

# 6-DOF Pose Estimation from Single Ultrasound Image Using 3D IP Models

Bo Zheng<sup>1</sup> Ryo Ishikawa<sup>2</sup> Takeshi Oishi<sup>1</sup> Jun Takamatsu<sup>1</sup> Katsushi Ikeuchi<sup>1</sup>

<sup>1</sup> The University of Tokyo  
Ikeuchi Lab, IIS, 4-6-1 Komaba,  
Meguro-ku, Tokyo, 153-8505, JAPAN

{zheng, oishi, j-taka, ki}@cvl.iis.u-tokyo.ac.jp

<sup>2</sup> Medical Imaging Project, Canon Inc.  
4-6-1 shimomaruko,  
oda-ku, Tokyo, 146-8501, JAPAN

ishikawa.ryo@canon.co.jp

## Abstract

*Real-time pose estimation of a free-hand Ultrasound (US) image without any position sensor is desirable for diagnostics and image guidance, but it suffers from poor image quality as a result of processing. The paper confronts this problem by proposing a new 6-DOF pose estimation method based on a fast registration process making use of 3D implicit polynomials (IP). The proposed registration method has some main advantages over the traditional methods. First, our formulation is based on minimization of energy functional derived from IP gradient flow, and thus it is more efficient than traditional registration because it eliminates the cost for calculating point-wise correspondences. Second, the efficient calculation benefits from the property of IP having few coefficients, which means that both the gradient field and its transformation can be calculated in an extremely light manner. Third, applying a real-time US image pose estimation, we demonstrate the capabilities of overcoming the limitations of unconstrained free-hand US data, resulting in robust and fast pose estimation.*

## 1. Introduction

To support medical diagnosis, various imaging modalities, such as CT scan, MRI, PET, and ultrasound (US), are widely used in clinics. Among these modalities, US has beneficial characteristics such as free-hand manner, non-invasiveness, compactness, low cost, and synchronization of operations and imaging. Thus US is attractive for assistance with surgical operations and real-time diagnosis of problems with the circulatory system, abdomen, breast, prostate gland, etc.

However, US images are notorious for the poor image quality, due to speckle noises, low signal-to-noise ratio, occlusions, and uniform brightness. And field of view (FOV) in US imaging is very limited; in severe cases, only 2D

cross-sectional images are obtained. These aspects confuse the doctors in making the right decisions for diagnosis.

In order to solve these issues, some recent literature advocates the fusion of modality techniques. For example, before the surgical operation, 3D models of target parts are obtained by rich but time-consuming modalities such as CT, MRI, and PET. By superimposing US images obtained during the operation on these 3D models, the result will provide rich information to help a doctor's diagnosis. To achieve this, the key for superimposing is to estimate the pose of US images related to the images derived from multiple modalities.

To do this, the methods in [11, 20] bind the optical position sensors to a US probe, and measure the relative US position to 3D models; the methods in [9, 13, 21] estimate the relative positions according to the image features of US images and preoperative 3D models. For enhancing robustness, the methods in [3, 12, 22] combine the information from position sensors and image features. Although each of the methods has its effectiveness, they suffer from heavy computation and thus cannot work in real time.

To overcome the time-consuming issue, we present a method to register the 3D model and 2D US image using boundary information, which is independent of the types of modalities. As illustrated in Fig. 1, our method represents 3D model with an implicit polynomial (IP) in an advance offline process. In an online process, for example, during a surgical operation, it aligns a 3D IP model to a 2D US image based on the energy minimization manner. Then the desired information regarding the 2D contour, the 2D position, and the relative 3D pose of the US probe is obtained.

Our method has three significant advantages: 1) Because our method is based on the registration technique, no special equipment, such as position sensor, is required. 2) Differentiability of IPs enables the application of gradient-based minimization technique. Thus our method achieves fast registration. 3) Since IP representation needs few parameters

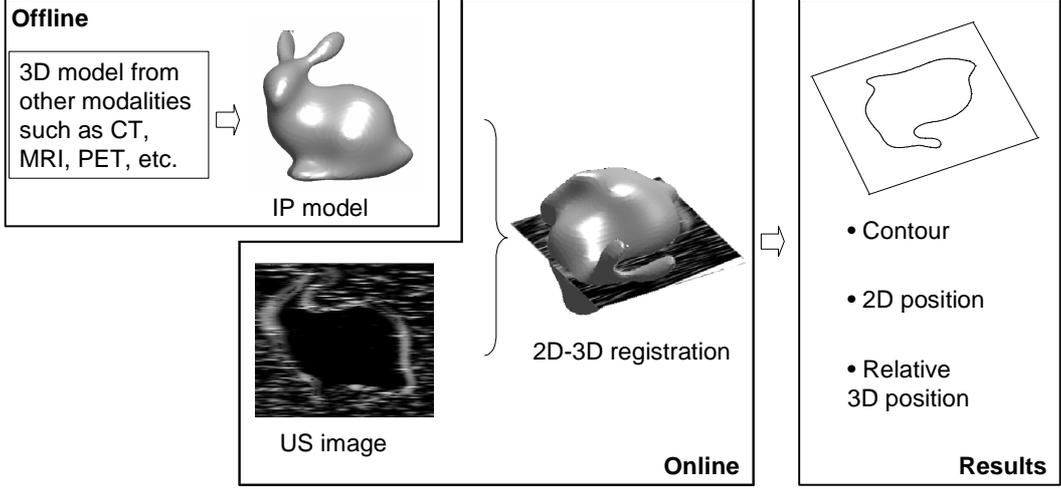


Figure 1. Method outline: offline, we model the boundary of the target object, which is extracted from 3D imaging techniques such as CT, MRI, and PET, with an implicit polynomial (IP). The online process registers the 3D IP model and 2D US image using gradient flows of the IP. Then, as results, the image contour, the 2D position, and the relative 3D position of the IP model are extracted.

and is based on the implicit form, both the calculation of gradient flow and its transformation for the minimization are extremely fast.

