Volume Registration using the 3-D Pseudo-polar FFT

Y. Keller[1], Y. Shkolnisky[2], A. Averbuch[2]

[1]Mathematics Department

Yale University, Connecticut, USA

Phone: +1-203-432-4345

Email: yosi.keller@yale.edu

[2]School of Computer Science

Tel Aviv University, Tel Aviv 69978, Israel

Abstract

This paper introduces an algorithm for the registration of rotated and translated volumes, which operates in the frequency domain. The Fourier domain allows to compute the rotation and translation parameters separately, thus reducing a problem with six degrees of freedom to two problems of three degrees of freedom each. We propose a three-step procedure. The first step estimates the rotation axis. The second computes the planar rotation relative to the rotation axis, and the third recovers the translational displacement. The rotation estimation is based on Euler's theorem, which allows to represent a rotation using only three parameters. Two parameters represent the rotation axis and the third represents the planar rotation perpendicular to the axis. The 3-D pseudo-polar FFT is used to accurately estimate the rotation axis and a variant of the angular difference function image registration scheme is derived to estimate the planar rotation around the axis. Experimental results show that the algorithm is accurate and robust to noise.

*This work was supported by a grant from the Ministry of Science, Israel.

1 Introduction

Rigid volume registration is a major component in 3-D object modeling in a diverse range of applications. Examples for such applications are the assembly of 3-D models from complementary patches [1, 2, 3], range imaging [4] and bio-informatics [5, 6]. Let $V_1(\vec{x})$ and $V_2(\vec{x})$, $\vec{x} = (x, y, z)$, be two partially overlapping volumes, that are related by a 3-D rigid transformation

$$V_1(\vec{x}) = V_2(R\vec{x} + \Delta\vec{x}), \tag{1.1}$$

where R is a 3-D rotation matrix and $\Delta\vec{x} = (\Delta x, \Delta y, \Delta z) \in \mathbb{R}^3$. The goal of the registration process is to estimate R and $\Delta\vec{x}$.

There are several approaches towards 3-D registration, which can be categorized as either feature or intensity based. Feature based approaches [2, 7, 8, 9] detect a set of features in the registered volumes, and align the volumes by using the coordinates of these features. Intensity based schemes [5, 10] align the volumes by using the intensities of the voxels.

A broad family of feature based algorithms is based on the ICP (iterative closest point) algorithm and its modifications. The ICP algorithm, first presented in [7], is based on finding the transformation that minimizes the Euclidean distance between pairs of corresponding points on the two surfaces. This is an iterative approach that at each iteration computes points correspondence based on the transformation found at the previous iteration, and then, finds the transformation that best maps points on one surface to their corresponding points on the other surface. The method assumes that the two surfaces are roughly aligned and therefore, each point on one surface corresponds to its closest point on the other surface. [7] proves that this method always converges to a local minimum. Thus, to obtain the correct registration, which corresponds to the global minimum, the ICP algorithm requires an initial guess that is close to the correct solution.

Intensity based algorithms include optimization and Fourier based schemes. Optimization based schemes formulate the registration problem as the optimization of some cost function, such as the L_2 norm [11], and then, use a general purpose optimization algorithm to optimize the cost function. [11] extends the widely used gradient methods to 3-D image registration. It uses Newton's method to minimize the L_2 norm of the

intensity differences as a function of motion parameters. Due to the properties of non-linear optimization, these algorithms are unable to estimate large motions. To estimate large motions, such schemes are used in conjunction with a bootstrapping method, which computes a pre-alignment that is close to the optimum.

Frequency domain methods are of particular interest to the current paper. These methods use the properties of the Fourier transform to separately estimate rotation and translation. This reduces a problem with six degrees of freedom into two problems with three degrees of freedom (see [10, 12] as examples for these methods). The algorithm in [10] consists of three steps. The first recovers the rotation axis, the second recovers the rotation angle, and the third recovers the translation parameters. To recover the rotation parameters, the algorithm normalizes the Fourier transform of the input objects and integrates it in the radial direction. The direction in which this integral is minimal is the direction of the rotation axis. The polar Fourier coefficients are interpolated from the Cartesian Fourier coefficients. As this task is computationally intensive, the entire set of polar coefficients can not be computed directly. Thus, simulated annealing is used to find the minimum of the integral. Simulated annealing might not converge to the global minimum and performance of such algorithms in terms of speed and accuracy cannot be estimated. This renders such approaches prohibitive for time critical applications, or applications where the accuracy must be predictable. Furthermore, the integration in the radial direction suffers from inaccuracies caused by discretization and interpolation errors and therefore, such schemes cannot achieve high accuracy. The planar rotation is recovered using a scheme similar to [13], where again, interpolating the Fourier coefficients results in inaccuracies and high computational complexity. Once the rotation parameters are recovered, the translation is recovered using phase correlation [14].

Another frequency domain approach is given in [12]. This algorithm recovers the rotation parameters by formulating the problem as a linear system, whose entries are computed by the frequency domain relations of the two objects. As before, the translation parameters are recovered by using phase correlation. As in [10], the algorithm requires integration in the radial direction, which incurs inaccuracies. The proposed method in this paper outperforms the method in [10].

[5, 6] align density volumes for Protein-Protein docking by computing the polar Fourier transform of the density volumes. Rotations are reduced to translations in the spherical coordinate system, which are

recovered by using phase correlation [14]. Again, the interpolation of the polar Fourier coefficients results in inaccuracies and high computational complexity

In this work we extend the preliminary results given in [15]. We present a Fourier based approach, which does not require interpolation in the frequency domain. It is based on the 2-D pseudo-polar FFT (PPFT2D) [16] and 3-D pseudo-polar FFT (PPFT3D) [17], which compute the Discrete Fourier Transform (DFT) on non-Cartesian grids. This allows a fast and algebraically accurate registration, which draws on Euler's theorem to estimate the 3-D rotation. The algorithm has three steps. First, the PPFT3D is used to recover the rotation axis. Then, the rotation around the axis is estimated using a pseudo-cylindrical representation computed with the PPFT2D. Finally, the translation is computed by using phase correlation [14].

