

パラメタ表現式に基づく三次元形状データの位置・形状合わせ

増田 智仁
東京大学

池内 克史
東京大学

概要

本論文では対象物の三次元形状をより正確に復元するために、回転・平行移動からなる剛体変換による従来の位置合わせ手法について調査し、最も正確な位置合わせ手法を設計する。位置合わせが正しく行われるためには、データノイズやデータの初期位置に対するロバスト性が求められる。実際に実装した位置合わせ手法の正確さを実証するため、従来手法との位置合わせのふるまいを比較し、また推定された剛体変換のパラメタを定量的に評価した。また、剛体変換だけではモデリングや形状比較などを行うのに十分ではないため、変形を伴う位置合わせを考える必要がある。本論文ではロバストな位置合わせを拡張し、剛体変換と形状パラメタを同時に推定する手法とその応用例を提案する。本手法では、変形のメカニズムから厳密に定義できる変形式が得られるものと仮定する。

キーワード: ロバスト位置合わせ, 同時位置合わせ, 形状パラメタ推定

Registration and Deformation of 3D Shape Data through Parameterized Formulation

Tomohito Masuda
University of Tokyo

Katsushi Ikeuchi
University of Tokyo

Abstract

In this paper, we investigate conventional registration implementation, consisting of rotation and translation, to design the most precise registration so as to accurately restore the 3D shape of an object. To achieve the most accurate registration, our registration implementation needs the robustness against data noise, or initial pose and position of data. To verify the accuracy of our implemented registration, we compare the registration behavior with the behavior of conventional registrations, and evaluate the numerical accuracy of transformation parameter obtained by our registration.

However, registration by rigid-body transformation is not enough for modeling and shape comparison: registration when deformation is needed. In this paper, we extend our robust registration to simultaneously estimate the shape parameter as well as the rigid-body transformation parameter. This extension method assumes that the deformation is formulated strictly from the deformation mechanism. We additionally introduce its applications of our extension method.

Keywords: robust registration, simultaneous registration, shape parameter estimation

1 Introduction

Recently, progress has been made in restoring the accurate 3D shapes of objects in the real world using computer graphics. In this research, a laser range sensor is usually used to capture the 3D shape data of an object. However, the shape data is just partial because of the view limitation of the sensor at one scanning. In order to reconstruct the whole shape of the object, therefore, it is necessary to restore the neighboring status of partial data that can compose the whole shape of the object. This restoration process involves registration among 3D data.

Registration among 3D data is usually achieved by rigid-body transformation consisting of translation and rotation. This is implemented by the iterative minimization framework of the squared sum of the distance between closest points among overlapping 3D data of point cloud (Iterative Closest Point, ICP). There are various kinds of implementation according to the purpose of the procedure.

In this paper, we investigate conventional registration implementation to design the most precise registration so as to accurately restore the 3D shape of an object. In our design of registration implementation, the top priority is its accuracy, even if its computation cost could be expensive as far as the computation complexity is within the limit of the current computer platform. To achieve this, our registration implementation needs the robustness against data noise, or initial pose and position of data. To verify the accuracy of our implemented registration, we compare the registration behavior with the behavior of conventional registrations, and evaluate the numerical accuracy of transformation parameter obtained by our registration.

However, registration by rigid-body transformation is not enough for modeling and shape comparison: registration when deformation is needed. In this paper, we extend our robust registration to simultaneously estimate the shape parameter as well as the rigid-body transformation parameter. This extension method assumes that the deformation is formulated strictly from the deformation mechanism. Using this extension framework, we implement a deformation registration to estimate the shape parameter from the shape measurement data of a mathematical plaster model made at the end of the 19th century.

The proposed deformation registration pays attention to the significance of estimated parameter as well as the convergent registration result. To remove the distortion of data obtained by the sensor suspended beneath the balloon (Floating Laser Range Sensor, FLRS), we exploit our deformation registration for the distortion rectification, regarding the movement of FLRS during scanning as shape parameter. In each implementation, we evaluate the accuracy of the estimation of the shape parameter.

