Global Volumetric Image Registration Using Local Linear Property of Image Manifold

Hayato Itoh[1] Atsushi Imiya[2] Tomoya Sakai[3]

[1] Graduate School of Advanced Integration Science, Chiba University
[2] Institute of Management and Information Technologies, Chiba University, Yayoicho 1-33, Inage-ku, Chiba 263-8522, Japan
[3] Graduate School of Engineering, Nagasaki University, Bunkyo-cho 1-14, Nagasaki 852-8521, Japan

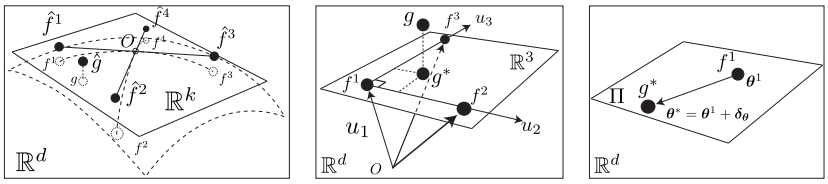
Abstract. We propose a three-dimensional global image registration method for a sparse dictionary. To achieve robust and accurate registration, which based on template matching, a large number of transformed images are prepared and stored in the dictionary. To reduce the spatial complexity of this image dictionary, we introduce a method of generating a new template image from a collection of images stored in the image dictionary. This generated template image allows us to achieve accurate image registration even if the population of the image dictionary is relatively small and the template has a small pattern perturbation. To further reduce the complexity, we compute a matching process in a low-dimensional Euclidean space projected by a random projection.

1 Introduction

We propose a three-dimensional global image registration method for a compressed sparse dictionary. Our method allows us to achieve accurate volumetric image registration even if the population of pregenerated images is relatively small. Furthermore, our method also achieves robust registration for template images with a small pattern perturbation.

Three-dimensional image registration mainly focuses on registration of point clouds and volumetric data. The former problem is applied to a point cloud representing depth map. Since this depth map expresses a terrain surface, registration is achieved as surface registration in three-dimensional space [1, 2]. For this surface registration, iterative closest algorithm is a well established method [3, 4]. The later problem mainly deals with volumetric medical data obtained by computed tomography, magnetic resonance imaging and positron emission tomography [5, 6].

Medical image registration categorised into linear and nonlinear methods. For nonlinear image registration, global image registration is used as preprocess because nonlinear image registration is mainly valid for local deformation between images. Nonlinear registration is used to detect optimally local transform between a template and the reference images, if the difference between these



(a) (b) (c)

Fig. 1. (a) Nearest neighbours of g searched for by the k-nearest-neighbour search on a manifold. Our method of projecting the manifold to a low-dimensional subspace. (b) Generation of a new entry in a dictionary. The input image g is projected onto the subspace spanned by three nearest neighbours. (c) Interpolation of parameter. For the new entry g^*, we interpolate the parameter θ^* of the image g^*. Here, $\mathit{\Pi}$ represents the parameter space of the transform.

two images are small and local [6]. Linear registration, however, detects global geometric relations between a template and the references [7].

In global image registration, the best geometrical transform between reference and template images is accomplished by computing the best geometrical matching between these two images. Therefore, for accurate registration, we are required to prepare as many template images as possible in the dictionary [8]. This implies that the spatial complexity of global image registration depends on the population of templates in the dictionary. The generation of new template images from existing templates reduces the spatial complexity of global image registration. However, in image registration with a sparse dictionary, we are required to generate a new entry and to compute the transform from this new entry simultaneously [9, 10].

Since the image pattern space is a curved manifold in higher-dimensional space, on the tangent space of this curved manifold, an image pattern is expanded to a finite Fourier series using local bases. This expansion means that local bases span a local part of the manifold. Figure 1(a) illustrates a manifold on a low-dimensional subspace [11, 12]. In the neighbourhood of an image pattern, image patterns can be expressed as a linear combination of this image and derivatives of this image. We call this property of an image manifold the local linear property. Figure 1(b) shows the generation of a template image. Combining these two local expressions for an image pattern implies two advantages for pattern generation. One is that we can compute the parameters of the transform. The other is that we can compute a new pattern that is sufficiently close to a reference image. Figure 1(c) shows the relation between a generated image g^* and the nearest neighbour in a local subspace. In Fig. 1(c), the perturbation $\boldsymbol{\delta}_\theta$ is small because the generated image g^* is close to the nearest neighbour. If the rotation angle, scaling factor and shear ratio are all small, these transforms can be expressed as a linear sum of the identity transform and linear transforms with parameters that define the type of transform. Therefore, we can decompose the affine matrix into a rotation, scaling and shear to estimate each transform.

To use the local linear property for image registration, a curved manifold of image patterns is generated from reference images. For the template image, we generate this curved manifold using the k-nearest-neighbour search (k-NNS) [13]. To reduce the temporal and spatial complexities of the search of the k-neighbourhood, a dimension-reduction method is used to generate the curved manifold. We adopt random projection as the dimension-reduction method [8–10].

Random projection is a metric-embedding method that approximately preserves distances between points in the original space [14]. Therefore, the random projection is used in the nearest-neighbour search (NNS) to speed up numerical computation [15]. Furthermore, the random projection preserves manifolds [16]. The validity of random projection for manifold learning of noiseless images, noisy images and text data is shown in [17]. We use the random projection to generate a manifold in a low-dimensional subspace.

In section 2, we summarise the linear subspace method for pattern recognition, the random projection for dimension reduction and the local linear property of an image pattern space. Then, in section 3, we derive relations between the global transform of an image and the local linear properties of an image pattern space. The relations in section 3 are used to derive a method of performing the global image registration using a dictionary with a small population of entries.

2 Mathematical Preliminaries

2.1 Global Image Registration

Setting $\mathit{\Pi}$ to be an appropriate parameter space for image generation, we assume that images are expressed as $f(\boldsymbol{x}, \boldsymbol{\theta}_i)$ for $\exists \boldsymbol{\theta}_i \in \mathit{\Pi}$, $\boldsymbol{x} \in \mathbb{R}^3$. The parameter $\boldsymbol{\theta}_i$ generates a transform for $f(\boldsymbol{x})$. We call the set of generated images $f(\boldsymbol{x}, \boldsymbol{\theta}_i)$ and parameters $\{\boldsymbol{\theta}_i\}_{i=1}^N$ a dictionary.