Our approach tackles some fundamental issues, which were previously overlooked, especially in [10, 12]. We provide an algorithm that is both efficient and mathematically rigorous. The scheme efficiently and accurately computes the radial integration, and hence, the execution time and accuracy of the algorithm are predictable and depend only on the size of the input volumes. Thus, the accuracy and the error are controlled by the algorithm. In particular, it does not use general purpose optimization techniques, whose performance depend on the content of the input volumes. Therefore, the complexity of the algorithm is of the same order as the 3-D FFT. The second step in our scheme is based on an extension of the image registration scheme given in [18]. We extend it to handle cylindrical geometry, that is, simultaneous 2-D rotation of a set of planes around a common axis. Unlike [10] and [13], it is fast, non-iterative, does not use interpolation, guarantees convergence in a predictable time regardless of the volume's content, and can produce a controlled registration error.

The proposed scheme accurately estimates arbitrary large rotations without applying a general purpose optimization scheme (gradient based, simulated annealing, etc.). The only "optimization" required is finding the minimal element in an array. It is fast, robust to noise, and the registration accuracy can be increased arbitrarily. The implementation requires only 1-D operations and is appropriate for real-time implementations.

The paper is organized as follows. Sections 2 and 3 present the PPFT3D and apply Euler's theorem to 3-D rotations, respectively. Planar rotations are recovered in Section 4 while Section 5 describes the estimation

of translations. Experimental results and concluding remarks are given in Sections 6 and 7, respectively.

2 The 3-D pseudo-polar FFT

Given a volume I of size $N \times N \times N$, its 3-D Fourier transform, denoted $\hat{I}(\omega_x, \omega_y, \omega_z)$, is given by

$$\hat{I}(\omega_x, \omega_y, \omega_z) = \sum_{u,v,w=-N/2}^{N/2-1} I(u,v,w) e^{-\frac{2\pi\imath}{M}(u\omega_x + v\omega_y + w\omega_z)}, \quad \omega_x, \omega_y, \omega_z \in \mathbb{R}. \tag{2.1}$$

We assume for simplicity that I has equal dimensions in the x, y, and z directions and N is even. For ω_x, ω_y, and ω_z that are sampled on the Cartesian grid $(\omega_x, \omega_y, \omega_z) = (m,k,l)$, $m,k,l = -\frac{M}{2}, \ldots, \frac{M}{2} - 1$, the Fourier transform in Eq. 2.1 has the form

$$\hat{I}_{Cart}(m,k,l) \triangleq \hat{I}(m,k,l) = \sum_{u,v,w=-N/2}^{N/2-1} I(u,v,w) e^{-\frac{2\pi\imath}{M}(um+vk+wl)}, \tag{2.2}$$

where $m,k,l = -\frac{M}{2}, \ldots, \frac{M}{2} - 1$, which is usually referred to as the 3-D DFT of the volume I. The parameter M ($M \geq N$) sets the frequency resolution of the DFT. It is well-known that the DFT of I, given by Eq. 2.2, can be computed in $O(M^3 \log M)$ operations.

For some applications it is desirable to compute the Fourier transform of I in spherical coordinates. Formally, we want to sample the Fourier transform in Eq. 2.1 on the grid $(\omega_x, \omega_y, \omega_z)$ where

$$\begin{gathered} \omega_x = r_m \cos\theta_k \sin\phi_l, \quad \omega_y = r_m \sin\theta_k \sin\phi_l, \quad \omega_z = r_m \cos\phi_l, \\ r_m = m, \quad \theta_k = 2\pi k/K, \quad \phi_l = \pi l/L, \\ m = 0, \ldots, M-1, \quad k = 0, \ldots, K-1, \quad l = 0, \ldots, L-1. \end{gathered} \tag{2.3}$$

The Fourier transform of I in spherical coordinates has the form

$$\hat{I}_{sph}(m,k,l) \triangleq \sum_{u,v,w=-N/2}^{N/2-1} I(u,v,w) e^{-\frac{2\pi\imath m}{M}(u\cos\theta_k \sin\phi_l + v\sin\theta_k \sin\phi_l + w\cos\phi_l)}. \tag{2.4}$$

The spherical grid in Eq. 2.3 is equally spaced in both the radial and angular directions

$$\Delta r = r_{m+1} - r_m = 1, \tag{2.5}$$

$$\Delta\theta = \theta_{k+1} - \theta_k = \frac{2\pi}{K}, \tag{2.6}$$

$$\Delta\phi = \phi_{l+1} - \phi_l = \frac{\pi}{L}. \tag{2.7}$$

Unfortunately, there is no fast algorithm for computing the Fourier transform of a volume in spherical coordinates.

The 3-D pseudo-polar Fourier transform (PPFT3D) [17] evaluates the 3-D Fourier transform of a volume on the 3-D pseudo-polar grid, which approximates the 3-D spherical grid, given in Eq. 2.3. Formally, the 3-D pseudo-polar grid is given by the set of samples

$$P \triangleq P_1 \cup P_2 \cup P_3, \tag{2.8}$$

where

$$P_1 \triangleq \{(m, -\frac{2k}{N}m, -\frac{2l}{N}m)\}, \tag{2.9}$$

$$P_2 \triangleq \{(-\frac{2k}{N}m, m, -\frac{2l}{N}m)\}, \tag{2.10}$$

$$P_3 \triangleq \{(-\frac{2k}{N}m, -\frac{2l}{N}m, m)\}, \tag{2.11}$$

and $k, l = -\frac{N}{2}, \ldots, \frac{N}{2}$, $m = -\frac{3N}{2}, \ldots, \frac{3N}{2}$. See Figs. 1(a), 1(b), and 1(c) for an illustration of the sets P_1, P_2, and P_3, respectively. We define the 3-D pseudo-polar Fourier transform of I as the samples of the Fourier transform $\hat{I}$, given in Eq. 2.1, on the 3-D pseudo-polar grid P, given by Eqs. 2.8 – 2.11. Formally, the 3-D pseudo-polar Fourier transform, denoted by $\hat{I}^s_{PP}$ ($s = 1, 2, 3$), is a linear transformation, which is defined for $m = -\frac{3N}{2}, \ldots, \frac{3N}{2}$ and $k, l = -\frac{N}{2}, \ldots, \frac{N}{2}$, as