2 Related Work

2.1 Iterative Closest Point Algorithm

Automatic registration we consider here needs to give the initial pose and position resulting in the optimal registration. This acquisition of initial pose and position can be achieved by a user through Graphic User Interface (GUI).

However, initial registration through GUI is, at most, the result that the user subjectively and visually regards as the optimal one, so the closest points in this stage might not be the closest points in the optimal registration result. In an ICP algorithm framework, therefore, the point correspondence in between neighbor data sets is taken as the closest point temporarily in the current registration status, and then the registration is gradually improved. These two steps, the point correspondence and registration improvement, are iteratively repeated until the optimal registration is reached [1].

The straightforward quantitative function, which we call "objective function" here, is defined as follows:

$$f(\mathbf{t}, \mathbf{R}) = \sum_i \|\mathbf{R}\mathbf{x}_i + \mathbf{t} - \mathbf{y}_i\|^2, \quad (1)$$

where \mathbf{t} translation vector,
 \mathbf{R} rotation matrix,
 \mathbf{x}_i i th point in the transformed data set,
 \mathbf{y}_i the corresponding point (closest point) of \mathbf{x}_i .

The registration problem is to find the parameter vector \mathbf{t} and \mathbf{R} in this function.

The above equation is formulated just for one pair of data sets, but multiple neighbor data sets are considered in some implementations. In this case, the quantitative function is again shown as follows:

$$f(\mathbf{t}, \mathbf{R}) = \sum_j \sum_i \|\mathbf{R}\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ji}\|^2, \quad (2)$$

where \mathbf{t} translation vector,
 \mathbf{R} rotation matrix,
 \mathbf{x}_i i th point in the transformed data set,
 \mathbf{y}_{ji} the corresponding point (closest point) of \mathbf{x}_i in the j th neighbor data set.

After obtaining their parameter set at each iterative step, \mathbf{x}_i can be updated to \mathbf{x}'_i as follows:

$$\mathbf{x}'_i = \mathbf{R}\mathbf{x}_i + \mathbf{t}. \quad (3)$$

2.2 Registration Strategies

The above ICP algorithm was proposed by Besl and McKay [1], and became the most fundamental framework for 3D data registration. This algorithm framework reduces registration to the minimization problem of the distance sum

between the corresponding data by the iterative calculation. The function minimization with respect to the transformation parameter leads the optimal one which represents the plausible transformation between the aligning data sets, for example, three translation and three rotation parameters in the case of the rigid-body transformation. This framework assumed that two data sets were aligned, and that the shape of a transformed data set was the partial one of the neighbor data set. Currently it is extended in various way in order to handle multiple data sets and to pursue the robustness and the speed of convergence.

We can classify them from the viewpoint of the registration ordering, matching unit, point correspondence, error metric, and outlier elimination.

2.2.1 Registration Ordering

In the registration of multiple sets of 3D data, the ordering affects the convergence of the final result. The sequential ordering chooses a corresponding pair of data sets at each iteration for the registration, and repeats this process until all the data sets are aligned [2] [3] [4] [5]. Its computation cost is lower because only two data sets are handled at each registration. However, it is susceptible to registration failure since the registration errors are locally accumulated and this causes the local discrepancy of the registration result.

In contrast, the simultaneous ordering aligns all the data together at each iteration. Although its computation cost is higher, it enables more accurate registration because the registration error is distributed globally. Consequently, we adopt the simultaneous ordering.

In relationship to the equation (2), registration ordering determines how many data sets \mathbf{y}_{ji} is considered for the registration of the transformed data. In the case of sequential registration, j is unity. Namely, only one neighbor data set is considered for the registration data. Simultaneous registration considers all of the corresponding data \mathbf{y}_{ji} , in which j does vary.

2.2.2 Matching Unit

Matching unit determines the point sampling. The matching unit of the ICP algorithm has two kinds: All-points matching uses all points of a data set. Feature-points matching uses only points satisfied with some condition, for example, only high-curvature points.