This paper is organized as follows: Section 2 introduces the mathematics of IP modeling and transformation; In Section 3, we propose the registration technique using IP, based on both a general case and a US image case, followed by implementation in Section 4; Section 5 reports experimental results followed by discussion and conclusions in Sections 6.

## 2. Mathematics of IP

The 3D organ model can be easily derived from the pre-operative 3D imaging that was performed with a modality such as MRI, CT-SCAN, or 3D US. In this section, supposing the boundary data is given from the segmentation result of a volume data obtained by one of the above modalities, let us present the process of modeling these data with IP representation.

### 2.1. Modeling

Suppose the model  $\mathcal{M}$  represents a closed surface that defines two 3D regions in the 3D space  $\mathcal{R}_\Omega$ , namely the inner region  $\mathcal{R}_\mathcal{M}$  enclosed by the model  $\mathcal{M}$  and the outer region  $\mathcal{R}_\Omega \setminus (\mathcal{R}_\mathcal{M} \cup \mathcal{M})$ . The model is represented implicitly by its distance transform, *i.e.*, the signed distance function,

$$\Phi(\mathbf{x}) = \begin{cases} 0, & \mathbf{x} \in \mathcal{M} \\ +d(\mathbf{x}, \mathcal{M}), & \mathbf{x} \in \mathcal{R}_\mathcal{M} \\ -d(\mathbf{x}, \mathcal{M}), & \mathbf{x} \in \mathcal{R}_\Omega \setminus (\mathcal{R}_\mathcal{M} \cup \mathcal{M}) \end{cases}, \quad (1)$$

where  $\mathbf{x} = (x \ y \ z)$  is the location of one data point in Cartesian coordinates, and  $d(\mathbf{x}, \mathcal{M}) = \min_{\mathbf{x}_\mathcal{M} \in \mathcal{M}} \|\mathbf{x} - \mathbf{x}_\mathcal{M}\|$  is

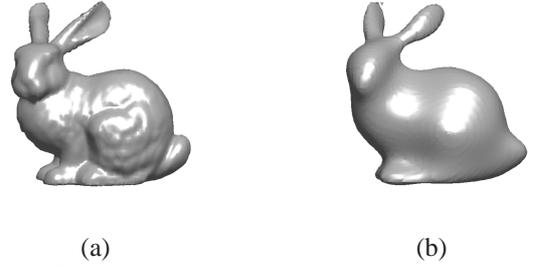


Figure 2. IP fitting results: (a) original object; (b) modeling with 8-degree IP.

the Euclidean distance from  $\mathbf{x}$  to the model consisting of the points  $\mathbf{x}_\mathcal{M}$ . In this work, we adopt an implicit polynomial (IP) to approximately satisfy  $\Phi$ , and it can be formulated as:

$$\begin{aligned} \Psi_{3D}^n(\mathbf{x}) &= \sum_{0 \leq i, j, k; i+j+k \leq n} a_{ijk} x^i y^j z^k \\ &= \underbrace{(1 \ x \ \dots \ z^n)}_{\mathbf{m}(\mathbf{x})^T} \underbrace{(a_{000} \ a_{100} \ \dots \ a_{00n})}_{\mathbf{a}}^T, \quad (2) \end{aligned}$$

where  $n$  is the degree of an IP, and an  $n$ -degree IP can be represented as a inner product between a monomial vector  $\mathbf{m}(\mathbf{x})$  and a coefficient vector  $\mathbf{a}$  as:  $\mathbf{m}(\mathbf{x})^T \mathbf{a}$ .

To achieve the representation of 3D shapes using an IP zero set, the fast and simple optimization of using linear least-squares methods is advocated in the considerable literature [4, 8, 14, 17, 23]. In this paper, we adopt the modeling method proposed by Zheng *et al.* in [23], which allows adaptively determining the moderate IP degree for modeling the shapes with various complexities and gives a more

accuracy IP model. Fig. 2 shows an example in which a 3D model “bunny” is represented with an IP; their method selects an 8-degree IP.

## 2.2. Transformation

When a transformation is applied to the IP model  $\Psi(\mathbf{x}) = \mathbf{m}(\mathbf{x})^T \mathbf{a}$ , vector of monomials  $\mathbf{m}$  is transformed as  $\mathbf{m}' = V(\mathbf{p})\mathbf{m}$ , where we suppose vector  $\mathbf{p}$  consists of the parameters for affine transformation, and the square matrix  $V$  is transformation for monomials. The zero set of the polynomial is defined by  $\mathbf{m}^T \mathbf{a} = 0$ . Thus after substituting  $\mathbf{m} = V^{-1} \mathbf{m}'$ , the transformed coefficients are  $\mathbf{a}' = V^{-T} \mathbf{a}$ .

To our knowledge, although prior literature, such as [19], have proved the existence of matrix  $V$ , the computational implementation is not explicitly described. Tarel *et al.* [16] proposed a tensor-based transformation for IP, but it still suffers from heavy computational cost. Therefore, let us present our symbolic computational in Appendix A based on the Taubin and Cooper’s theory, which can be seen as a kind of incremental calculation that increases beginning from the transformation for first-degree monomials, until it obtains one for  $n$ -th degree monomials. Note: IP model transformation is computationally efficient in the case that the size of the modeled data set is great larger than the number of IP coefficients, since obviously transforming a few IP coefficients is superior to transforming whole data points with respect to computational cost.