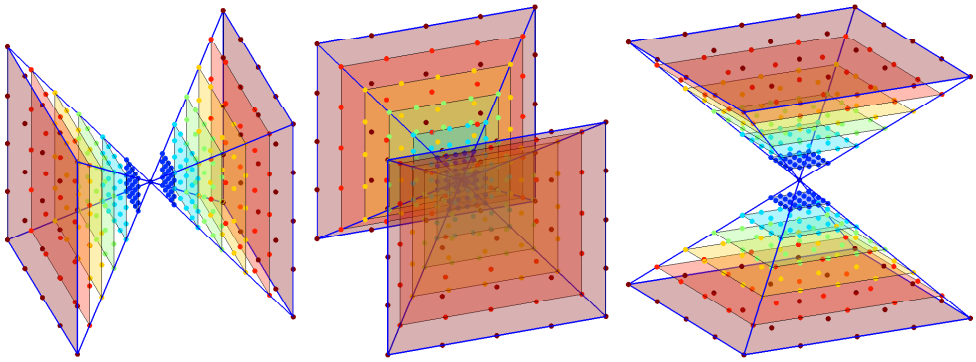
$$\hat{I}^1_{PP}(m,k,l) \triangleq \widehat{I}(m, -\frac{2k}{N}m, -\frac{2l}{N}m) = \sum_{u,v,w=-N/2}^{N/2-1} I(u,v,w) e^{-\frac{2\pi\imath}{M}(mu - \frac{2k}{N}mv - \frac{2l}{N}mw)}, \tag{2.12}$$

$$\hat{I}^2_{PP}(m,k,l) \triangleq \widehat{I}(-\frac{2k}{N}m, m, -\frac{2l}{N}m) = \sum_{u,v,w=-N/2}^{N/2-1} I(u,v,w) e^{-\frac{2\pi\imath}{M}(-\frac{2k}{N}mu + mv - \frac{2l}{N}mw)}, \tag{2.13}$$

$$\hat{I}^3_{PP}(m,k,l) \triangleq \widehat{I}(-\frac{2k}{N}m, -\frac{2l}{N}m, m) = \sum_{u,v,w=-N/2}^{N/2-1} I(u,v,w) e^{-\frac{2\pi\imath}{M}(-\frac{2k}{N}mu - \frac{2l}{N}mv + mw)}, \tag{2.14}$$

where $\hat{I}$ is given by Eq. 2.1.

As we can see from Figs. 1(a)–1(c), for fixed angles k and l, the samples of the 3-D pseudo-polar grid are equally spaced in the radial direction. However, this spacing is different for each angle. Also, the grid is not equally spaced in the angular direction, but has equally spaced slopes.



(a) 3-D pseudo-polar sector P_1 (b) 3-D pseudo-polar sector P_2 (c) 3-D pseudo-polar sector P_3

Figure 1: The 3-D pseudo-polar grid

The set P, given by Eq. 2.8, can be written in polar coordinates as

$$P = \{(r\cos\theta\sin\phi, r\sin\theta\sin\phi, r\cos\phi) \mid (r,\theta,\phi) \in \Gamma\}, \tag{2.15}$$

where the set Γ contains all triplets that correspond to points on the pseudo-polar grid P.

Two important properties of the 3-D pseudo-polar Fourier transform are that it is invertible and that both the forward and inverse pseudo-polar Fourier transforms can be implemented using fast algorithms. Moreover, the implementations require only 1-D equispaced FFT's. In particular, the algorithms do not require re-gridding or interpolation.

The algorithm for computing the 3-D pseudo-polar Fourier transform is based on the fractional Fourier transform. The fractional Fourier transform [19], with its generalization given by the chirp z-transform [20], is an algorithm that evaluates the Fourier transform of a sequence X on any equally spaced set of N points on the unit circle. Specifically, given a vector X of length N, $X = (X(u),\ u = -N/2, \ldots, N/2-1)$, and an arbitrary $\alpha \in \mathbb{R}$, the fractional Fourier transform is defined as

$$(\mathcal{F}^{\alpha}X)(k) = \sum_{u=-N/2}^{N/2-1} X(u)e^{-2\pi\imath\alpha ku/N}, \quad k = -N/2, \ldots, N/2. \tag{2.16}$$

The fractional Fourier transform samples the spectrum of X at the frequencies

$$\omega_k = \alpha k, \qquad k = -N/2, \ldots, N/2, \tag{2.17}$$

and its complexity for a given vector X of length N and an arbitrary $\alpha \in \mathbb{R}$ is $O(N\log N)$ operations.

The algorithm for computing the 3-D pseudo-polar Fourier transform samples the Fourier transform of a volume I on the pseudo-polar grid, with arbitrary frequency resolution in the radial and angular directions. The algorithm we present uses frequency resolution of $3N+1$ in the radial direction and $N+1$ in the angular directions. Denote,

- E– Zero padding operator that accepts a volume I of size $N \times N \times N$ and zero pads it to size $(3N+1) \times N \times N$ (along the x direction).
- $\mathcal{F}_1$ – 1-D DFT.
- $\mathcal{F}_3$ – 3-D DFT.
- $\mathcal{F}^\alpha$ – Fractional Fourier transform with factor α. The operator $\mathcal{F}^\alpha$ accepts a sequence of length N, pads it symmetrically to length $3N+1$, applies to it the fractional Fourier transform with factor α, and returns the $N+1$ central elements.
- $G_{k,n} \triangleq \mathcal{F}^{2k/n} \circ \mathcal{F}_1^{-1}$.

The algorithm for computing $\hat{I}^1_{PP}$, given by Eq. 2.12, is then

1. Let $\widehat{I}_d \leftarrow \mathcal{F}_3(E(I))$.
2. For each m and k set $U \leftarrow \widehat{I}_d(m,k,\cdot)$ and compute $T_1(m,k,\cdot) \leftarrow G_{k,n}(U)$.
3. For each m and l set $V \leftarrow T_1(m,\cdot,l)$ and compute $T_1'(m,\cdot,l) \leftarrow G_{k,n}(V)$.
4. For each m, k, l set $\hat{I}^1_{PP}(m,k,l) \leftarrow T_1'(m,-k,-l)$.

The algorithm for computing $\hat{I}^2_{PP}$ and $\hat{I}^3_{PP}$, given by Eqs. 2.13 and 2.14, is similar. The complexity of the algorithm for computing $\hat{I}^1_{PP}$ is $O(N^3 \log N)$. Since the complexity of computing $\hat{I}^2_{PP}$ and $\hat{I}^3_{PP}$ is also $O(N^3 \log N)$, the total complexity of computing the 3-D pseudo-polar Fourier transform is $O(N^3 \log N)$.