Assuming that one-to-one correspondence exists among all the feature points, the feature-points matching usually does not change their correspondence at any iteration [6] [7]. So it cannot achieve the accurate registration in the case in which the correspondence cannot be taken precisely. Even if it changes their correspondence, the feature points are unreliable when the range data has considerable noise, because the feature points are derived by some differential operation.

The all-points matching updates the correspondence so that it can be more plausible as the iteration proceeds [1] [8], and therefore can achieve more accurate registration. Hence, our registration uses all-points matching.

2.2.3 Point Correspondence

Point correspondence determines how the corresponding $\mathbf{x}_i, \mathbf{y}_{ji}$ is chosen in equation (2). There are many implementations in finding corresponding pairs.

As described in Section 2.1, the typical ones are nearest neighbor correspondence [1] [8] and normal direction correspondence [9]. Nearest neighbor correspondence is taken as the nearest pair in Euclidian space.

Normal direction correspondence is taken as the nearest pair in the normal direction of a point, and they are time-consuming. In contrast, laser ray direction correspondence can reduce the computational cost drastically [5] [10] [11]. This correspondence is taken in the direction of a laser ray emitted from the sensor in 3D point measurement. In [11], its search computation mainly depends on the graphics hardware. In the case of normal and laser ray direction correspondence, the correspondence is taken between the point \mathbf{x}_i and the point (\mathbf{y}_{ji}) on the plane hit in the laser direction of the point \mathbf{x}_i . Since the plane is calculated by the differential operation, so lots of wrong correspondences are caused because of the data noise.

Registration accuracy and convergence speed change greatly according to their point correspondence, and Rusinkiewicz et. al. quantitatively evaluate this in [12]. Paying attention to the difference of these convergence characteristics, [4] adopts the hybrid correspondence of nearest point-to-point and point-to-plane. The top priority in our implementation is a registration accuracy, so we employ the nearest neighbor correspondence because the accuracy is guaranteed for the registration of various classes of shape in this correspondence.

2.2.4 Error Metric

The error metric depends on what kind of value \mathbf{x}_i is. Namely, \mathbf{x}_i may represents a position vector, or a color (red, blue, green: RGB) vector associated with the point.

In most implementations, the euclidian distance of the matching point is mainly used [9] [10]. Some other algorithms adopt such additional information as the surface normal and curvature [13], the reflectance (the reflection ratio of the laser ray) [14] and color of the captured point as the error metric [15] in order to make up for the inaccuracy of point coordinate. In our implementation, we use only the euclidian distance.

2.2.5 Outlier Elimination

To cope with outlier, such as data noise and wrong point correspondence in an initial registration, we need to reconsider the objective function. The straightforward function is represented as follows:

$$E(\mathbf{p}) = \sum_{i,j} z_{ij}(\mathbf{p}), \quad (4)$$

where

$$\begin{aligned} \mathbf{p} &= (\mathbf{t}, \mathbf{R}), \\ z_{ij}(\mathbf{p}) &= \|\mathbf{R}\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ji}\|^2. \end{aligned} \quad (5) \quad (6)$$

In this straightforward least-square (LS) objective function, noise leads to an imprecise registration of 3D data, because the exact correspondences between the noisy data are unknown. Any erroneous correspondences must be eliminated before registration, and a thresholding is often used to eliminate such false correspondences [5] [3] [4]. The threshold value can be determined as a fraction of the standard deviation, σ , to the errors in the data [16]. Typically, it is set to greater than or equal to 3σ . This is the simplest method, but it is unreliable method because elimination is affected by the binary classification of the threshold value.

Better outlier elimination can be provided by M-estimation [17] [14], since probability distribution of the error is considered. M-estimation maximizes the probability by minimizing a function of the form

$$E(\mathbf{p}) = \sum_i \rho(z_i(\mathbf{p})),$$

where $\rho(z)$ is an arbitrary function of the errors z_i in the data set. The M-estimator is the maximum-likelihood estimator such that the probability distribution P is equivalent to $E(z_i)$.

