} + \mu\mathbf{I})^{-1}\mathbf{a} + w \int \|\triangle\mathbf{u}(\mathbf{x})\|^2 \, d\mathbf{x} \quad (8)$$

The solution for the displacement field $\mathbf{u}$ (and thus for $\mathcal{T}$) is obtained iteratively. We use fast FFT-based solvers to update the displacement field [14].

The computational bottleneck: The bottleneck of the method is to compute the inverse matrix $(\mathbf{G} + \mu\mathbf{I})_{N \times N}$ (once), and the inverse matrix vector product $(\mathbf{G} + \mu\mathbf{I})^{-1}\mathbf{a}$ (at each registration iteration). Not only the exact inversion requires enormous computational load $\mathcal{O}(N^3)$ (where N is a number of image pixels), but the construction of the $\mathbf{G}$ matrix itself is prohibited due to the large memory requirement. Nevertheless, we propose a fast solution, taking advantage of the high structure of the kernel matrix $\mathbf{G}$.

The kernel matrix $\mathbf{G}$ (Eq.5) can be factorized into intensity $\mathbf{G}_I$ and spatial $\mathbf{G}_S$ kernels

$$\mathbf{G} = \mathbf{G}_I \circ \mathbf{G}_S; \; g_I^{ij} = e^{-\frac{(B(\mathbf{x}_i) - B(\mathbf{x}_j))^2}{2\sigma_{int}^2}}; \; g_S^{ij} = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma_{spat}^2}} \quad (9)$$

where $\circ$ is the Hadamard (elementwise) product. We first consider a special cases, when $\mathbf{G} = \mathbf{G}_I$, that is $\mathcal{F}$ depends

---

[3]Corresponding to the largest eigenvalues.

[4]Notice that the transformation $\mathcal{T}$ depends only on spatial coordinates, but in general we can include intensity and other image attributes, e.g in order for different anatomical regions to have different transformation properties.

only on intensities. Matrix $\mathbf{G}_I$ has rank K, where $K \ll N$ is a number of the distinct intensity levels. Let $\mathbf{c}_{K \times 1}$ be a vector of sorted distinct intensity levels and $\tilde{\mathbf{G}}_I$ be a kernel matrix of these elements,then $\mathbf{G}_I$ decomposes as

$$\mathbf{G} = \mathbf{G}_I = \mathbf{P}\tilde{\mathbf{G}}_I\mathbf{P}^T; \; \tilde{g}_{ij} = e^{-(c(i)-c(j))^2/2\sigma_{int}^2} \quad (10)$$

where $\mathbf{P}_{N \times K}$ is all zeros, but $\mathbf{P}(n, Ind(B(\mathbf{x}_n))) = 1$ for $n = 1..N$ and $Ind(B(\mathbf{x}_n)) = 1..K$ is an integer index of the corresponding intensity level. Using the Woodbury identity [17], we obtain

$$(\mathbf{P}\tilde{\mathbf{G}}_I\mathbf{P}^T + \mu\mathbf{I})^{-1} = \frac{1}{\mu}\mathbf{I} - \frac{1}{\mu}\mathbf{P}(\mu\tilde{\mathbf{G}}_I^{-1} + \mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T \quad (11)$$

The inside matrix inversion can be precomputed in $\mathcal{O}(K^3)$. Thus, the product $(\mathbf{G} + \mu\mathbf{I})^{-1}\mathbf{a}$ requires only $\mathcal{O}(N + K^2)$ multiplications.

In the general case, the kernel matrix is built using both intensity information and spatial information (Eq. 5). We propose to compute the inverse approximately through lowrank matrix approximation. We estimate L leading eigenvectors $\tilde{\mathbf{Q}}$ and eigenvalues $\tilde{\Lambda}$ of $\mathbf{G}$, where $L \ll N$, through the Lanczos algorithm [7], and use the Woodbury identity [17]:

$$(\mathbf{G}+\mu\mathbf{I})^{-1} \simeq (\tilde{\mathbf{Q}}\tilde{\Lambda}\tilde{\mathbf{Q}}^T+\mu\mathbf{I})^{-1} = \frac{1}{\mu}\mathbf{I}-\frac{1}{\mu}\tilde{\mathbf{Q}}(\mu\tilde{\Lambda}^{-1}+\mathbf{I})^{-1}\tilde{\mathbf{Q}}^T \quad (12)$$

Such approximation is precomputed before the registration iterations. The product $(\mathbf{G} + \mu\mathbf{I})^{-1}\mathbf{a}$ takes only $\mathcal{O}(LN)$.