3 Euler's theorem and 3-D rotation estimation

Rotations in a 3-D Cartesian coordinate system may be represented by various formulations. In this paper we adopt the Euler angles representation [21], where 3-D rotations are expressed using three angles (α, β, γ), where α and β specify the direction of the rotation axis and γ specifies the angle of rotation in the plane perpendicular to the rotation axis. We denote the rotation axis by $\vec{n} \triangleq (n_x, n_y, n_z)$.

Euler's rotation theorem An arbitrary 3-D rotation can be expressed as a rotation by an angle γ around an axis given by a unit vector $\vec{n} = (n_x, n_y, n_z)$.

The rotation matrix R is given by

$$R = I\cos\gamma + (1-\cos\gamma)\begin{bmatrix} n_x^2 & n_x n_y & n_x n_z \\ n_y n_x & n_y^2 & n_y n_z \\ n_z n_x & n_z n_y & n_z^2 \end{bmatrix} + \sin\gamma \begin{bmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ n_y & n_x & 0 \end{bmatrix} \tag{3.1}$$

(see [21]), where I is the identity matrix. This representation of R is not unique. The same rotation can also be obtained by the rotation $(-\gamma)$ around the axis $(-\vec{n})$ [21]. Both γ and $\vec{n}$ can be easily recovered from the rotation matrix R. The three eigenvalues of R are $\lambda_1 = 1$ and $\lambda_{2,3} = e^{\pm\imath\gamma}$. The rotation axis $\vec{n}$ can be computed as $\vec{n} = \vec{v}/|\vec{v}|$, where $\vec{v}$ is the eigenvector that corresponds to λ_1, and γ can be recovered from $\lambda_{2,3}$. Any point on the rotation axis $\vec{n}$ is invariant under R as it is also an eigenvector of R. The rotation axis $\vec{n}$ can be recovered by finding the vector where the difference between the volume and its rotated replica is minimal. Given volumes V_1 and V_2, where $V_1(x) = V_2(Rx)$, we identify the rotation matrix R by the angles (α, β, γ). Then, the rotation axis can be recovered by computing

$$\Delta V(\theta,\phi) = \int_0^\infty |V_1(r,\theta,\phi) - V_2(r,\theta,\phi)|\, dr, \tag{3.2}$$

where (r, θ, ϕ) are spherical coordinates, and the rotation axis (α, β) is given by

$$(\alpha,\beta) = \arg\min_{\theta,\phi} \Delta V(\theta,\phi). \tag{3.3}$$

For non-centered rotations, where the volumes V_1 and V_2 are translated and rotated, Eq. 3.2 is applied to the magnitudes of the Fourier transforms of V_1 and V_2, denoted M_1 and M_2, respectively. Based on the phase

shift property of the Fourier transform [22], the magnitudes M_1 and M_2 are related by a three-dimensional rotation

$$M_1(R_n\vec{x}) = M_2(R_{z,\gamma}R_n\vec{x}), \tag{3.4}$$

where R_n is a 3-D rotation which aligns $\vec{n}$ with the z axis, and $R_{z,\gamma}$ is a rotation of angle γ around the z axis. Thus, given the volumes V_1 and V_2, where V_1 is a rotated and translated replica of V_2, we register V_1 and V_2 as follows:

1. Let M_1 and M_2 be the magnitudes of the Fourier transform of V_1 and V_2, respectively,

$$M_1 = |\mathcal{F}(V_1)|, \qquad M_2 = |\mathcal{F}(V_2)|, \tag{3.5}$$

   where the modulus is taken element-wise.

2. The rotation axis $\vec{n}$ is recovered by computing $\Delta V(\theta, \phi)$, given by Eq. 3.2, and locating its minimum (α, β), which corresponds to the Euler angles that define the rotation axis $\vec{n}$.

3. Given the rotation axis $\vec{n}$, we denote by $\tilde{R}$ the rotation that aligns $\vec{n}$ with the z axis. Denote

$$\tilde{M}_1 = \mathcal{F}(V_1(\tilde{R}\vec{x})), \qquad \tilde{M}_2 = \mathcal{F}(V_2(\tilde{R}\vec{x})). \tag{3.6}$$

   $\tilde{M}_1$ and $\tilde{M}_2$ are related by a planar rotation of angle γ around the z axis, which can be recovered by the cylindrical motion estimation scheme described in Section 4.

4. Given the rotation parameters (α, β, γ), the 3-D rotation matrix R is computed using Eq. 3.1 and it is applied to V_2. $V_1(\vec{x})$ and $V_2(R\vec{x})$ are related by a 3-D translation, which is recovered by using phase correlation (Section 5).

We propose a fast and algebraically accurate scheme for the computation of ΔV in step #2, which is based on the 3-D pseudo-polar Fourier transform (PPFT3D) [17] presented in Section 2. An important property of $\Delta V(\theta, \phi)$, given by Eq. 3.2, is that it can be discretized using very general sampling grids with respect to θ and ϕ. Specifically, the discretization of ΔV, denoted by ΔV^d, does not require a uniform spherical representation of the Fourier transforms of V_1 and V_2. All that is required is a grid on which we can

efficiently evaluate the Fourier transform of a given volume, and whose samples lie along rays. The PPFT3D provides such a grid, with uniform radial sampling along each ray. Thus, the algorithm for computing ΔV^d is

1. Compute M_1^d and M_2^d

$$M_1^d = \left|\mathcal{F}_{PP}(V_1)\right|, \qquad M_2^d = \left|\mathcal{F}_{PP}(V_2)\right|, \tag{3.7}$$

   where $\mathcal{F}_{PP}$ is the 3-D pseudo-polar Fourier transform defined in Section 2.