For the global alignment of images with respect to the region of interest Ω, we find the linear transformation $\boldsymbol{x}' = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{t}$ that minimises the criterion

$$R(f, g) = \sqrt{\int_{\Omega} |f(\boldsymbol{x}') - g(\boldsymbol{x})|^2 d\boldsymbol{x}} \tag{1}$$

for functions $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ defined on $\mathbb{R}^3$ such that

$$\int_{\mathbb{R}^3} |f(\boldsymbol{x})|^2 d\boldsymbol{x} < \infty, \quad \int_{\mathbb{R}^3} |g(\boldsymbol{x})|^2 d\boldsymbol{x} < \infty. \tag{2}$$

In image registration, we assume that the parameter $\boldsymbol{\theta}_i$ in $\mathit{\Pi}$ generates the affine coefficients $\boldsymbol{A}$ and $\boldsymbol{t}$. Solving the NNS problem using the dictionary, we can estimate the transform $\boldsymbol{A}$ and $\boldsymbol{t}$ as $\boldsymbol{\theta}_i$. The computational cost of a naive approach for the NNS is $\mathcal{O}(Nd)$, where N and d are the cardinality of the set of points in the metric space and the dimension of the metric space, respectively. The factor d in the nearest-neighbour search [14] is reduced by using the random projection. Furthermore, using the local linear property in section 2.2, we can also reduce N in the nearest NNS.

2.2 Local Eigenspace

Setting the Hilbert space H to be the space of patterns, we assume that the inner product (f, g) is defined in H. Let $f \in H$ and P be a pattern and an operator for a class, respectively. We then define the class $\mathcal{C} = \{f \,|\, Pf = f,\, P^*P = I\}$. For recognition, we construct P for $f \in \mathcal{C}$ while minimising $E[\|f - Pf\|_2]$ with respect to $P^*P = I$, where $f \in \mathcal{C}$ is the pattern for a class, I is the identity operator and E is the expectation in H. This methodology is known as the subspace method [11, 12]. For the practical calculation of P, we adopt the Karhunen-Loeve expansion for the construction of the eigenspace.

We deal with images $f(\boldsymbol{x})$ defined in the three-dimensional Euclidean space $\boldsymbol{x} = (x, y, z)^\top \in \mathbb{R}^3$. We assume that a small perturbation of the parameter causes a small geometrical transform of the image pattern, that is, we assume the relation $f(\boldsymbol{x}+\boldsymbol{\delta}, \boldsymbol{\theta}) = f(\boldsymbol{x}, \boldsymbol{\theta}+\boldsymbol{\psi})$. Therefore, a small perturbation of an image caused by parameter perturbation is replaced with a geometrical perturbation of the image, that is,

$$f(\boldsymbol{x}, \boldsymbol{\theta} + \boldsymbol{\psi}) = f(\boldsymbol{x} + \boldsymbol{\delta}) = f(\boldsymbol{x}) + \boldsymbol{\delta}^\top \nabla f(x, y), \tag{3}$$

where $\boldsymbol{\delta} \in \mathbb{R}^3$ is a perturbation vector. Setting f_x, f_y and f_z to $\partial_x f(\boldsymbol{x}), \partial_y f(\boldsymbol{x})$ and $\partial_z f(\boldsymbol{x})$, respectively, since

$$\int_{\mathbb{R}^3} f f_x d\boldsymbol{x} = 0, \int_{\mathbb{R}^3} f f_y d\boldsymbol{x} = 0, \int_{\mathbb{R}^3} f f_z d\boldsymbol{x} = 0, \tag{4}$$

$$\int_{\mathbb{R}^3} f_x f_y d\boldsymbol{x} = 0, \int_{\mathbb{R}^3} f_y f_z d\boldsymbol{x} = 0, \int_{\mathbb{R}^3} f_z f_x d\boldsymbol{x} = 0, \tag{5}$$

for images $g(\boldsymbol{x}) = f(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{t})$ with a small perturbation affine transform $\boldsymbol{A}$ and a small translation vector $\boldsymbol{t}$, we can assume the relation

$$g(\boldsymbol{x}) = a_0 f + a_1 f_x + a_2 f_y + a_3 f_z,\ \boldsymbol{x} \in \mathbb{R}^3. \tag{6}$$

Equation (6) implies that the number of independent images among the collections of images,

$$L(f) = \{f_{ij} | f_{ij}(\boldsymbol{x}) = \lambda f(\boldsymbol{A}_i \boldsymbol{x} + \boldsymbol{t}_j)\}_{i,j=1}^{p,q} \tag{7}$$

is four, if the domain of the image is $\mathbb{R}^3$. We can use the first four principal vectors of $L(f)$ as the local basis for image expression for a three-dimensional image. We call this property the local linear property and the space spanned by $\{f, f_x, f_y, f_z\}$ the local eigenspace. Figure 1(b) shows the projection of the input image g to the three-dimensional local subspace.

2.3 Three-Dimensional Affine Transform

To avoid the estimation of a translation, we set the origin of the coordinates to be the centre of an image. We assume small rotations around the x, y and z axes

of angles ϕ_1, ϕ_2 and ϕ_3, given by the transform matrices such that

$$\boldsymbol{R}_x = \boldsymbol{I} + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -\phi_1 \\ 0 & \phi_1 & 0 \end{pmatrix} = \boldsymbol{I} + \boldsymbol{R}'_x, \tag{8}$$

$$\boldsymbol{R}_y = \boldsymbol{I} + \begin{pmatrix} 0 & 0 & \phi_2 \\ 0 & 0 & 0 \\ -\phi_2 & 0 & 0 \end{pmatrix} = \boldsymbol{I} + \boldsymbol{R}'_y, \tag{9}$$

$$\boldsymbol{R}_z = \boldsymbol{I} + \begin{pmatrix} 0 & -\phi_3 & 0 \\ \phi_3 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \boldsymbol{I} + \boldsymbol{R}'_z. \tag{10}$$

Multiplying $\boldsymbol{R}_x, \boldsymbol{R}_y$ and $\boldsymbol{R}_z$, and ignoring terms of order larger than one, we have an arbitrary rotation expressed as

$$\boldsymbol{R}(\boldsymbol{\phi}) = \begin{pmatrix} 1 & -\phi_3 & \phi_2 \\ \phi_3 & 1 & -\phi_1 \\ -\phi_2 & \phi_1 & 1 \end{pmatrix} = \boldsymbol{I} + \boldsymbol{R}'_x + \boldsymbol{R}'_y + \boldsymbol{R}'_z = \boldsymbol{I} + [\boldsymbol{R}]_\times, \tag{11}$$

where $[\boldsymbol{R}]_\times$ is the outer-product operator of vector $\boldsymbol{\phi} = (\phi_1, \phi_2, \phi_3)^\top$.