To estimate L leading eigenvectors and eigenvalues of $\mathbf{G}$, we use the Lanczos algorithm [7] (we used Matlab "eigs" implementation). Lanczos algorithm is iterative and requires the forward matrix-vector product $\mathbf{Ga}$ at each iteration. Here we introduce the fast kernel vector product (FKVP) algorithm that computes $\mathbf{Ga}$ exactly in $\mathcal{O}(KN \log N)$ with $\mathcal{O}(N)$ memory requirement. The FKVP algorithm utilizes the reach structure of the kernel matrix $\mathbf{G}$ in image processing.

FKVP: We use the idea that the full kernel matrix $\mathbf{G} = \mathbf{G}_I \circ \mathbf{G}_S$, $\mathbf{G}_I$ is low rank and $\mathbf{G}_S$ is block Toeplitz. The fast algorithm to find $\mathbf{Ga}$ is

$$\mathbf{Ga} = \sum_{i=1}^{K} \mathbf{P}(:, i) \circ (\mathbf{G}_S\tilde{\mathbf{a}}_i); \; \tilde{\mathbf{a}}_i = \left(\mathbf{P}\tilde{\mathbf{G}}_I(:, i)\right) \circ \mathbf{a} \quad (13)$$

where $\mathbf{P}$ and $\tilde{\mathbf{G}}_I$ are form Eq. 10 and $(:, i)$ denotes the ith column of the corresponding matrix. The product $\mathbf{G}_S\tilde{\mathbf{a}}_i$ takes $\mathcal{O}(N \log N)$, because $\mathbf{G}_S$ is symmetric block Toeplitz. To see it, consider the following 2D case of block Toeplitz matrix vector product $\mathbf{G}_S\tilde{\mathbf{a}}_i$

$$\mathbf{G}_S\tilde{\mathbf{a}}_i = (\mathbf{G}_S^{x_2} \otimes \mathbf{G}_S^{x_1})\tilde{\mathbf{a}}_i = \mathbf{G}_S^{x_2} Mat(\tilde{\mathbf{a}}_i)\mathbf{G}_S^{x_1} \quad (14)$$

where $\otimes$ is the Kronecker product, each $\mathbf{G}_S^{x_d}$ has elements $\left(\mathbf{G}_S^{x_d}\right)_{ij} = e^{-\frac{(x_i^d - x_j^d)}{2\sigma_{spat}^2}}$; $i, j = 1..N_d$, and N_d is the image size in the dth direction. $Mat(\tilde{\mathbf{a}}_i)$ denotes the matricize operation (or to reshape a vector into a matrix in Matlab notations). In the D-dimensional case, $Mat(\tilde{\mathbf{a}}_i)$ reshapes the vector into the D-dimensional array (or tensor) and each of the $\mathbf{G}_S^{x_d}$ matrices ($i = 1, .., D$) multiplies consecutively along the ith principal tensor direction. Each of the matrix vector product $\mathbf{G}_S^{x_d}\mathbf{z}$ is computed by embedding the Toeplitz matrix $\mathbf{G}_S^{x_d}$ into the circular matrix and using the FFT [2]. Thus the total computation complexity of forward matrix vector product $\mathbf{Ga}$ is $\mathcal{O}(KN \log N)$ with no explicit matrix constructions.

4. Results

We have implemented the algorithm in Matlab, and tested on a AMD Opteron CPU 2GHz with 4GB RAM machine. We empirically found the parameter values $\mu = 0.1$, $w = 1$, $\sigma_{int} = 0.1$, $\sigma_{spat} = 0.5M$ (where M is the size of the smallest image dimension) to be satisfactory, which we use in all experiments unless explicitly stated. We demonstrate the performance of the MC similarity measure on the BrainWeb MRI images [5]. Figure 1 shows the T1-weighted, T2-weighted and proton density (PD) images from the same subject. For testing, we used 2D images (217×181 pixels). All images are intensity normalized to $[0, 1]$ interval. We add an artificial intensity distortions and a spatial deformation to the test images and evaluate the registration performance. To simulate the synthetic spatial deformation we put a uniform grid of 6×6 control points over the image, randomly perturb them and interpolate the image using the Thin Plate Spline (TPS) [3]. This way we obtain smooth locally varying synthetic deformations. We compute the average absolute transformation error between the true and estimated transformations to evaluate the registration performance as $\frac{1}{2N} \sum |\mathbf{x}_{true} - \mathbf{x}_{estimated}|$. We do not include the area outside the scull (roughly found by simple thresholding) during registration and evaluation of the transformation accuracy.