In this paper, let us assume that it is under the Euclidean transformation, and then  $\mathbf{p}$  includes 6 DOF parameters only for rotation and translation. If the transformation can be written as  $\mathbf{x}' = R\mathbf{x} + \mathbf{t}$  for one point, then  $\mathbf{a}' = V(R, \mathbf{t})^{-T} \mathbf{a}$  for IP coefficients.

## 3. IP registration

### 3.1. Background of registration methods

Two main kinds of methods are used to achieve the registration: one is the iterative closest point (ICP)-based approach [2, 6, 10], the other is the approach relying on algebraic/geometric invariant natures, such as moment [15], IPs [16, 19], *etc.* The former can achieve fine registration, but requires very time-consuming computation of point-to-point/surface correspondences. The latter can achieve fast registration, but cannot deal with registration in the case of partially overlapping the target objects [15].

Unlike prior registration methods, our method uses approximation of the signed distance generated by IP representation, which avoids time-consuming correspondence computation. Since the other features of our method inherit from the ICP-based registration, partial overlapping issues naturally dissolve.

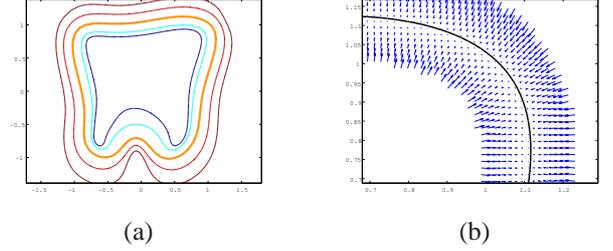


Figure 3. IP gradient field of a tooth shape: (a) level sets of 0.2, 0.1,  $-0.1$  and  $-0.2$  calculated by (4) around the zero level set shown in the thick orange line; (b) distant vectors in the right-upper part of (a) between the level sets of  $-0.1$  and  $0.1$  calculated by (5).

### 3.2. Distance metric

To simplify the explanation, we consider the registration between a discrete point set and an IP model in this subsection. Given  $N$  points  $\{\mathbf{x}\}_1^N \in \Omega$ , where  $\Omega$  is a surface or partial surface of a certain object, the objective of IP registration is to find a transformation through which the zero set  $\{\mathbf{x} | \Psi_{3D}^n(\mathbf{x}) = 0\}$  can “best” match the given points. Thus the registration methods aims at finding the transformation parameters by minimizing the distance between the data set and the zero set of IP, such as:

$$\mathbf{p} = \arg \min_{\mathbf{p}} \left\{ \sum_{i=1}^N \text{dist}(\mathbf{x}_i, \Psi_{3D}^n) \right\}, \quad (3)$$

where  $\text{dist}(\mathbf{x}_i, \Psi_{3D}^n)$  means a certain distance from the data point  $\mathbf{x}_i$  to the zero set  $\{\mathbf{x} : \Psi_{3D}^n(\mathbf{x}) = 0\}$  of the IP. We choose the signed distance manner defined as:

$$\text{dist}(\mathbf{x}, \Psi_{3D}^n) = \frac{\Psi_{3D}^n(\mathbf{x})}{\|\nabla \Psi_{3D}^n(\mathbf{x})\|}, \forall \mathbf{x} \in \Omega, \quad (4)$$

where  $\nabla$  denotes the gradient of IP function; this is the approximation of Euclidean distance for which the computational burden is relatively light (see [18]). For minimization of Eq. (3), we define the vector distance function  $G: \mathcal{R}^3 \rightarrow \mathcal{R}^3$  as

$$G(\mathbf{x}) = \text{dist}(\mathbf{x}, \Psi_{3D}^n) \frac{\nabla \Psi_{3D}^n(\mathbf{x})}{\|\nabla \Psi_{3D}^n(\mathbf{x})\|}, \forall \mathbf{x} \in \Omega. \quad (5)$$

This can be viewed as a resized version of the original gradient vectors with distance  $\text{dist}(\mathbf{x})$ . A 2D example is shown in Fig. 3 for illustration.

### 3.3. Registration

Given the definition of a distance between a certain data set and an IP, the next problem is how to solve the minimization (3) to obtain the transformation parameters. The registration task is to predict the transformation through which IP can be transformed to the data set.

In our method, the minimization is done through the following two steps: for accelerating the convergence, first, it minimizes the function without any constraint in the transformation. This means every point can move freely during the first minimization. Next, it determines the transformation parameters to better describe the first minimization result. These two steps are repeated until convergence.

Let us describe the registration between an IP and a partial object, in a general case and a US image case.

**General case** Eq. (3) can be simply formulated with the energy functional of the sum of differences as follows:

$$E = \int_{\Omega} dist^2(\mathbf{x}, \Psi(\mathbf{x}))d\Omega, \quad (6)$$

where  $\Omega$  represents the partial surface of an object, and  $\mathbf{x} \in \Omega$ . Then, by calculus of variations [7], the Gateaux derivative (first variation) of the functional  $E$  to point  $\mathbf{x}$  can be approximately formulated as

$$\frac{\partial E}{\partial \mathbf{x}} \approx 2G(\mathbf{x}), \quad (7)$$

if we consider  $\nabla\Psi_{3D}$  as a constant value for computational convenience.

Therefore, we need to minimize this functional to satisfy the Euler-Lagrange equation  $\frac{\partial E}{\partial \mathbf{x}} = 0$ . Thus the steepest descent process is executed in the following gradient flow for each point  $\mathbf{x}$ :

$$\frac{\partial \mathbf{x}}{\partial t} = -2G(\mathbf{x}). \quad (8)$$

**US image case** For the task of pose estimation of the US probe, the US images from the 2D probe that image the organ slice boundaries are usually expected to be matched with the corresponding planar intersection of the 3D model.