For small scaling factors ϕ_4, ϕ_5 and ϕ_6, and small shearing ratios ϕ_7, ϕ_8, ϕ_9, ϕ_{10}, ϕ_{11} and ϕ_{12}, we have the scaling matrix and shearing matrix

$$\begin{pmatrix} 1+\phi_4 & 0 & 0 \\ 0 & 1+\phi_5 & 0 \\ 0 & 0 & 1+\phi_6 \end{pmatrix} = \boldsymbol{I} + \boldsymbol{\Lambda}, \tag{12}$$

$$\begin{pmatrix} 1 & \phi_7 & \phi_8 \\ \phi_9 & 1 & \phi_{10} \\ \phi_{11} & \phi_{12} & 1 \end{pmatrix} = \boldsymbol{I} + \boldsymbol{S}, \tag{13}$$

respectively.

Combining these rotation, scaling and shearing matrices in eqs. (11)-(13), we can define all affine transforms except translation. Multiplying these three matrices and ignoring terms of order larger than one, we have the affine transform matrix

$$\boldsymbol{A} = \boldsymbol{I} + [\boldsymbol{R}]_\times + \boldsymbol{\Lambda} + \boldsymbol{S} = \boldsymbol{I} + \boldsymbol{A}_\delta, \tag{14}$$

where the rotation, scaling and shear transforms are commutative since their transform matrices consist of only small-value elements. Here, $\boldsymbol{A}_\delta$ represents a small affine transform.

2.4 Neighbours of Template Image

For the reference image f and template image g in Hilbert space H, applying affine transforms $\{\boldsymbol{A}_i\}_{i=1}^N$ except for translation to f, we have the finite collection $\{f_i|\boldsymbol{A}_i\boldsymbol{x}\}_{i=1}^N$. For $0 < k \ll N$, let $\pi(i)$ be one-to-one injection from $1 \leq i \leq N$

to $1 \leq \pi(i) \leq k$ such that $\pi(i) \neq \pi(j)$ for $i \neq j$. Using $\pi(i)$, we define the k-neighbourhood $KN(g) \in L(f)$ of g. For a finite collection of images $\{f_i\}_{i=1}^N$, $KN(g)$ is a collection $\{f_{\pi(i)}\}_{i=1}^N$ that satisfies the inequalities

$$\|g - f_{\pi(1)}\|_2 \leq \|g - f_{\pi(2)}\|_2 \leq \cdots \leq \|g - f_{\pi(N)}\|_2 \tag{15}$$

where $\|\cdot\|_2$ is the L_2 metric on H.

2.5 Manifold Generation by Random Projection

We construct the image manifold of entries in the dictionary using the nearest-neighbour method. To reduce the time complexity of the nearest neighbourhood mesh on the image manifold, we adopt the random projection.

The random projection reduces the dimension of the discrete vector space while preserving both local and global topologies and geometries. The random projection satisfies the following theorem. For a set $X = \{\boldsymbol{x}_i\}_{i=1}^N$ of N points in d-dimensional Euclidean space, consider a mapping onto the set $\hat{X} = \{\hat{\boldsymbol{x}}_i\}_{i=1}^N$ in k-dimensional Euclidean space. For the vector $\boldsymbol{x} = (x_1, \ldots, x_d)^\top$, we define the Euclidean norm as $\|\boldsymbol{x}\|_2 = \left(\sum_{i=1}^d x_i\right)^{1/2}$. The Johnson-Lindenstrauss lemma indicates that there is a mapping approximately preserving the Euclid distance between two arbitrary points [18]. Setting $|\boldsymbol{x} - \boldsymbol{y}|_2$ to be the Euclidean distance between two points $\boldsymbol{x}$ and $\boldsymbol{y}$ in appropriate dimensional Euclidean space, the next therem is satisfied [19].

Theorem 1 *(Johnson-Lindenstrauss lemma). For a subspace with dimension* $\hat{d} \geq \hat{d}_0 = \frac{9 \log N}{\epsilon^2 - \frac{2}{3}\epsilon^3} + 1 = \mathcal{O}(\epsilon^{-2} \log N)$*, where* ϵ *is a real number such that* $0 < \epsilon < \frac{1}{2}$*, a set* X *of* N *d-dimensional points* $\{\boldsymbol{x}_i\}_{i=1}^N$ *and an integer* $\hat{d}$ *with* $\hat{d} \ll d$*, there exists a mapping* f *from* $\mathbb{R}^d$ *to* $\mathbb{R}^{\hat{d}}$ *such that*

$$(1 - \epsilon)|\boldsymbol{x}_j - \boldsymbol{x}_i|_2 \leq |\hat{\boldsymbol{x}}_j - \hat{\boldsymbol{x}}_i|_2 \leq (1 + \epsilon)|\boldsymbol{x}_j - \boldsymbol{x}_i|_2, \tag{16}$$

for all $i, j = 1, 2, \ldots, N$.

Therefore, setting $\boldsymbol{R}$ to be the random projection from $\mathbb{R}^d$ to $\mathbb{R}^{\hat{d}}$, Theorem 1 implies the relation

$$P(\,|\,|\boldsymbol{x} - \boldsymbol{y}|_2 - |\boldsymbol{R}\boldsymbol{x} - \boldsymbol{R}\boldsymbol{y}|_2\,| < \varepsilon) > 1 - \delta \tag{17}$$

for small positive constants ε and δ, where P is a probability distribution. To use these topological and geometrical properties for fast computation in the nearest neighbour method, we use sampled images to construct the image manifold of data in the dictionary.

Let $\mathbb{Z}^d$ be the integer grid in $\mathbb{R}^d$, Setting $\mathbf{D}$ and Δ to be a finite subset of $\mathbb{Z}^d$ and a positive number that defines the resolution of sampling, respectively, the distance

$$D(f, g) = \sqrt{\int_{\mathbb{R}^d} |f(\boldsymbol{x}) - g(\boldsymbol{x})|^2 d\boldsymbol{x}} \tag{18}$$

is approximately computed as

$$D(f,g) = \sqrt{\sum_{\boldsymbol{z}\in\mathbf{D}\subset\mathbb{Z}^d} |f(\Delta\boldsymbol{x}) - g(\Delta\boldsymbol{z}))|^2 \Delta}, \tag{19}$$

for functions $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ defined on $\mathbb{R}^d$.