Figure 2 shows the registration example where one image (the source image) is a deformed version of the other image (the reference image) with no intensity artifacts. In this simple example the images have spatially stationary identity intensity relationship. We set $\sigma_{spat} = \infty$; we construct a kernel matrix from the intensity information only $\mathbf{G} = \mathbf{G}_I$. In this case, the inverse matrix construction (Eq. 11) takes less than a second. The registration algorithm reduces the absolute transformation error from 5.65 to 0.58 per pixel. The registered image is visually almost identical to the reference image. The composite view, obtained by overlapping edges extracted from the registered image over
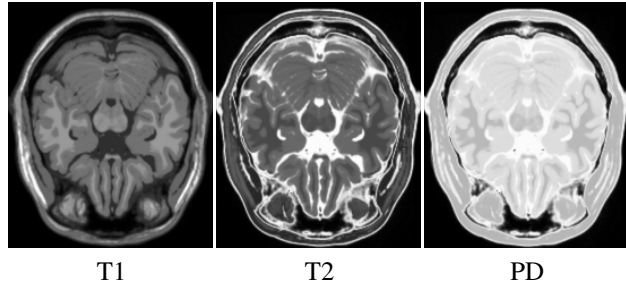

T1　　　　T2　　　　PD

Figure 1. The set of test images: T1-weighted, T2-weighted and proton density (PD) MRI images. The images are spatially aligned.

(a) reference　　(b) source　　(c) true transform

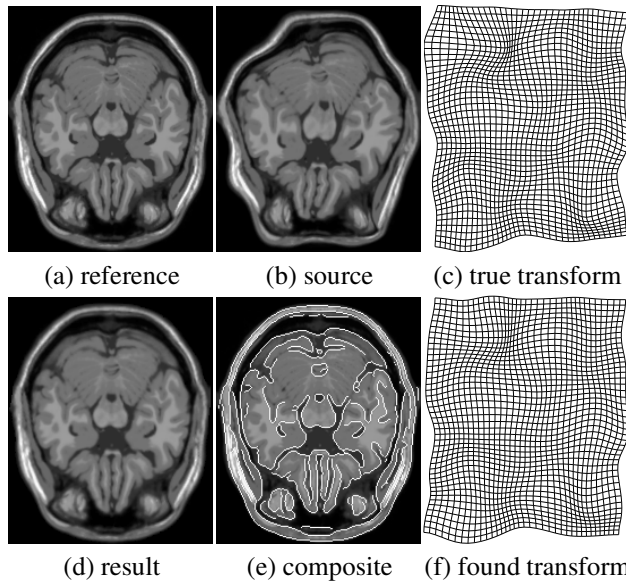(d) result　　(e) composite　　(f) found transform

Figure 2. Monomodal (T1) registration with spatial deformation only. We register the source image (b) onto the reference image (a). The absolute transformation error between the true (c) and estimated transformation (f) is reduced from 5.65 to 0.58 per pixel. The composite view (e) demonstrates the accurate alignment.

the reference one, also demonstrates the accurate alignment.

In Figure 3 and Figure 4, we tested the method for multi-modal image registration, by registering T1-T2 and PD-T2 image modalities with no intensity artifacts. Once again, we construct the kernel matrix from the intensity information only: $\mathbf{G} = \mathbf{G}_I$. The registration algorithm reduces the absolute transformation error from 5.43 to 1.08 per pixel for T1-T2 image pair and from 5.12 to 1.22 per pixel for PD-T2 image pair. The registration results are accurate.

Finally, we demonstrate the algorithm performance on mono- and multi-modal examples with spatially varying intensity nonuniformity. We simulate a multiplicative intensity distortion field by mixture of random low-frequency 2D cosine functions. Precisely we set a matrix of all zeros (same size as images), randomly initialized the first 25 coefficients and take the inverse discrete cosine transform.