Furthermore, US images are heavily noise-contaminated. Since precise extraction of the organ's boundary is therefore very difficult, maybe impossible, simple boundary-based registration described in the general case is not sufficient for US images. But fortunately, the organ's inside region is clearly seen in US images as shown in Fig. 1. These two points are key for improving the general method for US images.

For solving the first point, we use a Gaussian smoothed edge indicator defined by [5]. Let  $I(\mathbf{x})$  be a 2D US image function in 3D space, and pixel position  $\mathbf{x} (\in \mathcal{R}^3)$  be in the 3D plane  $\Omega$ . Let  $g(\mathbf{x})$  be the indicator defined as follows:

$$g = \frac{1}{(1 + |\nabla G_{\sigma} * I|/k)^2}, \quad (9)$$

where  $G_{\sigma}$  is a Gaussian filter with standard deviation  $\sigma$ , and  $*$  denotes convolution.

For making use of the second point, we compose the energy function with combining a boundary constraint  $\mathcal{L}(\mathbf{x})$  and an inside constraint  $\mathcal{A}(\mathbf{x})$ , such as

$$\mathcal{E}(\mathbf{x}) = \alpha\mathcal{L}(\mathbf{x}) - \beta\mathcal{A}(\mathbf{x}), \quad (10)$$

where  $\alpha$  and  $\beta$  are constants, and the terms  $\mathcal{L}(\mathbf{x})$  and  $\mathcal{A}(\mathbf{x})$  are defined by

$$\mathcal{L}(\mathbf{x}) = \int_{\Omega} \delta(dist(\mathbf{x}, \Psi(\mathbf{x})) \cdot g(\mathbf{x}))dist^2(\mathbf{x}, \Psi(\mathbf{x}))d\Omega, \quad (11)$$

and

$$\mathcal{A}(\mathbf{x}) = \int_{\Omega} H(-dist(\mathbf{x}, \Psi(\mathbf{x})) \cdot g(\mathbf{x}))dist^2(\mathbf{x}, \Psi(\mathbf{x}))d\Omega, \quad (12)$$

respectively, where  $\delta$  is the univariate Dirac function,  $H$  is the Heaviside function, and  $\Omega$  corresponds to the image plane.

Let us explain the meaning of these terms. Minimize the energy functional  $\mathcal{E}$  in (10), which is equivalent to minimizing  $\mathcal{L}$  in (11), and maximize  $\mathcal{A}$  in (12). First, the energy function with respect to the boundary  $\mathcal{L}$  is equivalent to the integral of edge indicator values along the intersected curve between IP and US image plane because the IP model should be converged around an edge-like part in US images. Second, the energy function with respect to the inside region  $\mathcal{A}$  is the integral of edge indicator values over the inner region surrounded by the intersected curve. It is expected to be as large as possible. To summarize, minimizing  $\mathcal{L}$  and  $-\mathcal{A}$  achieves the registration.

By calculus of variations, the Gateaux derivative (first variation) of the functional  $\mathcal{E}$  in (10) can be approximately written as

$$\frac{\partial \mathcal{E}}{\partial \mathbf{x}} \approx (\alpha\delta(dist \cdot g) - \beta H(-dist \cdot g))G. \quad (13)$$

The steepest descent process for minimization of the functional  $\mathcal{E}$  is the following gradient flow:

$$\frac{\partial \mathbf{x}}{\partial t} = -(\alpha\delta(dist \cdot g) + \beta H(-dist \cdot g))G. \quad (14)$$

## 4. Implementation

### 4.1. Function definition

The Dirac function  $\delta$  in (11) and the Heaviside function  $H$  in (12) are simply defined as:

$$\delta(x) = \begin{cases} 0, & |x| > \kappa \\ \frac{1}{2\kappa}[1 + \cos(\frac{\pi x}{\kappa})], & |x| \leq \kappa \end{cases} \quad (15)$$

and

$$H(x) = \begin{cases} 0, & x < 0 \\ x + x^2/2, & x \geq 0 \end{cases} \quad (16)$$

respectively, where  $\kappa$  is constant and practically set with  $\kappa = 2$  in our experiments.

## 4.2. Estimating the transformation for one step

Let  $X \in \mathcal{R}^{N \times 3}$  preserve  $N$  3D data points. Then the approximation of (8) and (14) by the above difference scheme can be simply written as:

$$\frac{\partial X}{\partial t} = X^{k+1} - X^k. \quad (17)$$

It is the approximation of discrete data set transformation by the above spatial difference scheme in (8) and (14).

$$A = (X^k - \bar{X}^k)^T (X^{k+1} - \bar{X}^{k+1}), \quad (18)$$

where  $\bar{X}$  is a matrix in which each row consists of the mean value (center point) of  $X$ , and if  $A$  is decomposed with singular value decomposition (SVD) algorithm as  $A = USV^T$ , then the transformation is given as:

$$\begin{aligned} R &= UV^T, \\ \mathbf{t} &= \bar{X} - \bar{X}'R^T \end{aligned} \quad (19)$$

where  $R$  and  $\mathbf{t}$  are rotation and translation parameters respectively. Then, if in the case that number of points is not too large, each point  $\mathbf{x}$  will be updated as  $\mathbf{x}^{k+1} = R\mathbf{x}^k + \mathbf{t}$ ; otherwise, IP will be updated as  $\mathbf{a}^{k+1} = V(R, \mathbf{t})^{-T} \mathbf{a}^k$ , through inputting  $R$  and  $\mathbf{t}$  into the algorithm described in Appendix A.