We can find the parameters \mathbf{p} that minimize E by taking the derivative of E with respect to \mathbf{p} and setting the derivative to 0.

$$\frac{\partial E}{\partial \mathbf{p}} = \sum_i \frac{\partial \rho}{\partial z_i} \cdot \frac{\partial z_i}{\partial \mathbf{p}} = \sum_i w(z_i) z_i \frac{\partial z_i}{\partial \mathbf{p}} = 0, \quad (7)$$

$$\text{where } w(z) = \frac{1}{z} \frac{\partial \rho}{\partial z}.$$

This equation shows that the weight is added to the straightforward least-square objective function.

A Lorentz function is used as the M-estimator; a Lorentz function can be represented as:

$$\rho(z_i(\mathbf{p})) = \log \left(1 + \frac{1}{\sigma^2} z_i(\mathbf{p}) \right). \quad (8)$$

In practice, the weight is imposed at the differential operation stage, as follows:

$$\frac{\partial \rho}{\partial \mathbf{p}} = \frac{\partial \rho}{\partial z} \cdot \frac{\partial z}{\partial \mathbf{p}} = \frac{1}{2\sigma^2 + z} \cdot \frac{\partial z}{\partial \mathbf{p}}. \quad (9)$$

Wheeler summarized the registration behavior according to the probability distribution in M-estimator in [18].

2.3 Deformation Registration

In this paper, we propose the extended framework of the conventional registration algorithm to allow the shape deformation during registration process. This kind of registration, namely, deformation registration, has been researched in such field as the medical imaging, and the target object for the registration is mainly soft tissues. They adopt similarity [19], affine [13], geometric hashing [20], quadric/superquadric [21], and displacement-field-based transformation [22] so that their deformation works well for any kind of target shape.

These methods can be generally adopted in shape modeling and fitting. However, if the deformation is strictly defined by some parameterized formulation derived from the deformation mechanism, the deformation is much more accurate when using its formulation than when derived from their methods. The parameters obtained from our strict formulation carry with them the essential information about the cause and origination of the deformation. So our framework pays as much attention to the obtained parameters as to the appearance resulting from the deformation. In this point, our aim is different from theirs. So in our assumption that the shape changes are strictly represented with a mathematical formula including some variable parameters and its formula is known a priori, we formulate the generally extended registration which allows the 3D data to be deformed and determines both the deformation and the translation and rotation parameters.

3 Robust Determination of Translation and Rotation Parameters

3.1 Robust Simultaneous Registration Algorithm

Based on the previous section, here we explain the details of our designed registration. As a preprocess, multiple data sets are initially aligned. In iterative process, the followings are done:

- Constructing kd-trees of data sets.
- Searching nearest neighbors using kd-trees.
- Minimizing the objective function (squared sum of nearest neighbor distance) to find the better (optimal) registration parameter.
- Updating data sets according to the obtained registration parameter.

The above process is repeated until the optimal registration is reached.

This kind of algorithm is usually time-consuming, and most of the computation cost depends on searching the corresponding point. We have already proposed the effective

kd-tree algorithm for nearest neighbor search [23], and we adopt it in our designed registration.

3.2 Minimization of Objective Function for Parameter Estimation

3.2.1 Derivation of Descent Gradient

Our registration algorithm aligns all data sets simultaneously so as to minimize the squared sum of nearest neighbor point-to-point distances. The objective function is represented as follows:

$$E(\mathbf{p}) = \sum_i \sum_j \rho(z_{ij}(\mathbf{p})), \quad (10)$$

where

$$\mathbf{p} = (\mathbf{t}, \mathbf{q}), \quad (11)$$

$$z_{ij}(\mathbf{p}) = \|\mathbf{R}(\mathbf{q})\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ji}\|^2, \quad (12)$$

$$\rho(z_{ij}(\mathbf{p})) = \log\left(1 + \frac{1}{2}z_{ij}(\mathbf{p})\right), \quad (13)$$

- \mathbf{t} : translation vector,
- $\mathbf{R}(\mathbf{q})$: rotation matrix corresponding to quaternion q ,
- \mathbf{x}_i : i th point in the data set of interest,
- \mathbf{y}_{ji} : the corresponding point of x_i in the j th measured data.