(a) reference   (b) source   (c) true transform   (d) intensity distortion field

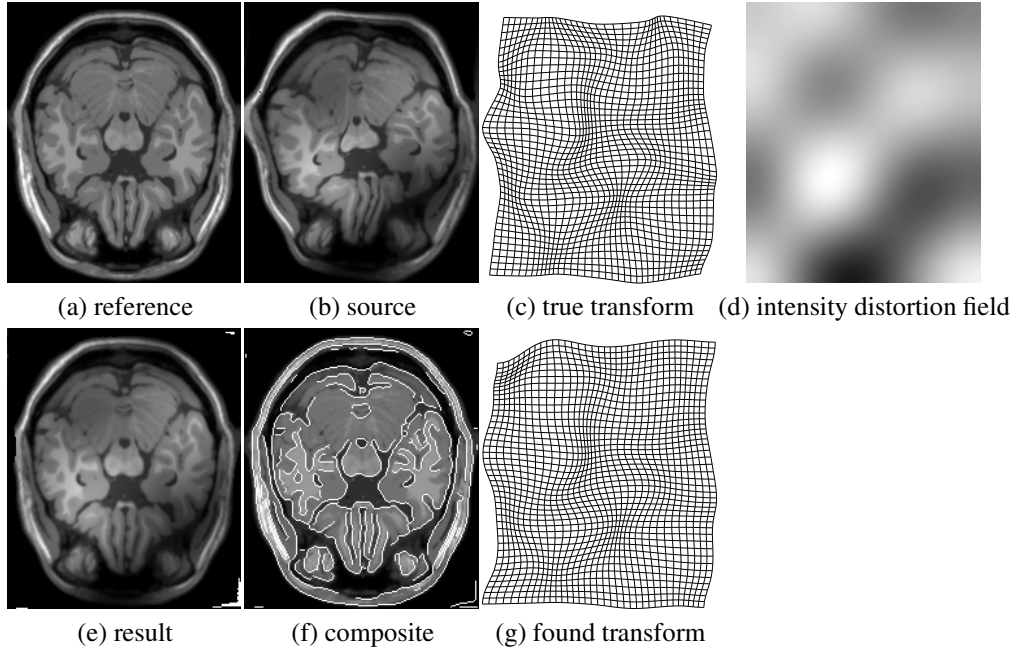(e) result   (f) composite   (g) found transform

Figure 5. Monomodal (T1) registration with spatial deformation and intensity non-uniformity. We register the source image (b) onto the reference image (a). The source image (b) includes a synthetic multiplicative intensity field (d). The absolute transformation error between the true (c) and estimated transformation (f) is reduced from 5.12 to 1.23 per pixel. The composite view (e) demonstrates the accurate alignment.

(a) reference   (b) source   (c) true transform   (d) intensity distortion field

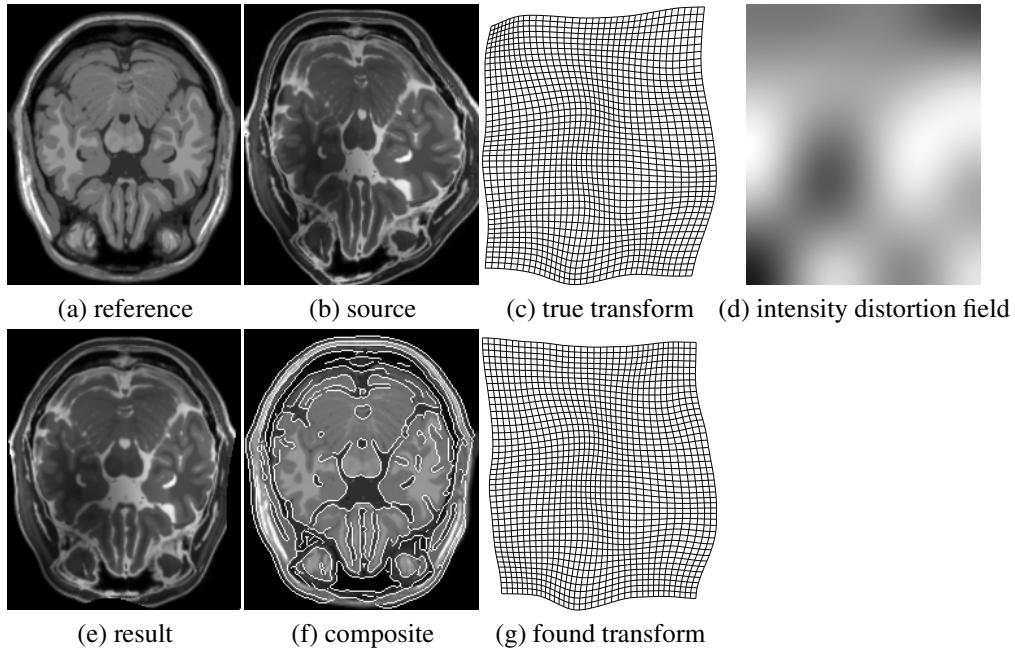(e) result   (f) composite   (g) found transform