## 5. Experimental Results

The proposed registration method has been applied to a variety of synthetic and real data sets. First, for all the experimental results shown in this section, let us set the pre-conditions as: 1) the time step  $t$  is simply set with 1; 2) the IP models are scaled according to the US images; 3)  $\alpha$  and  $\beta$  in (10) are set with about 0.9 and 0.1 respectively for US image case; 4) US images ( $640 \times 480$  pixel image) are measured by TOSHIBA SONOLAYER SSA-260A with two types of 2D probe: TOSHIBA PLF-703NT (7.5MHz) and TOSHIBA PVF-375MT (3.75MHz); 5) Registration is performed with a modern PC such as Intel core 2 CPU with 2.4 GHz and 2 GB memory, and the algorithm is implemented with Matlab 7.

**Synthesized data in general case** Fig. 4 and Fig. 5 show the registrations in the general case that registers two synthetic data sets to “bunny” object. The synthetic data sets are selected from the “bunny” slice (Fig. 4 (a)) and surface (Fig. 5 (b)). The initial positions are shown with blue points in each figure. By carrying out the registration algorithm described in Section 3, they are successfully registered to the “bunny” IP model with the final positions shown with red points. The mean absolute distance of Eq. (4) of the points for each registration versus the iteration number is plotted in (b) of each figure. The consumed CPU times are 10.3 *ms* and 23.1 *ms* respectively.

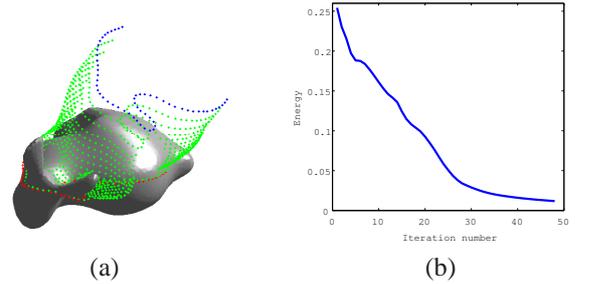


Figure 4. IP registration with planar 3D points: (a) registration process between “bunny” IP model and points (blue points: initial position; green points: selected mid-step positions; red points: final position). (b) the mean absolute distance  $\frac{1}{N} \sum |dist(\mathbf{x}, \Psi)|$  versus iteration number.

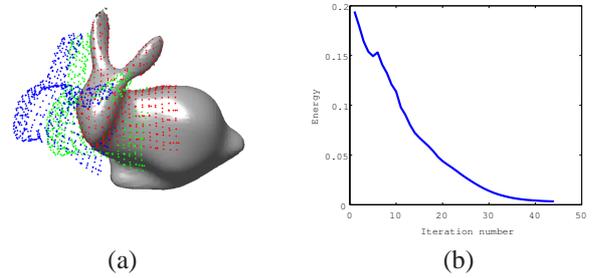


Figure 5. IP registration with surface 3D points: (a) registration process between “bunny” IP model and points (blue points: initial position; green points: selected mid-step positions; red points: final position). (b) the mean absolute distance  $\frac{1}{N} \sum |dist(\mathbf{x}, \Psi)|$  versus iteration number.

**US images** Fig. 6 shows a US image case, where the US images are obtained by measuring phantom ATS514 [1]. Then we use a cylinder-like IP model (2-degree IP), according to the shape of the phantom, to register the US images. The initial position for each case is set as follows: the cylinder-like IP model cross the US image plane with an angle of  $85^\circ$ , and it is placed near to one of the hole-like areas of the images, as shown in the second row of Fig. 6. The registration result of each case is shown at each column, respectively; the first row shows the intersection curve between the IP and the US plane at the final iteration, and the second and third row show the initial and final position of each case. For each case, the consumed CPU time is within 30 *ms*.

Fig. 8 shows other results of the US images that are obtained by measuring a duck toy made of rubber in the cistern shown in Fig. 7 left. We modeled the duck object with an 8-degree IP shown in Fig. 7 right. The initial position for each case is set randomly. The registration result of each case is shown in Fig. 8, where the first row shows the intersection curve between the IP and US image planes at the

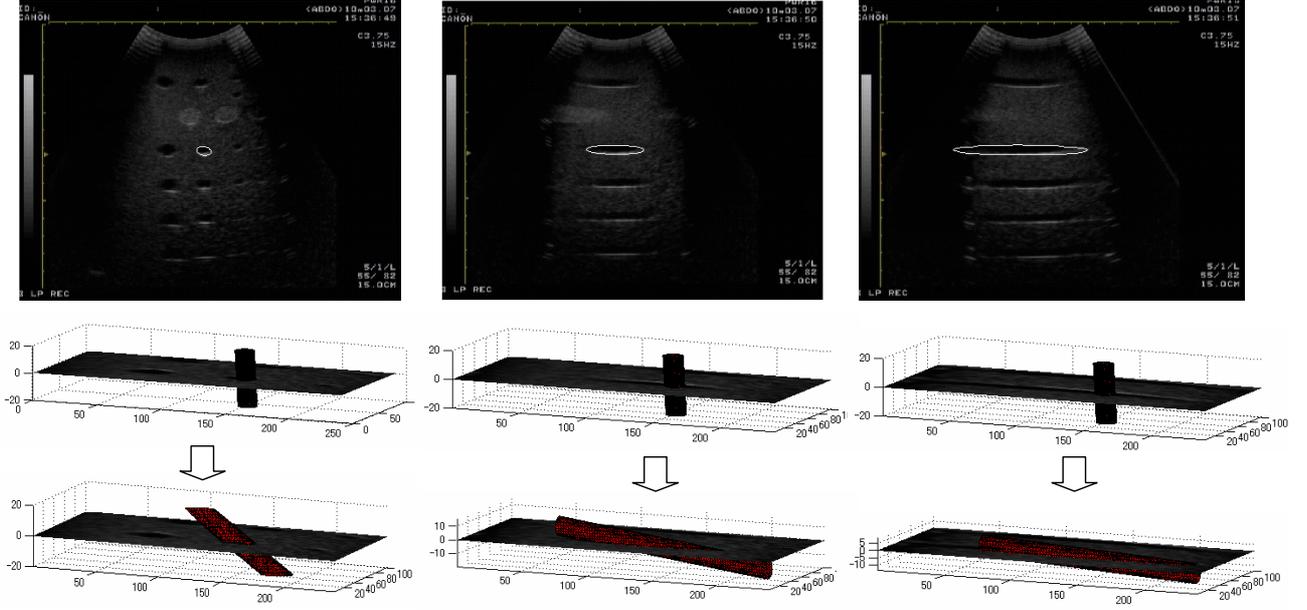


Figure 6. Pose Estimation for phantom AT514. First row: intersection curve of IP and US image (white points); Second row: initial positions of IP; Third row: final positions of IP.