By expressing $\{f(\Delta\boldsymbol{z})\}_{\boldsymbol{z}\in\mathbf{D}}$ and $\{g(\Delta\boldsymbol{z})\}_{\boldsymbol{z}\in\mathbf{D}}$ as finite vectors $\boldsymbol{f}$ and $\boldsymbol{g}$, respectively, eq. (19) is expressed as

$$D(f,g) = |\boldsymbol{f} - \boldsymbol{g}|_2 \tag{20}$$

if we set $\Delta = 1$. Using the random projection, the distance between $\boldsymbol{f}$ and $\boldsymbol{g}$ is computed as

$$D(f,g) \approx |\boldsymbol{R}\boldsymbol{f} - \boldsymbol{R}\boldsymbol{g}|, \tag{21}$$

for functions $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ defined on $\mathbb{R}^n$ such that

$$\int_{\mathbb{R}^n} |f(\boldsymbol{x})|^2 d\boldsymbol{x} < \infty, \ \int_{\mathbb{R}^n} |g(\boldsymbol{x})|^2 d\boldsymbol{x} < \infty. \tag{22}$$

Therefore, by searching $KN(g)$ in $\hat{d}$-dimensional Euclidean space with the random projection, we obtain the discrete version of $KN(g)$. For practical computation, we adopt an efficient random projection [20].

3 Local Linear Method

Using the local linear property of images in the image space, we first generate an image in a sparse dictionary. To register a template g, using the generated image g^*, we next estimate the small affine transform between the generated image g^* and the nearest neighbour f^1 of g in the dictionary. From the generated image and the estimated transform, the local linear method can generate new entries in the dictionary. Figure 1 shows a flow of this local linear method.

For image generation, we use the k nearest neighbours of g in the dictionary. Let $\{f^i\}_{i=1}^k \in \mathcal{L}(g)$, be the ith neighbour of g. The random projection preserves the pairwise distances between vectorised images. Therefore, f^i is searched for in a random projected space. For a template $g(\boldsymbol{x})$, we assume $g(\boldsymbol{x}) = f^1(\boldsymbol{A}\boldsymbol{x}, \theta) + \epsilon$, where $\boldsymbol{A}$ gives the best matching between g and f^1, and ϵ is a small difference between the reference pattern and the registered template pattern. For three-dimensional images, using the local linear property, we can approximate the space spanned by $\{u_i\}_{i=1}^4$ using the space spanned by $\{g\} \cup \{f^i\}_{i=1}^4$ if the data space $\mathcal{L}(g)$ is not extremely sparse. Using Gram-Schmidt orthonormalisation for $\{f^i\}_{i=1}^4$, we obtain the basis $\{u_i\}_{i=1}^4$. Projecting the template to the space spanned by $\{u_i\}_{i=1}^4$, we obtain a new image,

$$g^* = \sum_{i=1}^4 b_i u_i, \tag{23}$$

from a triplet of preprepared entries in the dictionary. Here, $\{b_i\}_{i=1}^4$ represents the coefficients of the linear combination.

For the projected template image and its nearest neighbour $f^1(\boldsymbol{x},\boldsymbol{\theta})$, using the Taylor expansion, we have the relation

$$\begin{aligned} g^* = f^1(\boldsymbol{x}+\boldsymbol{\delta},\boldsymbol{\theta}) = f^1(\boldsymbol{A}\boldsymbol{x},\boldsymbol{\theta}) = f^1((\boldsymbol{I}+\boldsymbol{A}_\delta)\boldsymbol{x},\boldsymbol{\theta}) \\ = f^1(\boldsymbol{x},\boldsymbol{\theta}) + (\boldsymbol{A}_\delta\boldsymbol{x})^\top \nabla f^1(\boldsymbol{x},\boldsymbol{\theta}) \end{aligned} \tag{24}$$

if the higher order terms with respect ***delta*** is sufficiently small. For the transform matrix $\boldsymbol{A}_\delta$, we have the relation

$$(\boldsymbol{A}_\delta\boldsymbol{x})^\top \nabla f^1(\boldsymbol{x},\boldsymbol{\theta}) = g^* - f^1(\boldsymbol{x},\boldsymbol{\theta}). \tag{25}$$

Representing the left side of eq. (25) in terms of the variables that generate each transform, we can decompose the small affine transform between the reference and template. Using matrices $[\boldsymbol{R}]_\times, \boldsymbol{\Lambda}$ and $\boldsymbol{S}$, and coefficients $\gamma_i \in \{0,1\}$, $i = 1,2,\ldots,9$, we can represent the left side of eq. (25) as

$$\boldsymbol{x}^\top \left(\begin{pmatrix} 0 & \gamma_3 & \gamma_2 \\ \gamma_3 & 0 & \gamma_1 \\ \gamma_2 & \gamma_1 & 0 \end{pmatrix} \circ [\boldsymbol{R}]_\times^\top + \mathrm{diag}(\gamma_4,\gamma_5,\gamma_6)\boldsymbol{\Lambda} + \mathrm{diag}(\gamma_7,\gamma_8,\gamma_9)\boldsymbol{S}^\top \right) \nabla f^1, \tag{26}$$

where $\boldsymbol{A}\circ\boldsymbol{B}$ is the Hadamard product of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. Furthermore, setting

$$\alpha_1 = yf_z^1 - zf_y^1, \alpha_2 = zf_x^1 - xf_z^1, \alpha_3 = yf_x^1 - xf_y^1, \tag{27}$$

$$\alpha_4 = xf_x^1, \alpha_5 = yf_y^1, \alpha_6 = zf_z^1, \tag{28}$$

$$\alpha_7 = yf_y^1, \alpha_8 = zf_z^1, \alpha_9 = xf_x^1, \alpha_{10} = zf_y^1, \alpha_{11} = xf_z^1, \alpha_{12} = yf_z^1, \tag{29}$$

we rewrite eq. (25) as

$$\sum_{i=1}^{6} \gamma_i\alpha_i\phi_i + \sum_{j}^{3} (\alpha_{2(j-1)+7}\phi_{2(j-1)+7} + \alpha_{2(j-1)+8}\phi_{2(j-1)+8}) = g^* - f^1(\boldsymbol{x},\boldsymbol{\theta}). \tag{30}$$