Figure 6. Multimodal (T1-T2) registration with spatial deformation and intensity non-uniformity. We register the source image (b) onto the reference image (a). The source image (b) includes a synthetic multiplicative intensity field (d). The absolute transformation error between the true (c) and estimated transformation (f) is reduced from 4.96 to 1.03 per pixel. The composite view (e) demonstrates the accurate alignment.

Figure 5 demonstrates a monomodal (T1) image registration with a similar set-up as in the Figure 2, but the source image is also corrupted by the simulated multiplicative intensity field. In this case we have to compute the full matrix inverse (Eq. 12) through low-matrix approximation. To speed up the computations we reduce the colorspace of the reference

(a) reference   (b) source   (c) true transform

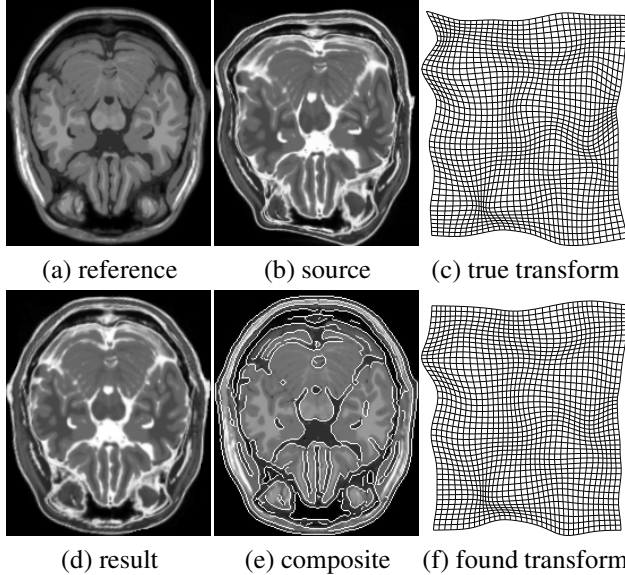(d) result   (e) composite   (f) found transform

Figure 3. Multimodal (T1-T2) registration with spatial deformation only. We register the source image (b) onto the reference image (a). The absolute transformation error between the true (c) and estimated transformation (f) is reduced from 5.43 to 1.08 per pixel. The composite view (e) demonstrates the accurate alignment.

(a) reference   (b) source   (c) true transform

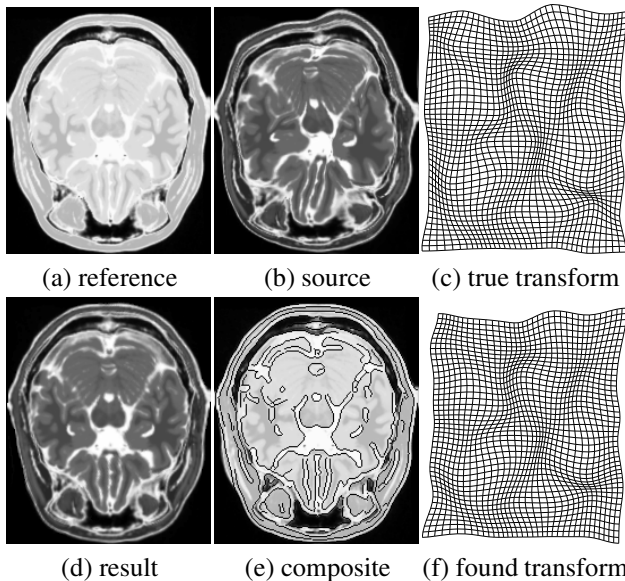(d) result   (e) composite   (f) found transform