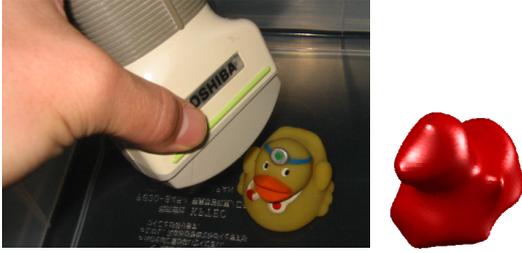


Figure 7. Left: photograph of measuring a duck toy. Right: IP model of duck object

final iteration and the second and third rows show the final position of each case from a different view. For each case, the consumed CPU time is within 180 *ms*.

## 6. Conclusions

In this paper, we presented a novel pose estimation method for US images using gradient flows of IPs. Acceleration of the registration is achieved by these two aspects: 1) IP transformation can be performed faster than transformation of data sets; the number of IP coefficients is usually much less than the number of data sets. 2) Thanks to IP gradient flows, the registration procedure can be performed by simple steepest descent-based minimization without time-consuming calculation of point-wise correspondences.

We demonstrated the performance of the proposed algorithm using synthetic data sets and real US images, and in

particular we showed its robustness against the presence of weak boundaries and strong noise. Our algorithm works in real time (at most 180 *ms* in our conducted experiments), and therefore has the potential for various real-time US diagnosis and image guidance.

## Appendix

### A. Implementation for IP transformation

#### A.1 Pure rotation

First let us explain some notations for polynomial operation which was first introduced by Taubin and Cooper [19]. Let a coefficient  $a_{ijk}$  of an IP in Eq. (2) be presented as  $\frac{\Phi_{ijk}}{i!j!k!}$ , and a vector  $\Phi_{[l]} = (\frac{\Phi_{l00}}{\sqrt{l!0!0!}}, \frac{\Phi_{l-1,1,0}}{\sqrt{(l-1)!1!0!}}, \dots, \frac{\Phi_{00l}}{\sqrt{0!0!l!}})^T$  is according to the  $l$ -th form of IP. An operation on this vector is defined as:  $\Phi_{[l,m]} = \Phi_{[l]} \star \Phi_{[m]}^T$ , where  $\star$  represents the classic matrix multiplication with the difference that the individual element-wise multiplications are performed according to the rule  $\frac{\Phi_{ijk}}{\sqrt{i!j!k!}} \star \frac{\Phi_{abc}}{\sqrt{a!b!c!}} = \frac{\Phi_{i+a,j+b,k+c}}{\sqrt{i!j!k!}\sqrt{a!b!c!}}$ . For example,

$$\Phi_{[1]} = \left( \frac{\Phi_{100}}{\sqrt{1!0!0!}}, \frac{\Phi_{010}}{\sqrt{0!1!0!}}, \frac{\Phi_{001}}{\sqrt{0!0!1!}} \right)^T, \quad (20)$$

and

$$\Phi_{[1,1]} = \Phi_{[1]} \star \Phi_{[1]}^T = \begin{pmatrix} \Phi_{200} & \Phi_{110} & \Phi_{101} \\ \Phi_{110} & \Phi_{020} & \Phi_{011} \\ \Phi_{101} & \Phi_{011} & \Phi_{002} \end{pmatrix}. \quad (21)$$

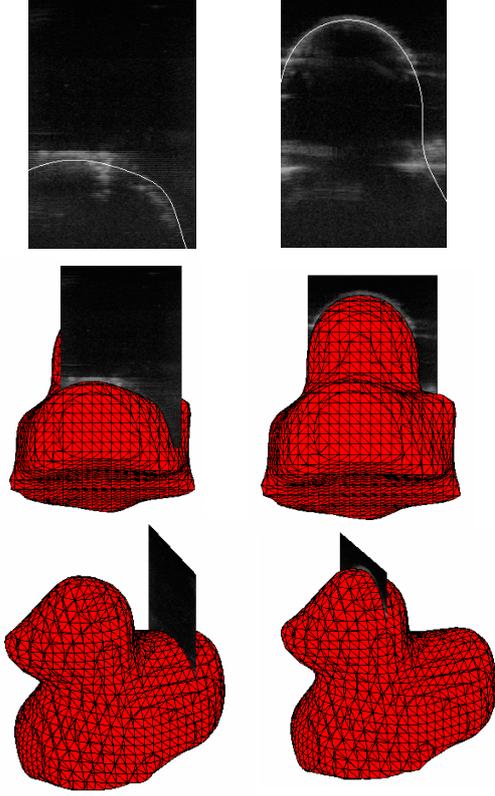


Figure 8. Pose Estimation for duck toy. First row: the intersection of IP and US images (shown by white line). Second and third rows: final positions from different view