Equation (30) contains 12 unknowns in a single equation. The sum of coefficients $\sum_{i=1}^{6}\gamma_i + \sum_{i=7}^{9} 2\gamma_i$ is greater or equal to one even though we have only one template. We adopt the surface integration for eq. (30) for this template image. Selecting different surfaces of a surface integration, we obtain more than one independent equation. For the centre $\boldsymbol{\mu} = (\mu,\mu,\mu)^\top$ of a template image and radius $\{r_i\}_{i=1}^n,\ r_i \neq r_j$, we define surface of a sphere as

$$\mathcal{S}_3(r) = \{\boldsymbol{x} \mid \|\boldsymbol{x}-\boldsymbol{\mu}\|_2 = r\}. \tag{31}$$

For the centre $\boldsymbol{\mu} = (\mu,\mu,\mu)^\top$ of a template image, radius r, rotation angles $\boldsymbol{\phi}_i = (\phi_{i1},\phi_{i2},\phi_{i3})^\top$ and vectors $\boldsymbol{p}_1 = (x-\mu, y-\mu, -\mu)^\top$, $\boldsymbol{p}_2 = (-\mu, y-\mu, z-\mu)^\top$ and $\boldsymbol{p}_3 = (x-\mu, -\mu, z-\mu)^\top$, we define the surface comprising three planes as

$$\mathcal{P}_3(r,\boldsymbol{\phi}) = \{\boldsymbol{\mu}+\boldsymbol{R}(\boldsymbol{\phi})\boldsymbol{p}_1, \boldsymbol{\mu}+\boldsymbol{R}(\boldsymbol{\phi})\boldsymbol{p}_2, \boldsymbol{\mu}+\boldsymbol{R}(\boldsymbol{\phi})\boldsymbol{p}_3 \mid \mu-r \leq x,y,z, \leq \mu+r\}. \tag{32}$$

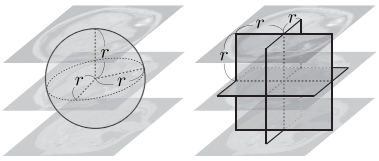


Fig. 2. Surfaces used for surface integration to obtain independent equations. (a) and (b) show surface $\mathcal{S}_3(r)$ of the sphere and surface $\mathcal{P}_3(r,0)$ comprising three planes. Integration of the volume gives a equation for an image. The integration of different surfaces, such as a different spheres and orthogonal square planes, gives several independent equations for an image.

For a set of radius $\{r_i\}_{i=1}^n$, we obtain sets of $\{\mathcal{S}_3(r_i)\}_{i=1}^n$ and $\{\mathcal{P}_3(r_i)\}_{i=1}^n$. We adopt $\{\mathcal{S}_3(r_i)\}_{i=1}^n$ and $\{\mathcal{P}_3(r_i)\}_{i=1}^n$ as surfaces $\{\Omega_i\}_{i=1}^n$ for the surface integration. Figure 2 shows the surfaces used for surface integration. For $\{\Omega_i\}_{i=1}^n$ and $\beta_{ij} = \int_{\Omega_i} \alpha_j(\boldsymbol{x}) d\boldsymbol{x}, j = 1, 2, \ldots, 12$, we set the coefficient vector

$$\boldsymbol{\chi}_i = (\beta_{i\,1}, \beta_{i\,2}, \ldots, \beta_{i\,12}). \tag{33}$$

Here, we have the relations $\chi_i \neq \chi_j$ and $h_i \neq h_j$ for $i \neq j$. Setting $n \geq \sum_{i=1}^6 \gamma_i + \sum_{i=7}^9 2\gamma_i$,

$$\begin{pmatrix} \beta_{1\,1} & \beta_{1\,2} & \cdots & \beta_{1\,12} \\ \beta_{2\,1} & \beta_{2\,2} & \cdots & \beta_{2\,12} \\ & & \vdots & \\ \beta_{n\,1} & \beta_{n\,2} & \cdots & \beta_{n\,12} \end{pmatrix} \boldsymbol{\zeta} = \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{pmatrix}. \tag{34}$$

where

$$\boldsymbol{\zeta} = (\gamma_1\phi_1, \gamma_2\phi_2, \ldots, \gamma_6\phi_6, \gamma_7\phi_7, \gamma_7\phi_8, \gamma_8\phi_9, \gamma_8\phi_{10}, \gamma_9\phi_{11}, \gamma_9\phi_{12})^\top, \tag{35}$$

and

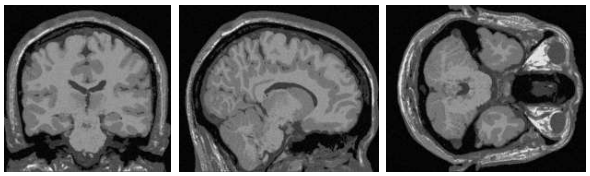
$$h_i = \int_{\mathcal{S}_i} (g^* - f^1) d\boldsymbol{x}, \tag{36}$$

then, we can estimate the transforms as a solution to the linear system of equations.

4 Numerical Examples

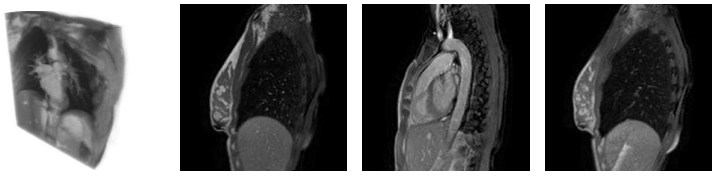
Three experiments evaluate the performance of our local linear method. The first and second experiments show the accuracy of estimation for a single transform and multiple transforms, respectively. The third experiment shows the robust estimation of templates with small pattern perturbations.

The first and second experiments use volumetric data obtained by MRI simulation of human brain [21]. Figure 3 shows slice images of the volumetric data.



(a) Coronal (b) Sagittal (c) Transverse

Fig. 3. Slice images extracted from volumetric data. (a)-(c) Slice images extracted from a voxel image obtained by MRI simulation of a human brain [21]. The size of the voxel image is 181×217×181 voxels. The slice images (a), (b) and (c) are extracted from the $z = 45$, $x = 90$ and $y = 100$ planes, respectively. In experiments, we embed the voxel image in a background image of 308×308×308 voxels. The intensities of the background images are 0.



(a) Image frame (b) Sagittal: $y = 9$ (c) Sagittal: $y = 29$ (d) Sagittal: $y = 41$

Fig. 4. Volumetric spatiotemporal MRI lung data [22]. (a) Voxel image of a frame of a sequence. (b)-(d) Sagittal slices of the frame. The spatial and time resolutions of the data are 50×224×224 and 200, respectively. The time between frames is 331 ms. In the experiments, we embed a volumetric image of a frame on a background image of 316×316×316 voxels. Each voxel value in the background image is 0.