Figure 4. Multimodal (PD-T2) registration with spatial deformation only. We register the source image (b) onto the reference image (a). The absolute transformation error between the true (c) and estimated transformation (f) is reduced from 5.12 to 1.22 per pixel. The composite view (e) demonstrates the accurate alignment.

image from 256 to 15 distinct intensity levels ($K = 15$) by piecewise thresholding. We precompute $L = 200$ (out of 39277!) leading eigenvectors and eigenvalues of $\mathbf{G}$, which takes around 2 min. The registration algorithm reduces the average absolute transformation error from 5.24 to 1.23 per

Table 1. The average transformation error after the registration (in pixels). The average initial transformation error was 5.12 per pixel. We have conducted 20 experiments for each combinations of images. The registration performance is accurate.

T1-T1	T1-T2	PD-T2	PD-T1
1.15 ± 0.52	1.32 ± 0.63	1.41 ± 0.74	1.39 ± 0.68

pixel and required around 3 min. Using the same set up, Figure 6 demonstrates a multimodal (T2-T1) image registration with a source image corrupted by multiplicative intensity field. The registration algorithm reduces the average absolute transformation error from 4.96 to 1.03 per pixel.

We have conducted 20 experiments for each combinations of image pairs: T1-T1, T1-T2, PD-T2, PD-T1. At each run we artificially randomly deformed one of the images and corrupt it with random multiplicative intensity field, similar to the ones shown in Figures 5,6. The average transformation error before the registration was 5.12 per pixel. The average transformation error after the registration is shown in Table 1. The registration performance is accurate.

For the comparison, we have also tested registration with the SAD, SSD, CC, CR, MI, NMI similarity measures, implemented in Deformable Registration using Discrete Optimization (DROP) software [11] and in Image Registration Toolkit (ITK) [19]. The performance of these similarity measures was accurate with no intensity non-uniformity. For the case of spatially varying intensity distortions, these similarity measures do not give satisfactory results, because of the spatially stationary underlying intensity relationship assumption. Some other registration methods, including SPM [9, 1] and ones based on local MI [12], implicitly assume intensity non-stationarity and can produce a reasonable registration results for our test images.

We have also tested the algorithm on the full 3D MRI images. The registration performance is equivalent to the one shown for 2D case. The computational time to pre-compute $L = 50$ largest eigenvectors was around 5 minutes. The size of each eigenvector is equal to the size of the image, which takes a big amount of memory in 3D case. As the computational accuracy improves for the better approximation (bigger L), the choice of the number of leading eigenvector is a trade-off between the computational load and accuracy of the registration. We discuss this and some other computational aspects in Sec. 5.

5. Discussion

Inverse kernel vector product: In Sec. 3, we have proposed to approximate the kernel matrix $\mathbf{G}$ in terms of its leading eigenvectors and eigenvalues. Such approximation is adequate for our problem. Recall that we are trying to transform image A so that it mostly lies withing a subspace

of the leading eigenvectors. With a low rank matrix approximation approach, we penalize all the smallest remaining eigenprojections equally (instead of weighting them). For the fast forward product $\mathbf{Ga}$, we have proposed the fast kernel vector product (FKVP) algorithm, which requires $\mathcal{O}(KN\log N)$ operations. Alternatively, one can compute $\mathbf{Ga}$ approximately using Fast Gaussian Transform (FGT) in $\mathcal{O}(pN)$, where p is a multiplicative constant [10, 22]. Unfortunately this constant rapidly increases for the higher accuracy approximations. FGT works better for the data easily clustered, which is not the case for digital images where each data point represent intensity and spatial coordinates. We note that FGT is a general algorithm, whereas our FKVP work only for the digital images, taking advantage of its structure. We prefer FKVP, because it computes the product exactly with a similar to FGT complexity (empirically FKVP is much faster).