In [19], it was pointed out that under a non-singular coordinate transformation  $A$ , *e.g.*,  $\Phi'_{[l]} = A_{[l]} \Phi_{[l]}$ , the transformed coefficient matrix is given by

$$\Phi'_{[l,m]} = A_{[l]}^{-T} \Phi_{[l,m]} A_{[m]}^{-1}, \quad (22)$$

where  $A_{[l]}$  is a non-singular  $h_l \times h_l$  transformation matrix (assuming  $h_l$  be the number of monomials in the  $l$ -th form), and  $A_{[m]}$  is in the same way. From this equation, we know that if  $A_{[l]}$  and  $A_{[m]}$  are known, then a linear relationship between  $\Phi'_{[l,m]}$  and  $\Phi_{[l,m]}$  can be found out; that is, the element-wise correspondence can be linearly expressed as

$$\Phi'_{ijk} = \sum_{\beta; i,j,k \leq l+m} a_{\alpha\beta} \Phi_{ijk} = \mathbf{a}_{\alpha}^T \Phi_{[l+m]}. \quad (23)$$

Then since all of the elements of  $\Phi_{[l+m]}$  are contained in  $\Phi_{[l,m]}$ , and  $\Phi'_{[l+m]}$ 's elements are contained in  $\Phi'_{[l,m]}$ , a new linear correspondence can be constructed between  $\Phi_{[l+m]}$  and  $\Phi'_{[l+m]}$  by arranging the necessary elements in the right

order into  $\Phi'_{[l+m]}$  as:

$$\begin{aligned} \Phi'_{[l+m]} &= (\Phi'_{l+m,0,0}, \Phi'_{l+m-1,1,0}, \Phi'_{l+m-2,2,0}, \dots)^T \\ &= (\mathbf{a}_1^T \Phi_{[l+m]}, \mathbf{a}_2^T \Phi_{[l+m]}, \mathbf{a}_3^T \Phi_{[l+m]}, \dots)^T. \end{aligned} \quad (24)$$

Thus the  $(l+m)$ -th transformation matrix is constructed as:  $A_{[l+m]} = [\mathbf{a}_1^T, \mathbf{a}_2^T, \mathbf{a}_3^T, \dots]^T$ .

As a simple example, the elements of  $\Phi_{[2]}$  ( $= (\Phi_{200}, \Phi_{110}, \Phi_{101}, \Phi_{020}, \Phi_{011}, \Phi_{002})^T$ ) are contained in  $\Phi_{[1,1]}$  shown in (21), and the elements of  $\Phi'_{[2]}$  are contained in  $\Phi'_{[1,1]}$ . From the equation  $\Phi'_{[1,1]} = R_{[1]} \Phi_{[1,1]} R_{[1]}^T$ , (supposing  $R_{[1]}$  is a pure rotation matrix) we can find out a relationship of a linear expansion for the first element of  $\Phi'_{[1,1]}$  as:  $\Phi'_{200} = \mathbf{r}_1 \Phi_{[1,1]} \mathbf{r}_1^T$ , where  $\mathbf{r}_1$  is the first row of  $R_{[1]}$ . Then since all the elements in  $\Phi_{[1,1]}$  are contained in  $\Phi_{[2]}$ , we can find a linear relationship  $\mathbf{r}_1 \Phi_{[1,1]} \mathbf{r}_1^T = \mathbf{a}_1^T \Phi_{[2]}$ . Thus  $\mathbf{a}_1$  can be viewed as the first row of  $A_{[2]}$ .

Therefore, transformation matrix for different forms can be calculated in an incremental manner as:

---

#### Algorithm

---

- 1) Initialization: given  $A_{[1]}$ ;
  - 2)  $\Phi'_{[1,1]} = A_{[1]}^{-T} \Phi_{[1,1]} A_{[1]}^{-1} \longrightarrow A_{[2]}$ ;
  - 3)  $\Phi'_{[1,2]} = A_{[1]}^{-T} \Phi_{[1,2]} A_{[2]}^{-1} \longrightarrow A_{[3]}$ ;
  - ⋮
  - $n$ )  $\Phi'_{[1,n-1]} = A_{[1]}^{-T} \Phi_{[1,n-1]} A_{[n-1]}^{-1} \longrightarrow A_{[n]}$ .
- 

Here,  $\longrightarrow$  represents constructing the transformation matrix for the next step, after finding out the linear mapping relationship between  $\Phi'_{[1,l]}$  and  $\Phi_{[1,l]}$ . Note that if the initial argument  $A_{[1]}$  represents the pure rotation, then  $A_{[l]}$  is an orthogonal matrix (see [19]), and thus Eq. (22) degenerates to

$$\Phi'_{[l,m]} = A_{[l]} \Phi_{[l,m]} A_{[m]}^T.$$

## A.2 Affine Transformation

By using homogeneous coordinates, an IP of  $n$  variables in Euclidean space can be described in projective space by a corresponding homogeneous IP of  $n+1$  variables. To convert a ternary (*i.e.* 3D) IP of degree  $d$

$$\Psi_{3D}^d(x, y, z) = \sum_{0 \leq i,j,k; i+j+k \leq d} a_{ijk} x^i y^j z^k \quad (25)$$

into its homogeneous representation, a new component  $t = 1$  is added to the 3D IP as:

$$\Psi_{4D}^d(x, y, z, t) = \sum_{0 \leq i,j,k,l; i+j+k+l=d} a_{ijkl} x^i y^j z^k t^l. \quad (26)$$

Therefore a homogeneous polynomial corresponding to a 3D IP of degree  $d$  is a form of degree  $d$  in a 4D IP. Thus the procedure mentioned in the last section can be used to transform the homogeneous IP (4D IP). The different is that the incremental algorithm starts from a  $4 \times 4$  affine transformation matrix  $A_{[1]}$ , e.g., in the Euclidean case

$$A_{[1]} = \begin{pmatrix} R_{3 \times 3} & \mathbf{t} \\ & 1 \end{pmatrix},$$

where  $R_{3 \times 3}$  and  $\mathbf{t}$  are pure 3D rotation matrix and translation vector. As a result, a 3D IP of degree  $d$  can be affine-transformed only by  $A_{[d]}$ , once it is worked out. Furthermore, the incremental scheme can be modified for saving the computational cost, e.g., for calculating  $A_{[9]}$  we can select the incremental order as:

$$A_{[1]} \rightarrow A_{[2]} \rightarrow A_{[4]} \rightarrow A_{[8]} \rightarrow A_{[9]}.$$