Furthermore, for the first and second experiments, we generate smooth images from these slice images by linear filtering of the convolution with Gaussian kernel of standard deviation τ. For third experiment, we use volumetric spatiotemporal MRI lung data. [22]. Figure 4 shows a few frame of the volumetric spatiotemporal MRI lung data. In a sequence, the volumetric data gradually changes with the breathing of the patient. Table 1 summarises the parameters for the first and second experiments. Table 2 summarises the data for the third experiment.

Figure 5 shows the results of the first experiment for the estimation of rotation angle. In Figs. 5(a), (b) and (c), for displacements of less than 4 voxels, the estimation errors are smaller than 2.5 degrees if we use the surfaces $\{\mathcal{S}_3(r_i)\}_{i=1}^{10}$ for surface integration. Figures 5(d), (e) and (f) show that for displacements of less than 4 voxels, the estimation errors are smaller than 3.5 degrees if we use the surfaces $\{\mathcal{P}(r_i, \mathbf{0})\}_{i=1}^{10}$ for surface integration. In Figs. 5(g), (h), (i), (j), (k)

Table 1. Parameters for the first and second experiments using voxel image of human brain data.

$\boldsymbol{A}_\delta$	Pregeneration	Template	Filtering	Dimension	Lines
$\boldsymbol{R}_1$	$-60 < \phi_1 < 60$ step of 12	$1 < \phi_1 < 6$ step of 1	$0 < \tau < 5$ step of 0.1	$\hat{d} = 1024$	$\mathcal{S}_3(r)$ $10 < r < 110$ step of 10
$\boldsymbol{R}_2$	$-60 < \phi_2 < 60$ step of 12	$1 < \phi_2 < 6$ step of 1	$0 < \tau < 5$ step of 0.1	$\hat{d} = 1024$	$\mathcal{S}_3(r)$ $10 < r < 110$ step of 10
$\boldsymbol{R}_3$	$-60 < \phi_3 < 60$ step of 12	$1 < \phi_3 < 6$ step of 1	$0 < \tau < 5$ step of 0.1	$\hat{d} = 1024$	$\mathcal{S}_3(r)$ $10 < r < 110$ step of 10
$\boldsymbol{R}$	$-7 < \phi_1, \phi_2, \phi_3 < 7$ step of 7	$1 < \phi_1, \phi_2, \phi_3 < 3$ step of 1	$0 < \tau < 5$ step of 0.1	$\hat{d} = 1024$	$\mathcal{S}_3(r)$ $10 < r < 50$ step of 10

Table 2. Data for the the third experiment using the volumetric spatiotemporal data.

$\boldsymbol{A}_\delta$	Pregeneration with 22nd frame	Template with 22nd, 23rd, 24th and 34th frame	Filtering	Dimension	Lines
$\boldsymbol{R}_3$	$-60 < \phi_3 < 60$ step by 12	$1 < \phi_1 < 6$ step by 1	not used	$\hat{d} = 1024$	$\mathcal{S}_3(r)$ $10 < r < 20$ step by 5

and (l), for displacements of greater than 4 voxels in smooth images, our method estimates rotation angles with errors smaller than 1 degree.

The second experiment evaluates estimation errors for multiple transforms. Figure 6 shows the results of the second evaluation. In Fig. 6, the results show that the estimation of multiple transforms is unstable. Furthermore, the estimation errors are larger than 1 degree even for small displacements of one voxel. However, for smoothed images, the mean estimation error of the multiple transforms is about 1.5 degrees for the three rotation axes.

The third experiment evaluates the accuracy and robustness of estimation of rotation for a template with a small pattern perturbation. Figures 7(a) and (b) show the differences between the 22nd frame and the 23rd-200th frames of the four-dimensional data. Figure 7(c) shows the results of the estimation. In Fig. 7(c), curves represent absolute values of the estimation error plotted against the displacement caused by the rotation. For differences from $-\infty$ to -6.94 dB, the estimation errors are smaller than 1.5 degrees. Table 3 shows the difference between a template and generated g^*. In Table 3, the distance between the generated g^* and template is smaller than one between template and its nearest neighbour.

Table 4 summarises the accuracy, the number of pregenerated images and the dimension of the search space in Figs. 5 and 6. The results in Fig. 5 imply that integration with the surfaces $\{\mathcal{S}_3(r_i)\}_{i=1}^{10}$ leads to more accurate and stable estimation than integration with the surfaces $\{\mathcal{P}(r_i, \mathbf{0})\}_{i=1}^{10}$ for the case of a rotation. For the estimation of a single transform, our method requires 16.7% of the number of pregenerated images of naive NNS. Furthermore, for the estimation of multiple transforms, our method requires 2.1% of the number of pregenerated images of the naive NNS. Moreover, our method reduces size of search space to 4.0×10^{-3}% . For both the estimations, the dimension of the search space is 3.5×10^{-3}% of the original dimension of the images. Moreover, the results