As for its rotation matrix, we use a quaternion representation of 3 Degrees Of Freedom (DOF).

Using error metric $E(\mathbf{p})$, we compute the parameters \mathbf{p} which fulfill the following equation:

$$\mathbf{p}_{opt} = \arg \min_{\mathbf{p}} E(\mathbf{p}). \quad (14)$$

For the gradient-based solution of our non-linear optimization, the descent gradient is:

$$\begin{aligned} \frac{\partial E}{\partial \mathbf{p}} &= \sum_i \sum_j \frac{\partial \rho(z_{ij})}{\partial z_{ij}} \cdot \frac{\partial z_{ij}}{\partial \mathbf{p}} \\ &= \sum_i \sum_j w(z_{ij}) z_{ij} \frac{\partial z_{ij}}{\partial \mathbf{p}}, \end{aligned} \quad (15)$$

$$\text{where } w(z_{ij}) = \frac{1}{z_{ij}} \cdot \frac{\partial \rho(z_{ij})}{\partial z_{ij}}.$$

If we evaluate $\partial z_{ij} / \partial \mathbf{p}$ by an identity quaternion \mathbf{q}_I , we can represent $\partial z_{ij} / \partial \mathbf{p}$ as

$$\frac{\partial z_{ij}(\mathbf{p})}{\partial \mathbf{p}} = 2(\mathbf{R}(\mathbf{q})\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ji}) \frac{\partial (\mathbf{R}(\mathbf{q})\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ji})}{\partial \mathbf{p}} \Big|_{\mathbf{q}_I} \quad (16)$$

$$= \begin{bmatrix} 2(\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ji}) \\ 4\mathbf{C}(\mathbf{x}_i)^T (\mathbf{x}_i \mathbf{t} - \mathbf{y}_{ji}) \end{bmatrix} \quad (16)$$

$$= \begin{bmatrix} 2(\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ji}) \\ 4\mathbf{x}_i \times (\mathbf{t} - \mathbf{y}_{ji}) \end{bmatrix}, \quad (17)$$

because the (negative) gradient of quaternion at an identity quaternion \mathbf{q}_I is obtained by equation (18).

$$\frac{\partial (\mathbf{R}(\mathbf{q})\mathbf{x}_i)}{\partial \mathbf{q}} \Big|_{\mathbf{q}_I} = 2\mathbf{C}(\mathbf{x}_i)^T. \quad (18)$$

From the obtained descent gradient, the conjugate gradient is calculated so that all the obtained gradient is guaranteed to be orthogonal. Transformation vector \mathbf{p} is acquired using the conjugate gradient [24] [25] [26] and line minimization method with a combination of golden ratio bracketing (golden section search) and parabolic fits.

3.3 Evaluation

In this section, we quantify the effectiveness of our robust registration on the basis of four issues by comparing previous registration methods. First, we argue the merits of adopting a simultaneous strategy. Second, we discuss the effectiveness of using stochastic outlier elimination to increase the robustness of the technique. Third, to evaluate the effectiveness of these two steps, we evaluate the overall estimation accuracy of our registration.

3.3.1 Simultaneous vs. Sequential Ordering

In this evaluation, we align seven partial data sets of the Fugoppe Cave in simultaneous and sequential strategies. The upper figure in Figure 1 shows the initial state of these data sets. They are slightly shifted among the overlapping data. The middle and lower figures in Figure 1 respectively show the registration result in simultaneous vs. sequential ordering. The sequential registration we used is basically the implementation proposed by [2]. In order to observe only the effect of simultaneous and sequential strategies, however, this sequential registration uses M-estimator for outlier elimination.

In sequential registration, we must determine the data pairing of alignment targets such that they are exactly overlapping each other. Sequential registration considers pairing only two data sets at a time, and assumes that the registration works well among each pair of data sets. So if the transformation is determined in one data set, it is transformed together with the rest of the data.

Good registration is visible as evenly mottled pattern in overlapping area because of slight differences in sampling and because of the random noise, even if the area has an identical shape.