The low rank matrix approximation is not the only way to compute $(\mathbf{G} + \mu\mathbf{I})^{-1}\mathbf{a}$ quickly. We can also compute it using linear conjugate gradients (CG) [16], as a solution to the symmetric positive definite (p.d.) linear system $(\mathbf{G} + \mu\mathbf{I})\mathbf{z} = \mathbf{a}$. This has the advantage of not requiring the eigen decomposition and the storage of the leading eigenvectors, but unfortunately has to be done at every iteration of the registration algorithm. In general, linear CG requires at most N matrix vector $\mathbf{Ga}$ products, which can we can compute fast through FKVP. Also, we can initialize the linear CG from the previous registration iteration (which will be close to the solution) rather than solving each system anew, and we can run only a few linear CG steps and obtain an approximate but good enough solution. Thus the computational complexity of linear CG in our problem is $\mathcal{O}(pKN\log N)$ per registration iteration, where $p < N$ is a number of CG steps. Such approach is much slower than low matrix approximation strategy, but does not require storage of eigenvectors, and can be a choice for the large data-set problems.

Colormap size: Recall that the computational complexity of FKVP algorithm is $\mathcal{O}(KN\log N)$, where K is the nubmer of distinct intensity levels (or colormap size) in the reference image, and N is the number of pixels. Thus the computational time can be significantly reduced by resizing the image or by reducing the colormap size. As the reference image is usually not intensity corrupted (e.g. when the reference image is a model or an atlas), its colormap is small or can be reduced to small without a significant loss of accuracy. Here we have used a simple piece-wise thresholding method to roughly reduce the reference image colorspace to 15 intensity levels out of 256 initially. Other more accurate methods, including the histogram clustering and dithering, can be also used. In the case when the colormap of the reference image can not be significantly reduced, the FKVP is

still much faster than the direct approach, because the full colormap is still much smaller than the total number of image pixels.

Number of leading eigenvectors: The low-rank matrix approximation requires selecting the number of the L leading eigenvectors and eigenvalues of $\mathbf{G}$. The accuracy of the approximation and the whole registration procedure depends on L. The eigenspectra of Gaussian kernel matrix decays fast in general, but the optimal selection of the number of eigenvalues is a common problem in computer vision. In our case, a heuristic approach is to consider that for a special case $\mathbf{G} = \mathbf{G}_I$ the number of non-zero eigenvalues is K and thus in the full case, when $\mathbf{G} \simeq \mathbf{G}_I$, the number of the largest eigenvalues should be close to K. The choice of the number of leading eigenvector is a trade-off between the computational load and accuracy of the registration.

Intensity relationship: In this paper, we have used the following image relationship $A(\mathbf{x}) = \mathcal{F}(B(\mathbf{x}), \mathbf{x})$, with $\mathcal{F} : \mathbb{R}^{D+1} \rightarrow \mathbb{R}$. Alternatively, we can also use the constraint that image intensities are positive, e.g. using $A(\mathbf{x}) = \exp(\mathcal{F}(\log(B(\mathbf{x})), \mathbf{x}))$. This does not change the derivative significantly, except the images should be log transformed before the registration, i.e. $\mathbf{a} = \log(A)$. The positive mapping of grayscale images can be generalized for tensor-valued images using matrix exponent, in order to include the p.d. constraint.

Also, one can require the intensity mapping to be identity transformation when its norm is zero. This can be accomplished using relationships $A(\mathbf{x}) = B(\mathbf{x}) + \mathcal{F}(B(\mathbf{x}), \mathbf{x})$ or $A(\mathbf{x}) = B(\mathbf{x}) \exp(\mathcal{F}(\log(B(\mathbf{x})), \mathbf{x}))$, which leads to the objective function $\mathbf{r}^T(\mathbf{G}+\mu\mathbf{I})^{-1}\mathbf{r}$, where $\mathbf{r} = A(\mathbf{x})-B(\mathbf{x})$ or $\mathbf{r} = \log(A(\mathbf{x})) - \log(B(\mathbf{x}))$. All these different variants were briefly tested and have different performance depending on the problem. We do not include the detailed comparisons of such variants in this paper.

FKVP algorithm: We believe, the fast kernel vector product (FKVP) is of general interest in computer vision. For instance, kernel methods for image processing, including kernel PCA [20] and spectral clustering [15], require to find the eigenvectors of the kernel matrix (which is build from spatial intensity image attributes) and/or to find the forward kernel matrix vector product. Usually, people use different kernel approximations, including truncating of the kernel (finite support), subsampling of the images, FGT and Nystrom approximations [8]. The FKVP algorithm is directly applicable for such kernel methods in image processing with $\mathcal{O}(KN\log N)$ computational complexity. Compared to the other approaches, it is exact (no approximations) and is defined for general kernel (not necessarily

Gaussian). Also FKVP can be further accelerated by reducing the colormap size. Finally, the strong analogy of our image registration approach with kernel PCA and spectral clustering suggest further intuition and possible improvements of our similarity measure, which we reserve for future works.