## References

- [1] <http://www.atslabs.com>.
- [2] P. Besl and N. McKay. A Method for Registration of 3-D Shapes. *IEEE Trans. on Patt. Anal. Mach. Intell (TPAMI)*, 14(2):239–256, 1992.
- [3] J. Blackall, G. Penney, A. King, and D. Hawkes. Alignment of sparse freehand 3-D ultrasound with preoperative images of the liver using models of respiratory motion and deformation. *Trans. on Medical Imaging*, 24(11):1405–1416, 2005.
- [4] M. Blane, Z. B. Lei, and D. B. Cooper. The 3L Algorithm for Fitting Implicit Polynomial Curves and Surfaces to Data. *IEEE Trans. on Patt. Anal. Mach. Intell (TPAMI)*, 22(3):298–313, 2000.
- [5] F. Catte, P.-L. Lions, J.-M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, 29(1):182–193, 1992.
- [6] Y. Chen and G. Medioni. Surface Description of Complex Objects from Multiple Range Images. *Proc. IEEE Conf. Computer Vision and Patt. Rec. (CVPR)*, pages 153–158, 1994.
- [7] L. Evans. *Partial Differential Equations*. Providence: American Mathematical Society, 1998.
- [8] A. Helzer, M. Barzohar, and D. Malah. Stable Fitting of 2D Curves and 3D Surfaces by Implicit Polynomials. *IEEE Trans. on Patt. Anal. Mach. Intell (TPAMI)*, 26(10):1283–1294, 2004.
- [9] A. Leroy, P. Mozer, Y. Payan, and J. Troccaz. Rigid registration of free-hand 3D ultrasound and CT-scan kidney images. *Proc. MICCAI2004*, 3216:837–844, 2004.
- [10] T. Oishi, A. Nakazawa, R. Kurazume, and K. Ikeuchi. Fast Simultaneous Alignment of Multiple Range Images using Index Images. *Proc. The 5th International Conference on 3-D Digital Imaging and Modeling (3DIM 2005)*, pages 476–483, 2005.
- [11] N. Pagoulatos, S. Member, W. S. Edwards, D. R. Haynor, and Y. Kim. Interactive 3-D Registration of Ultrasound and Magnetic Resonance Images Based on a Magnetic Position Sensor. *IEEE Trans. Information technology in biomedicine*, 3(4):278–288, 1999.
- [12] G. Penney, D. Barratt, C. Chan, M. Slomczykowski, T. Carter, P. Edwards, and D. Hawkes. Cadaver validation of intensity-based ultrasound to CT registration. *Medical Image Analysis*, 10(3):385–395, 2006.
- [13] A. Roche, X. Pennec, G. Malandain, and N. Ayache. Rigid registration of 3-D Ultrasound with MR images: A new approach combining intensity and gradient information. *IEEE Trans Medical Imaging*, 20(10):1038–1049, 2001.
- [14] T. Sahin and M. Unel. Fitting Globally Stabilized Algebraic Surfaces to Range Data. *Proc. IEEE Conf. Int. Conf. on Computer Vision (ICCV)*, 2:1083–1088, 2005.
- [15] J. Salvi, C. Matabosch, D. Fofi, and J. Forest. A review of recent range image registration methods with accuracy evaluation. *Image and Vision Computing*, 25(5):578–596, 2007.
- [16] J.-P. Tarel, H. Civi, and D. B. Cooper. Pose Estimation of Free-Form 3D Objects without Point Matching using Algebraic Surface Models. In *Proceedings of IEEE Workshop Model Based 3D Image Analysis*, pages 13–21, 1998.
- [17] T. Tasdizen, J.-P. Tarel, and D. B. Cooper. Improving the Stability of Algebraic Curves for Applications. *IEEE Trans. on Imag. Proc (TIP)*, 9(3):405–416, 2000.
- [18] G. Taubin. Estimation of Planar Curves, Surfaces and Non-planar Space Curves Defined by Implicit Equations with Applications to Edge and Range Image Segmentation. *IEEE Trans. on Patt. Anal. Mach. Intell (TPAMI)*, 13(11):1115–1138, 1991.
- [19] G. Taubin and D. Cooper. *Symbolic and Numerical Computation for Artificial Intelligence, chapter 6*. Computational Mathematics and Applications. Academic Press, 1992.
- [20] J. W. Trobaugh, W. D. Richard, K. R. Smith, and R. D. Bucholz. Frameless stereotactic ultrasonography: Method and applications. *Computerized Medical Imaging and Graphics*, 18:235–246, 1994.
- [21] W. Wein, B. Roper, and N. Navab. Automatic registration and fusion of ultrasound with CT for radiotherapy. *Proc. MICCAI 2005*, 2:303–311, 2005.
- [22] X. Huang and N.A. Hill and J. Ren and G. Guiraudon and T.M. Peters. Intra-cardiac 2d us to 3d ct image registration. *Proceedings of the SPIE International Symposium on Medical Imaging 2007*, (6509-85), 2007.
- [23] B. Zheng, J. Takamatsu, and K. Ikeuchi. Adaptively Determining Degrees of Implicit Polynomial Curves and Surfaces. *Proc. Conf. Asian Conf. on Computer Vision (ACCV)*, LNCS4844:289–300, 2007.