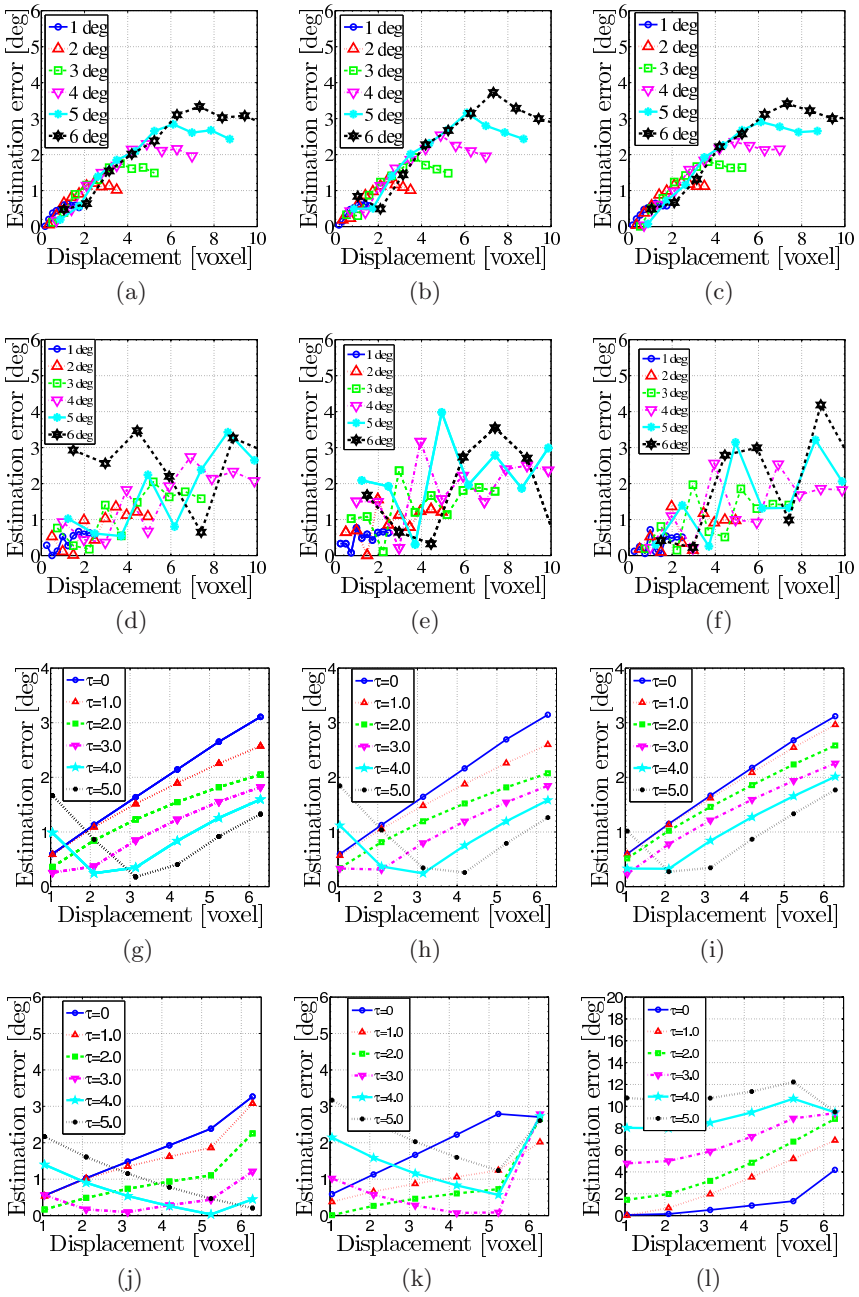
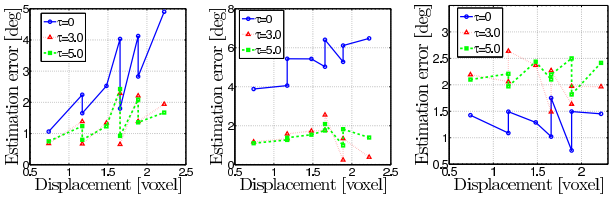


Fig. 5. Accuracy of estimation for a spatial rotation. We estimate the rotation angles ϕ_1, ϕ_2 and ϕ_3 independently. The first, second and third columns represent the accuracy of estimation for rotation around the x, y and z axes, respectively. (a) and (d), (b) and (e), and (c) and (f) show the accuracy of estimation without Gaussian filtering. (g) and (j), (h) and (k), and (i) and (l) show the accuracy of estimation for smooth images, for the rotation around x, y and z axes, respectively. In the first and third rows and the second and fourth rows, we adopt $\mathcal{S}_3(r)$ and $\mathcal{P}_3(r, \boldsymbol{\phi})$ as the surfaces for the surface integration, respectively. Displacements are given by $r\phi_1, r\phi_2$ and $r\phi_3$.



(a) Rotation around x (b) Rotation around y (c) Rotation around z

Fig. 6. Accuracy of estimation for multiple transforms. For the estimation, we adopt combinations of three rotations around the x, y and z axes. The left, middle and right graphs show the results of estimation for rotation around the x, y and z axes with Gaussian filtering with standard deviation τ, respectively. For the surface integration, we adopt surfaces $\{\mathcal{P}_i^1\}_{i=1}^n$. For the rotations around the x, y and z axes, the displacements are given by $r\sqrt{\phi_3^2+\phi_2^2}$, $r\sqrt{\phi_3^2+\phi_1^2}$ and $r\sqrt{\phi_1^2+\phi_2^2}$ with radius r in the surface integral, respectively.

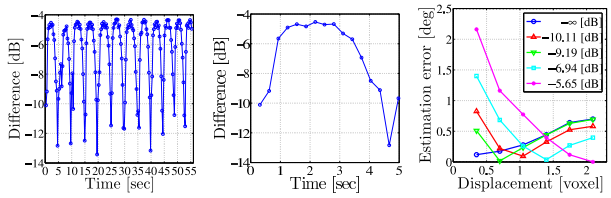
of third experiment show that our method estimate transform for the template image with small pattern perturbation.

Table 3. Evaluation of approximation for generated new entries. We generate new entries for rotated images with small pattern perturbation. For a generation of a new entry, we use 4-neighbours of a template. As templates, we use rotated images of the 22nd, 23rd, 24th and 25th frame of data with angle ϕ_3. For a template g, we first compute the difference between g and its nearest neighbour in pregenerated images as $10\log_{10}\left(\|f^1-g\|_2/\|g\|_2\right)$. Second, we compute the difference between g and a generated new entry g^* as $10\log_{10}\left(\|g^*-g\|_2/\|g\|_2\right)$. In this Table, the columns for the nearest neighbour (NN) and the local linear method (LLM) show the difference between f^1 and g and between g^* and g, respectively.

angle [degree]	Difference between f^1 and g[dB]									
	22nd		23rd		24th		25th		34th	
ϕ_3	NN	LLM	NN	LLM	NN	LLM	NN	LLM	NN	LLM
2	-4.77	-5.15	-4.85	-5.15	-4.78	-5.06	-4.39	-4.70	-4.60	-4.90
4	-3.19	-3.95	-3.31	-3.96	-3.26	-3.91	-3.21	-3.81	-3.25	-3.87
6	-2.57	-3.71	-2.70	-3.72	-2.66	-3.68	-2.69	-3.62	-2.68	-3.65

5 Conclusions

For volumetric images, we first defined the local linear property of the image manifold for a small geometrical perturbation. We then introduced an algorithm based on the local linear property for three-dimensional affine image registration to reduce the time and spatial complexity of computation. The algorithm first generates a new image for a template using a small number of images preproduced from the reference image. Second, using the new image, the proposed



(a) (b) (c)

Fig. 7. Estimation for a template with small pattern perturbation. (a) Difference between 22nd frame and 23rd-200th frames of four-dimensional MRI lung data. (b) Scaled-up graph of (a) showing difference between 22nd frame and 23rd-38th frames. (c) Accuracy of estimation for rotation angle ϕ_3 around the z axis. The differences between the 22nd frame and the 22nd, 23rd, 24th, 25th and 34th frames are $-\infty, -10.11, -9.19, -6.94$ and -5.65 [dB], respectively. For surface integration, we adopt the surface $\mathcal{S}_3(r)$. The displacement is given by $r\phi_3$.