2. Evaluate Eq. 3.2 by

$$\Delta V^d\left(\theta_i, \phi_j\right) = \sum_{0 \le r_k \le M} \left| M_1^d\left(r_k, \theta_i, \phi_j\right) - M_2^d\left(r_k, \theta_i, \phi_j\right) \right| \Delta r_{i,j}, \tag{3.8}$$

   where M_1^d and M_2^d are given by Eq. 3.7, M is the radial resolution of the pseudo-polar grid, and $\Delta r_{i,j}$ is the radial sampling interval of the 3-D pseudo-polar grid for the ray whose direction is specified by θ_i and ϕ_j. Equation 3.8 is evaluated for all θ_i and ϕ_j such that $(r, \theta_i, \phi_j) \in \Gamma$ for some r, where Γ is given by Eq. 2.15. In other words, Eq. 3.8 is evaluated for all directions of the 3-D pseudo-polar grid. Different rays in the 3-D pseudo-polar grid have a different sampling interval $\Delta r_{i,j}$.

Equation 3.8 uses only samples of the 3-D pseudo-polar Fourier transform that lie within a sphere of radius $M/2$. This summation ignores samples whose radius is in the interval $\left[\frac{M}{2}, \frac{\sqrt{2}M}{2}\right]$. This interval is located at the high frequency range, as for natural volumes, the magnitude of the PPFT3D in this frequency range is usually negligible.

3.1 The normalized correlation measure

In order to improve the robustness of the algorithm with respect to noise and intensity changes, we replace the L_1 norm in Eq. 3.8 with the normalized correlation [23]. The normalized correlation ΔV_N^d is defined by

$$\Delta V_N^d\left(\theta_i, \phi_j\right) \triangleq \frac{\sum_{0 \le r_k \le M} \overline{M_1^d}\left(r_k, \theta_i, \phi_j\right) \overline{M_2^d}\left(r_k, \theta_i, \phi_j\right)}{\sigma_1(\theta_i, \phi_j)\sigma_2(\theta_i, \phi_j)}, \tag{3.9}$$

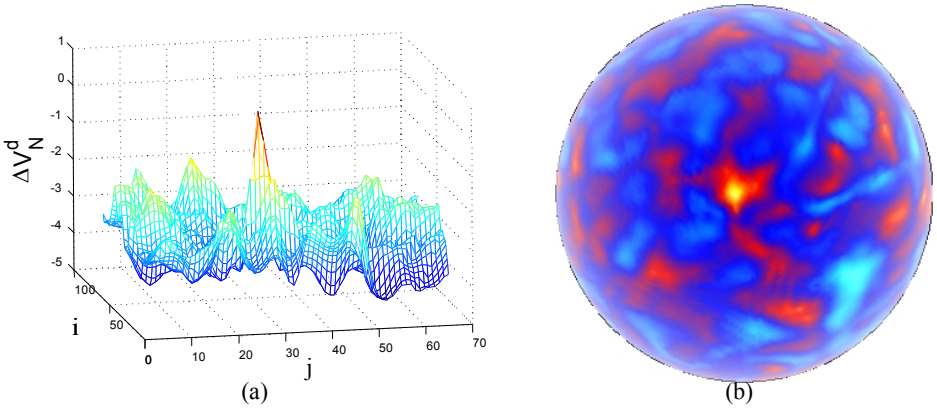
where

$$\overline{M_l^d}\left(r_k,\theta_i,\phi_j\right) \triangleq M_l^d\left(r_k,\theta_i,\phi_j\right) - \frac{1}{N(\theta_i,\phi_j)} \sum_{0\le r_m\le M} M_l^d\left(r_m,\theta_i,\phi_j\right), \qquad l=1,2,$$

$$\sigma_l(\theta_i,\phi_j) \triangleq \sqrt{\frac{1}{N(\theta_i,\phi_j)} \sum_{0\le r_k\le M} \left(M_i^d\left(r_k,\theta_i,\phi_j\right) - \overline{M_i^d}\left(r_k,\theta_i,\phi_j\right)\right)^2}, \qquad l=1,2,$$

$N(\theta_i,\phi_j)$ is the number of samples of the pseudo-polar grid with direction (θ_i,ϕ_j) whose radius is less than $M/2$, M_1^d and M_2^d are given by Eq. 3.7, and M is the radial resolution of the 3-D pseudo-polar grid. Equation 3.9 is evaluated for all θ_i and ϕ_j such that $(r,\theta_i,\phi_j)\in\Gamma$ for some r, where Γ is given by Eq. 2.15.

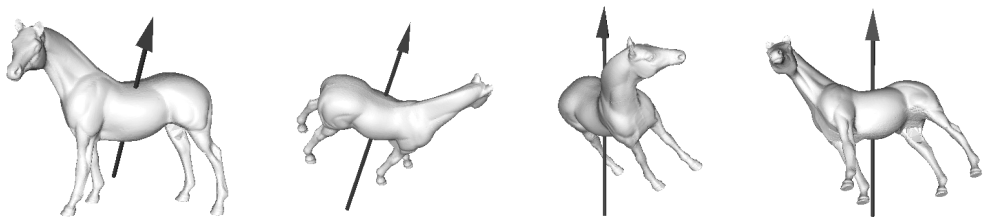
The normalized correlation is more robust than the L_1 norm since it normalizes differences in gain and offset. The normalized correlation appears in a similar form in other papers [23]. For our application, its superiority over Eq. 3.2 is verified experimentally in Section 6. Note that the normalized correlation is separately computed for each ray. An example of ΔV_N^d (Eq. 3.9) for the Skull volume (Fig. 4e) is shown in



(a) (b)

Figure 2: Applying the normalized correlation measure to compute ΔV_N^d of the Skull volume (Fig. 4e) and its rotated replica. (a) A visualization of $-\log\left(\Delta V_N^d\right)$. The surface's peak corresponds to the maximum of the ΔV_N^d. (b) ΔV_N^d overlaid on a sphere. The "hot" values correspond to the maximum. Notice its small angular support.

Fig. 2. Figure 2a is the surface $-\log(\Delta V_N^d)$, where the maximum is clearly visible and detectable. Figure 2b shows the small support of the maximum of ΔV_N^d and demonstrates the advantages of using the PPFT3D.



(a) (b) (c) (d)

Figure 3: Alignment of the rotation axis. The volumes in (a) and (b) are the input volumes, which are related by a rotation about the rotation axis. After recovering the rotation axis, the volumes are rotated such that the rotation axis is parallel to the z axis. Thus, the volumes in (c) and (d) are related by a translation and a planar rotation about the z axis.