Comparing simultaneous registration methods, sequential registration introduces local discrepancies (between the yellow and cyan-blue data sets, for example). The detailed observation is shown in Figure 2. In the lower figures, the green color indicates areas of little difference (less than 1 [cm]), while the red and blue colors indicate areas of larger

difference (more than 1 [cm]). Simultaneous registration results in almost no difference, by comparison; simultaneous registration is clearly better.

3.3.2 Straightforward Least Square Registration vs. Robust Registration

In this investigation, we align the data sets of two ancient mirrors that were cast from the same mold. They have local differences in their shapes.

The initial pose and position between them is shown in Figure 3. They are aligned using straightforward LS registration as well as robust M-estimator registration. The former is the registration proposed by [16], and the latter is our implemented registration. The registration result is shown in Figure 4. Figures in the first row show the convergence result.

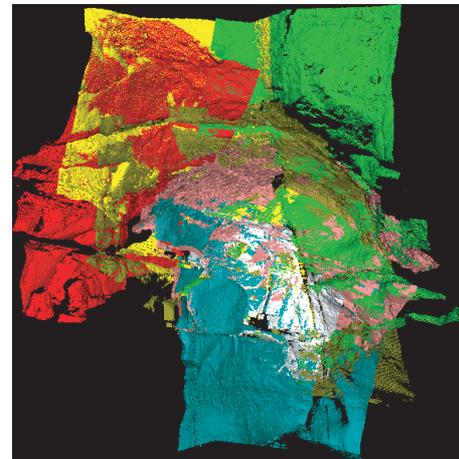
Figures in the second and third row, respectively, show the convex and concave areas of one mirror vs. the other when the length between each corresponding point is exceeded by the setting threshold. This threshold is respectively set to 0.5 and 0.25 [mm] in the second and third row. In the second row, the upper circular area has more concave area when using an M-estimator, but the lower area has more convex area otherwise. Similarly in the third row, the left area has more convex area in the M-estimator result, but the right area has more convex area otherwise. As shown in the numerical results, the green area, regarded as an area of no difference in shape, is 51.7 and 49.6 percent of the total in the middle, and is 77.0 and 77.3 percent in the lowest, respectively, in the case for which the thresholds are 0.25 and 0.5. This result shows that the outlier area is automatically recognized and ignored in the registration process in order to align as much area as possible.

3.3.3 Estimation Accuracy of Translation and Rotation parameters

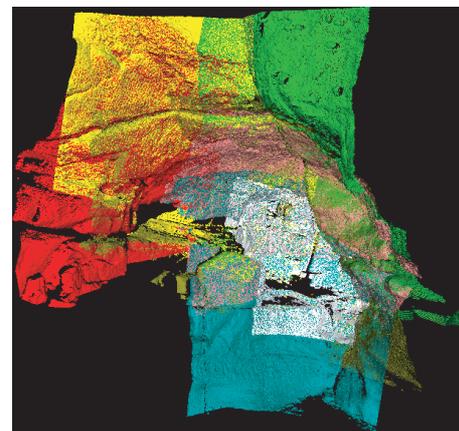
In this investigation, we align two data sets capturing the face of a tower at Bayon ruin in Cambodia (Figure 5-(a)). To consider the registration of the actual measurement data, we created two data sets from the same measurement data by sub-sampling the different points. Figure 5-(b) shows the appearance of two superimposed data, which is regarded as the correct registration between them.

To create the initial position states of two data sets – original and transformed data sets – the transformed data set is translated and rotated, then it is realigned to the original data. The estimation accuracy of the registration parameter is regarded as the difference between the amount of the translation and rotation of the transformed data set in its initial state and that of the registration result.