Table 4. Accuracy and compression ratio for volumetric data obtain by MRI simulation of human brain. First column shows given accuracy in the estimation. Second column shows necessary step sizes in pregeneration, which give the accuracy in first column, for the nearest neighbour search (NNS) and the local linear method. Third column shows dimensions of search space for NNS and LLM. Fourth column illustrates compression ration of the LLM compared with the NNS.

$\boldsymbol{A}_\delta$	Accuracy	Necessary step size		Dimension		Compression ratio
		NNS	LLM	Original	Search space	in pregeneration
$\boldsymbol{R}_1$	1 [degree]	2 [degree]	12 [degree]	29218112	1024	16.7 [%]
$\boldsymbol{R}_2$	1 [degree]	2 [degree]	12 [degree]	29218112	1024	16.7 [%]
$\boldsymbol{R}_3$	1 [degree]	2 [degree]	12 [degree]	29218112	1024	16.7 [%]
$\boldsymbol{R}$	1.5 [degree]	3 [degree]	12 [degree]			
	2.0 [degree]	4 [degree]	12 [degree]	29218112	1024	2.1 [%]
	1.5 [degree]	3 [degree]	12 [degree]			

method finds a small affine transform between the new image and the best matching image in the dictionary. Finally, our method estimates transforms using the new image and its neighbours. This algorithm reduces the computational cost of preprocessing and the size of the images used in the nearest-neighbour search. In the numerical examples, using volumetric data of the human brain obtained by simulated MRI, we show that our method can accurately estimate single and multiple transforms using a small number of pregenerated images. Furthermore, using four-dimensional MRI lung data, we show that our method can robustly estimate single transform for a template with small pattern perturbation.

This research was supported by the "Computational Anatomy for Computer-Aided Diagnosis and Therapy: Frontiers of Medical Image Sciences" and "Multidisciplinary Computational Anatomy and Its Application to Highly Intelligent Diagnosis and Therapy" projects funded by a Grant-in-Aid for Scientific Research on Innovative Areas from MEXT, Japan, and by Grants-in-Aid for Scientific Research funded by the Japan Society for the Promotion of Science.

References

1. Nishino, K., Ikeuchi, K.: Robust simultaneous registration of multiple range images. In: Digitally Archiving Cultural Objects. Springer (2008) 71–88
2. Salvi, J., Matabosch, C., Fofi, D., Forest, J.: A review of recent range image registration methods with accuracy evaluation. Image and Vision Computing **25** (2007) 578–596
3. Besl, P., McKay, N.D.: A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence **14** (1992) 239–256
4. Daniel, F.H., Hebert, M.: Fully automatic registration of multiple 3D data sets. Image and Vision Computing **21** (2003) 637–650
5. Markelj, P., Tomaževič, D., Likar, B., Pernuŝ, F.: A review of 3D/2D registration methods for image-guided interventions. Medical Image Analysis **16** (2012) 642–661
6. Klein, A., Andersson, J., Ardekani, B.A., Ashburner, J., Avants, B.B., Chiang, M.C., Christensen, G.E., Collins, D.L., Gee, J.C., Hellier, P., Song, J.H., Jenkinson, M., Lepage, C., Rueckert, D., Thompson, P.M., Vercauteren, T., Woods, R.P., Mann, J.J., Parsey, R.V.: Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. NeuroImage **46** (2009) 786–802
7. Capekm M: Optimisation strategies applied to global similarity based image registration methods. Proc. the 7th International Congerence in Central Europoe on Computer Graphic (1999) 369–374
8. Itoh, H., Lu, S., Sakai, T., Imiya, A.: Global image registration by fast random projection. In Proc. International Symposium on Visual Computing (2011) 23–32
9. Itoh, H., Lu, S., Sakai, T., Imiya, A.: Interpolation of reference images in sparse dictionary for global image registration. In Proc. the 8th International Symposium on Visual Computing (2012) 657–667
10. Itoh, H., Sakai, T., Kawamoto, K., Imiya, A.: Global image registration using random projection and local linear method. In Proc. Internatinonal Conference on Computer Analysis of Images and Patterns (2013) 564–571
11. Cock, K.D., Moor, B.D.: Subspace angles between ARMA models. Systems & Control Letters **46** (2002) 265–270
12. Hamm, J., Lee, D.D.: Grassmann discriminant analysis: A unifying view on subspace-based learning. In Proc. International Conference on Machine Learning (2008) 376–383
13. Altman, N.S.: An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician (1992) 175–185
14. Vempala, S.S.: *The Random Projection Method*. Volume 65. American Mathematical Society (2004)
15. Arya, S., Mount, D.M., Netanyahu, N.S., Silverman, R., Wu, A.Y.: An optimal algorithm for approximate nearest neighbor searching in fixed dimensions. In Proc. ACM-SIAM Symposium on Discrete Algorithms (1994) 573–582
16. Baraniuk, R.G., Wakin, M.B.: Random projections of smooth manifolds. Foundations of Computational Mathematics **9** (2009) 51–77
17. Bingham, E., Mannila, H.: Random projection in dimensionality reduction: Applications to image and text data. In Proc. International Conference on Knowledge Discovery and Data Mining (2001) 245–250
18. Johnson, W., Lindenstrauss, J.: Extensions of Lipschitz maps into a Hilbert space. Contemporary Mathematics **26** (1984) 189–206

19. Frankl, P., Maehara, H.: The Johnson-Lindenstrauss lemma and the sphericity of some graphs. Combinatorial Theory, Series B **44** (1988) 355–362
20. Sakai, T., Imiya, A.: Practical algorithms of spectral clustering: Toward large-scale vision-based motion analysis. In: Machine Learning for Vision-Based Motion Analysis. Springer (2011) 3–26
21. Cocosco, C., Kollokian, V., Kwan, R.S., Evans, A.: Brainweb. Online interface to a 3D MRI simulated brain database. NeuroImage **5** (1997) 425
22. Boye, D., Samei, G., Schmidt, J., Székely, G., Tanner, C.: Population based modeling of respiratory lung motion and prediction from partial information. In Proc. SPIE 8669, Medical Imaging 2013: Image Processing **8669** (2013)