4 Planar rotation

Given the rotation axis $\vec{n}$, computed by Eqs. 3.2 and 3.3, we use Eq. 3.4 to rotate V_1 and V_2 (Figs. 3a and 3b) such that the rotation axis $\vec{n}$ is parallel to the z axis (Figs. 3c and 3d). This results in translated and rotated volumes, whose relative rotation is around the z axis.

Given two vectors $\vec{u}_1, \vec{u}_2 \in \mathbb{R}^3$, the 3-D rotation that transforms $\vec{u}_1$ to $\vec{u}_2$ is given by $\widetilde{R} = (\vec{u}, \psi)$, where $\psi = \arccos\left(\frac{\vec{u}_1 \cdot \vec{u}_2}{||\vec{u}_1|| \, ||\vec{u}_2||}\right)$ is the rotation angle and $\vec{u} = \vec{u}_1 \times \vec{u}_2$ is the rotation axis. In order to align the rotation axis with the z axis, we set $\vec{u}_1 = (0, 0, 1)$, compute $\vec{u}_2$ using Eq. 3.3, and use ψ and $\vec{u}$ to compute the rotation matrix $\tilde{R}$.

We apply $\widetilde{R}$ to V_1 and V_2 and denote the resulting volumes by $\widetilde{V}_1$ and $\widetilde{V}_2$, respectively. $\widetilde{V}_1$ and $\widetilde{V}_2$ are related by a translation and a planar rotation of angle γ around the z axis (see Figs. 3c and 3d). It is possible to estimate the planar rotation by using any corresponding pair of planes in $\widetilde{V}_1$ and $\widetilde{V}_2$ that are perpendicular to the z axis [24, 13]. However, we improve the robustness of the estimate by extending the image registration scheme in [18] to cylindrical coordinates. We define the cylindrical Fourier transform of a volume V, denoted $\mathcal{F}^C$, by

$$\mathcal{F}^C(V) = \mathcal{F}_P\left(\mathcal{F}^z_{1D}\left(V\left(x, y, z\right)\right)\right), \tag{4.1}$$

where $\mathcal{F}^z_{1D}$ is the 1D Fourier transform in the z direction and $\mathcal{F}_P$ is the 2-D polar Fourier transform, which operates on each plane that is perpendicular to the z axis.

Given two volume $\tilde{V}_1$ and $\tilde{V}_2$, which are related by a 3-D translation and a relative rotation of angle γ around the z axis, we denote

$$\tilde{M}_1 = \left|\mathcal{F}^C(\tilde{V}_1)\right|, \qquad \tilde{M}_2 = \left|\mathcal{F}^C(\tilde{V}_2)\right|. \tag{4.2}$$

$\tilde{M}_1$ and $\tilde{M}_2$ are related by a planar rotation around the z axis, with no relative translation. In other words, each xy plane in $\tilde{M}_1$ is a rotated replica of the corresponding plane in $\tilde{M}_2$, i.e.,

$$\tilde{M}_1\left(r, \varphi, \omega_z\right) = \tilde{M}_2\left(r, \varphi + \gamma, \omega_z\right), \tag{4.3}$$

where r and φ are polar coordinates and γ is the relative planar rotation of the input volumes V_1 and V_2 around the rotation axis $\vec{n}$ (see Section 3). The relative rotation γ is recovered by finding the minimum of

$$\Delta\tilde{M}\left(\varphi\right) = \int_0^{\pi}\int_0^{\pi} \left|\tilde{M}_1\left(r, \varphi, \omega_z\right) - \tilde{M}_2\left(r, -\varphi, \omega_z\right)\right| dr\, d\omega_z. \tag{4.4}$$

As shown in [18], $\Delta\tilde{M}$ is minimal when $\varphi = -\gamma/2$. Equation 4.4 uses the magnitude of the Fourier transforms and therefore, due to conjugate symmetry, the rotation angle can be either γ or $\gamma + \pi$. This ambiguity is resolved in Section 5.

Equation 4.1 is discretized by applying the 1-D FFT in the z direction and then computing the 2-D pseudo-polar FFT of each xy plane. The angular axis is reversed in Eq. 4.4 by applying a left-to-right flip on each xy plane.

5 3-D translation estimation

Given the rotation parameters (α, β, γ), the 3-D rotation matrix R is computed by using Eq. 3.1. Let $\widetilde{V}_2(\vec{x}) = V_2(R\vec{x})$. V_1 and $\widetilde{V}_2$ are related by a 3-D translation, which is recovered by using the phase-correlation algorithm [14, 13]. As explained in Section 4, we recover the planar rotation γ by using the magnitude of the cylindrical Fourier transform. Therefore, we get that both γ and $\gamma + \pi$ are possible solutions for the planar rotation. To find the correct planar rotation, we rotate the original volume V_2 by both (α, β, γ) and $(\alpha, \beta, \gamma + \pi)$, and recover the translation by using the phase correlation. The value of the phase correlation function for each set of rotation parameters measures the quality of the alignment. Therefore,

the correct planar rotation is the angle that corresponds to the higher value of the phase correlation function. The accuracy of the phase correlation scheme is limited to integral values. Subpixel accuracy and improved robustness to noise can be achieved by applying [25].

6 Experimental results

The proposed algorithm was applied to the volumes shown in Fig. 4. For each input volume, a set of rotated and translated replicas was created using bilinear interpolation, without applying any other processing such as smoothing or denoising. All volumes are of size 64^3 voxels, and the average spacing of the 3-D pseudo-

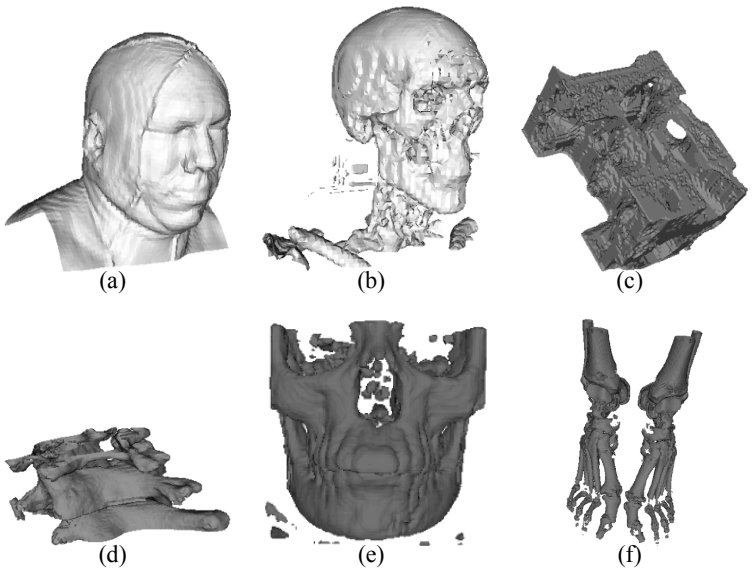


Figure 4: Volumes that are used to evaluate the performance of the algorithm. (a) and (b) are different iso-surface visualizations of the same MRI scan of a human head. (c) A CAD generated model of an engine. (d) A MRI scan of a human spine. (e) A CT scan of a human skull. (f) A MRI scan of human feet.