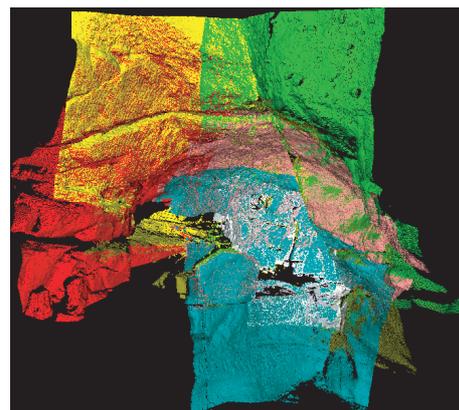
The initial position of transformed data set is set to three steps in translation and rotation respectively. It is translated to ± 0.5 [m] in each axis, and is rotated to ± 30 [deg] around



Initial State



Simultaneous Strategy



Sequential Strategy

Figure 1: Registration results in simultaneous and sequential strategies. The upper figure shows the initial state of partial 3D data. The middle and lower figures respectively show the registration results in simultaneous and sequential strategies.

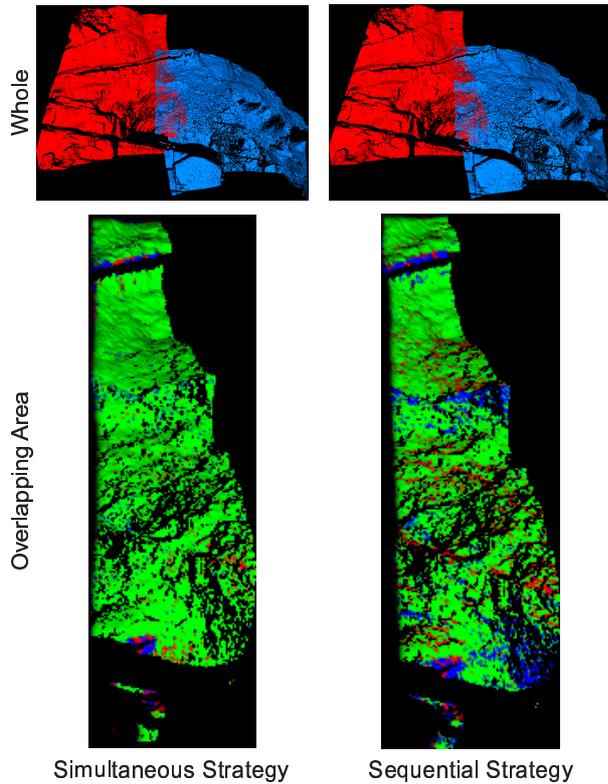


Figure 2: Detailed observation of registration results in simultaneous and sequential strategies. The upper figures show the whole appearance of registration results in simultaneous (left) and sequential (right) strategies. The lower figures show the detail of the overlapping area between them. The green shows no difference (less than 1 [cm]), while the red and blue show larger differences (more than 1 [cm]).

all the direction that can be represented as the combination of -1, 0, and 1 in each axis. As a result, the number of translation settings is 26 ($27 (3^3)$ minus 1 (to remove the trivial translation (0, 0, 0))). In rotation, the number of rotation axes is 26 ($27 (3^3)$ minus 1 (likewise, to remove the “rotation” axis (0, 0, 0))), but half of these axes are symmetrical with respect to the coordinate origin (For example, (-1, 1, 1) and (1, -1, -1)), so the actual number of rotation axes is 13. In each axis, transformed data set is rotated 30 [deg] in clockwise and counter-clockwise directions, so the number of rotation settings is 26. Combining translations and rotations, we have 676 cases (26×26).

As a numerical result, we show twelve parameter sets which are considered typical of all the estimation results, in Figure 6. In this figure, “x-t 0.5” means 0.5 [m] translation along x axis, and “x-r 30” means 30 [deg] rotation around x axis. When the initial translation and rotation is set as shown in the translation and rotation axes (e.g. x-t

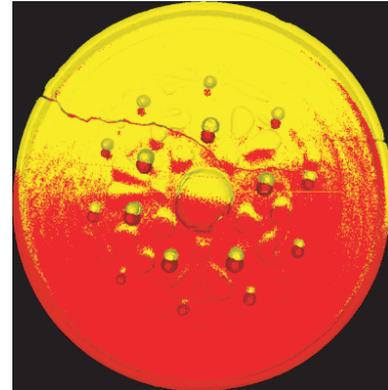


Figure 3: Initial pose and position between two mirrors. In this figure, The yellow mirror is slightly translated and rotated against the red. (Data Informant: Kashihara Institute of Archaeology and Tokyo National Museum.)