6. Conclusion

We derived a new similarity measure, called the Mapping Complexity (MC) similarity measure. We derived it by formulating the registration problem as the one that finds the spatial and intensity mappings of minimal complexity that make images exactly equal. Due to its general formulation, the similarity measure is applicable for both complex intensity relationships (e.g. multimodal registration) and for spatially-varying intensity distortions. Our similarity measure can be interpreted as the one that favors one image to lie mostly within a span of the leading eigenvectors of the kernel matrix, where the kernel matrix is constructed from the second image. We have proposed a fast algorithm to compute the similarity measure. In particular, we introduce a fast kernel vector product (FKVP) algorithm, which is of general interest in computer vision. We have demonstrated the accuracy of the new similarity measure on several mono and multimodal examples with spatially varying intensity distortions. The MC similarity measure shows accurate performance on the given set of test images.

References

[1] J. Ashburner and K. Friston. Unified segmentation. *NeuroImage*, 26:839–851, 2005.

[2] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. SIAM, 1987.

[3] F. L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *PAMI*, 11(6):567–585, 1989.

[4] Z. Chen and S. Haykin. On different facets of regularization theory. *Neural Computation*, 14(12):2791–2846, 2002.

[5] C. Cocosco, V.Kollokian, R.-S. Kwan, and A. Evans. Brainweb: Online interface to a 3D mri simulated brain database. In *ICFMHB*, page S425, 1997.

[6] W. R. Crum, T. Hartkens, and D. L. G. Hill. Nonrigid image registration: Theory and practice. *The British Journal of Radiology*, 77(special issue):S140–S153, Dec. 2004.

[7] J. K. Cullum and R. A. Willoughby. *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*, volume 1. Cambridge, 2002.

[8] P. Drineas and M.W.Mahoney. On the nystrom method for approximating a gram matrix for improved kernel-based learning. *JMLR*, 6:2153–2175, 2005.

[9] K. Friston, J. Ashburner, C. Frith, J. Poline, J. D. Heather, and R. Frackowiak. Spatial registration and normalization of images. *Human Brain Mapping*, 2:165–189, 1995.

[10] L. GeenGard and J. Strain. The fast gauss transform. *SIAM J Sci Stat Comput*, 12(1):79–94, 1991.

[11] B. Glocker, N. Komodakis, N. Paragios, G. Tziritas, and N. Navab. Inter and intra-modal deformable registration: Contin. deform. meet efficient opt. linear progr. In *Inf. Proc. in Med. Imag.*, July 2007.

[12] G. Hermosillo. *Variational Methods for Multimodal Image Matching*. PhD thesis, University of Nice, 2002.

[13] D. L. G. Hill, P. G. Batchelor, M. Holden, and D. J. Hawkes. Medical image registration. *Physics in medicine and biology*, 46(3):R1–R45, Mar. 2001.

[14] J. Modersitzki. *Numerical Methods for Image Registration*. Oxford university press, 2004.

[15] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*, pages 849–856. MIT Press, 2001.

[16] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 1999.

[17] K. Petersen and M. Pedersen. The matrix cookbook. http://matrixcookbook.com, Sept. 2007.

[18] A. Roche, G. Malandain, and N. Ayache. Unifying maximum likelihood approaches in medical image registration. *Int. J. of Imaging Syst. and Technology: Sp. issue on 3D imaging*, 11:71–80, 2000.

[19] 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. Image Processing*, 18(8):712–721, Aug. 1999.

[20] B. Scholkopf, A. Smola, and K. Muller. Nonlinear component analysis as a kernel eigenvalue problem. Technical report, Max Planck Institute, 1996.

[21] B. Schölkopf and A. J. Smola. *Learning with Kernels*. The MIT Press, Cambridge, MA, 2002.

[22] C. Yang, R. Duraiswami, N. Gumerov, and L. Davis. Improved fast gauss transform and efficient kernel density estimation. In *ICCV*, pages 464–471, 2003.

[23] A. L. Yuille and N. M. Grzywacz. A mathematical analysis of the motion coherence theory. *International Journal of Computer Vision (IJCV)*, 3(2):155–175, June 1989.