polar grid is $\Delta\theta = \frac{90^\circ}{2N} = 0.7^\circ$ and $\Delta\phi = \frac{180^\circ}{N} = 1.4^\circ$, $N = 64$. The translations are randomly chosen in the range of $[-10, 10]$ pixels in each direction. Neither the accuracy of the translation estimation nor the resolving between γ and $\gamma + \pi$ are presented, since they are computed by a straightforward implementation of the 2-D phase-correlation algorithm and are not in the focal point of this paper.

The results, given in Table 1, show the accuracy of the proposed algorithm in noise-free settings. In

all cases the registration accuracy is of the order of the angular spacing of the 3-D pseudo-polar grid. The accuracy of the registration is the same for all volumes.

Figure 5 presents the performance of the algorithm in noisy situations. The figure shows the standard deviations (STD) of the angular registration errors, as a function of σ_n, the STD of the additive White Gaussian Noise (WGN). The results are computed by adding noise to the Head and Skull volumes, shown in Figs. 4a and 4e, respectively. Obviously, different noise is added to the original volume and to its rotated replica. For each noise level σ_n we average the alignment error over the two sets of registered volumes. It follows that for non-noisy input volumes the accuracy of the proposed algorithm is in the range of $0° - 1°$, which corresponds to the average angular spacing of the 3-D pseudo-polar grid. As we can see from Fig. 5, the normalized correlation ΔV_N^d (Eq. 3.9) is less sensitive to additive noise. Up to $\sigma_n = 200$, the scheme is not affected by noise, and from that point on, its accuracy degrades. The non-normalized ΔV^d, given by Eq. 3.8, achieves an accuracy better than $1°$ for non-noisy volumes and then degrades. To conclude, the proposed algorithm is capable of giving reasonable estimates in extremely noisy situations.

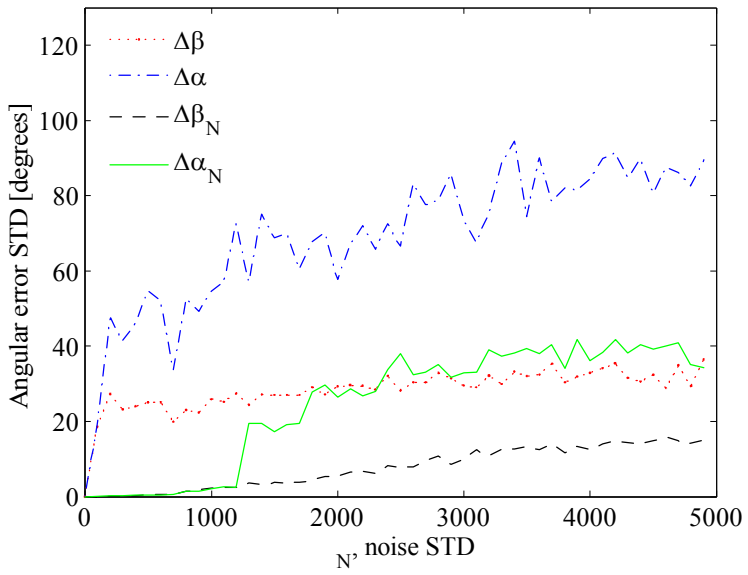


Figure 5: The angular registration error as a function of the standard deviation of the noise. $\Delta\alpha$ and $\Delta\beta$ are the errors in the estimation of the rotation axis when using the L_1 norm (ΔV^d). $\Delta\alpha_N$ and $\Delta\beta_N$ are the estimation errors when using the normalized correlation (ΔV_N^d). Using ΔV_N^d results in improved accuracy, especially for noisy volumes.

	Actual parameters			Estimated parameters			Estimation errors		
	α	β	γ	α	β	γ	$\lvert\varepsilon_\alpha\rvert$	$\lvert\varepsilon_\beta\rvert$	$\lvert\varepsilon_\gamma\rvert$
Human Head	82.89	45.00	28.21	83.39	47.81	28.11	0.50	2.81	0.10
	80.98	25.84	44.50	80.36	25.11	44.80	0.62	0.73	0.30
	12.02	15.11	78.47	11.89	15.70	79.42	0.13	0.59	0.95
Engine	82.89	45.00	28.21	81.37	43.19	27.78	1.51	1.80	0.42
	80.98	25.84	44.50	80.78	24.15	44.86	0.19	1.68	0.36
	12.02	15.11	78.47	12.88	15.67	76.94	0.86	0.56	1.52
Spine	82.89	45.00	28.21	84.46	43.76	26.36	1.57	1.23	1.84
	80.98	25.84	44.50	80.07	27.21	44.33	0.90	1.37	0.16
	12.02	15.11	78.47	11.03	13.80	79.94	0.98	1.30	1.47
Skull	82.89	45.00	28.21	84.35	43.68	29.94	1.46	1.31	1.73
	80.98	25.84	44.50	79.90	27.81	43.55	1.0	1.97	0.94
	12.02	15.11	78.47	13.23	14.86	77.11	1.21	0.24	1.35
feet	82.89	45.00	28.21	84.52	44.36	29.70	1.63	0.63	1.49
	80.98	25.84	44.50	79.90	25.09	43.45	1.0	0.74	1.0
	12.02	15.11	78.47	10.97	14.57	79.05	1.0	0.53	0.58

Table 1: Registration results that were obtained by using the L_1 distance ΔV^d for the volumes shown in Fig. 4. Columns 1 through 3 present the actual rotation parameters. Columns 4 through 6 present the estimated rotation parameters. Columns 7 through 9 present the estimation errors. All volumes are of size $64 \times 64 \times 64$.

7 Summary and conclusions

The paper presents a general purpose volume registration algorithm, which operates in the frequency domain and solves the alignment problem by computing a radial distance measure in two and three dimensions. By using Euler's theorem, the original problem, which involves six parameters, is decoupled into three sub-problems: estimating the rotation axis, estimating the planar rotation, and computing the translation. The computation is based on the PPFT2D and PPFT3D algorithms, which are fast and accurate. Compared to other schemes, we provide accurate results which are robust to noise and have a predictable execution time and controlled error.

Future work includes the registration of biological data, which is often noisy and can therefore benefit from the scheme's robustness to noise. We currently study the analysis of 3-D symmetries, which is a fundamental task in 3-D modeling and analysis. Symmetry axes can be considered as the multiple solutions of the registration of a symmetric object with itself. We have successfully applied a similar approach to images in [26].

References

[1] J. Wyngaerd and L. Van Gool, "Automatic crude patch registration: Toward automatic 3D model building," *Computer Vision and Image Understanding*, vol. 87, no. 1-3, pp. 8–26, July 2002.

[2] Raouf Benjemaa and Francis Schmitt, "Fast global registration of 3D sampled surfaces using a multi-Z-buffer technique," in *International Conference on Recent Advances in 3D Digital Imaging and Modeling*, 1997, pp. 113–120.

[3] Georgios Papaioannou, Evaggelia-Aggeliki Karabassi, and Theoharis Theoharis, "Reconstruction of three-dimensional objects through matching of their parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 114–124, January 2002.

[4] Marcos Rodrigues, Robert Fisher, and Yonghuai Liu, "Introduction: Special issue on registration and fusion of range images," *Computer Vision and Image Understanding*, vol. 87, pp. 1–7, 2002.

[5] J. A. Kovacs, P. Chacón, Y. Cong, E. Metwally, and W. Wriggers, "Fast rotational matching of rigid bodies by fast Fourier transform acceleration of five degrees of freedom," *Acta Cryst.*, vol. D59, no. 2, pp. 1371–1376, 2003.

[6] E. Katchalski-Katzir, I. Shariv, M. Eisenstein, A. A. Friesem, C. Aflalo, and I. A. Vakser, "Molecular surface recognition: Determination of geometric fit between proteins and their ligands by correlation techniques," in *Proc. Natl. Acad. Sci. USA*, vol. 89, March 1992, pp. 2195–2199.

[7] Paul J. Besl and Neil D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–255, February 1992.

[8] T. Pajdla and L. Van Gool, "Matching of 3-D curves using semi-differential invariants," in *Proc. IEEE Int. Conf. Computer Vision*. IEEE Computer Society Press, Cambridge, MA, 1995, pp. 390–395.

[9] Gregory C. Sharp, Sang W. Lee, and David K. Wehe, "ICP registration using invariant features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 90–102, January 2002.

[10] L. Lucchese, G. Doretto, and G. Cortelazzo, "A frequency domain technique for range data registration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 11, pp. 1468–1484, 2002.

[11] P. Thévenaz, U. Ruttimann, and M. Unser, "A pyramid approach to subpixel registration based on intensity," *IEEE Transactions on Image Processing*, vol. 7, no. 1, pp. 27–41, January 1998.

[12] Guido Maria Cortelazzo, Gianfranco Doretto, and Luca Lucchese, "Free-form textured surfaces registration by a frequency domain technique," *IEEE International Conference Image Processing*, vol. 1, pp. 813–817, October 1998.

[13] S. Reddy and B. N. Chatterji, "An FFT-based technique for translation, rotation, and scale-invariant image registration," *IEEE Transactions on Image Processing*, vol. 3, no. 8, pp. 1266–1270, August 1996.

[14] C. D. Kuglin and D. C. Hines, "The phase correlation image alignment method," *IEEE Conference on Cybernetics and Society*, pp. 163–165, September 1975.

[15] Y. Keller, Y. Shkolnisky, and A. Averbuch, "Algebraically accurate volume registration using euler's theorem and the 3-D pseudo-polar FFT," in *Proceedings, IEEE Conference on Computer Vision and Pattern Recognition*, June 2005.

[16] A. Averbuch, D. Donoho, R. Coifman, M. Israeli, and Y. Shkolnisky, "Fast slant stack: A notion of Radon transform for data in cartesian grid which is rapidly computable, algebrically exact, geometrically faithful and invertible," *SIAM Scientific Computing*, To appear.

[17] A. Averbuch and Y. Shkolnisky, "3D Fourier based discrete Radon transform," *Applied and Computational Harmonic Analysis*, vol. 15, pp. 33–69, 2003.

[18] Y. Keller, Y. Shkolnisky, and A. Averbuch, "The angular difference function and its application to image registration," *To appear in the IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, Available at: http://pantheon.yale.edu/~yk253/publications/pdf/adf_2d_registration.pdf.

[19] D. H. Bailey and P. N. Swarztrauber, "The fractional Fourier transform and applications," *SIAM Review*, vol. 33, no. 3, pp. 389–404, September 1991.

[20] L. R. Rabiner, R. W. Schafer, and C. Rader, "The chirp z-transform algorithm," *IEEE Transcations on Audio ElectroScoustics*, vol. AU, no. 17, pp. 86–92, June 1969.

[21] E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision*. New Jersey: Prentice-Hall, 1998, vol. pp. 333 - 334.

[22] B. Porat, *A Course in Digital Signal Processing*. John Wiley Pub., 1997.

[23] F. M. Dickey and L. A. Romero, "Normalized correlation for pattern recognition," *Optics Letters*, vol. 16, no. 15, pp. 1186–1188, August 1991.

[24] P. Milanfar, "Two-dimensional matched filtering for motion estimation," *IEEE Transactions on Image Processing*, vol. 8, no. 3, pp. 438–443, March 1999.

[25] W. Hoge and D. Mitsouras, "Registration of multi-dimensional image data via sub-pixel resolution phase correlation," *IEEE International Conference Image Processing*, vol. 2, pp. 707–710, September 2003.

[26] Y. Keller and Y. Shkolnisky, "An algebraic approach to symmetry detection," in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR 2004)*, vol. 3, August 2004, pp. 186–189.