0.5, x-t -0.5, ..., z-t -0.5, x-r 30, x-r -30, ..., z-r -30), the difference between each true initial parameter and the corresponding estimated parameter with respect to each axis (Translation along x axis, Translation along y axis, Translation z axis, Rotation around estimated axis) is shown on the vertical axis (e.g. 0, ± 0.1 , ...). This figure shows the translation and rotation estimation errors of our registration is respectively within 0.05 [m] and 0.5[deg].

Of our 728 cases, 694 result in good registration. Because we can easily observe large position differences in these 34 cases, our registration seldom fails if the initial position estimate is manually improved.

In addition, we investigate the result of two implementations proposed by [11] (laser ray direction, point-to-plane correspondence, thresholding) and [16] (nearest neighbor, point-to-point, thresholding) by aligning 728 pairs of data set in the same condition as the above. In the first registration method [11], 483 result in good registration. Observing the registration process, the convergence of this registration looks slow until the optimal registration is acquired. And in the second registration method [16], 715 result in good registration. These pairs are completely superimposed in all area each other, so it looks preferable not to employ the operation for outlier elimination. To verify this, we create and align the partial shape data sets as shown in Figure 7. The initial setting of translation and rotation is the same as the above. Then in our registration, 343 result in good registration, while 335 result in good registration in the registration [16]. Though our initial setting is rough in this evaluation, our implementation can prove to be more robust than [16], if the detail evaluation is done.

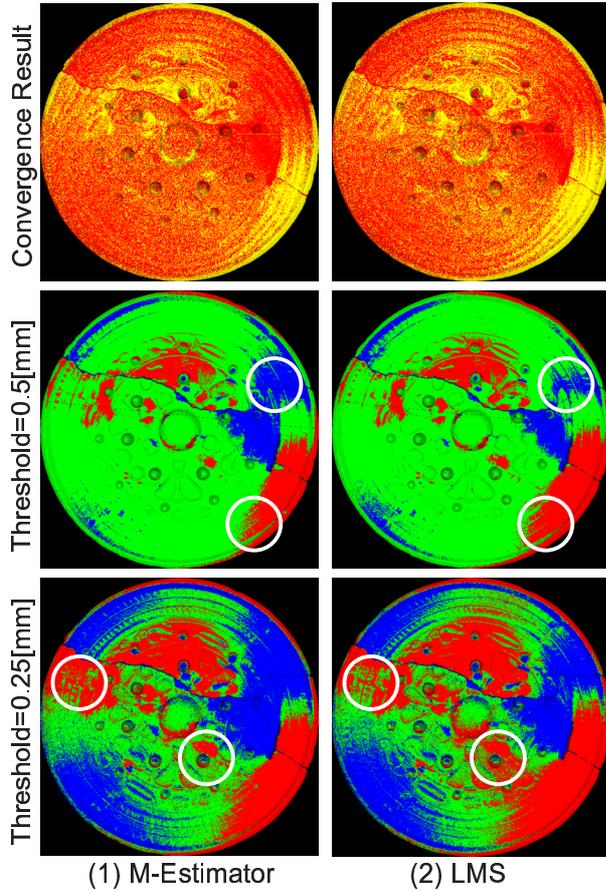


Figure 4: Convergence results of two mirrors. Figures in the first row show the convergence result, and figures in the second and third row show the convex and concave areas of one mirror against the other when the length between each corresponding point is exceeded by the setting threshold, regarded as shape difference. (Data Informant: Kashihara Institute of Archaeology and Tokyo National Museum.)

4 Extension of Rigid-body Transformation

In this section, we first generally extend the rigid-body transformation to allow deformation during a registration. Therefore, estimated parameters include those which affect their shape in addition to six parameters of the pose and position in a conventional registration. In later sections, we adopt this extended framework to solve each problem.

4.1 Simultaneous Determination of Registration and Deformation Parameters

Our proposal assumes that the deformation can be represented by a parameterized mathematical formula whose

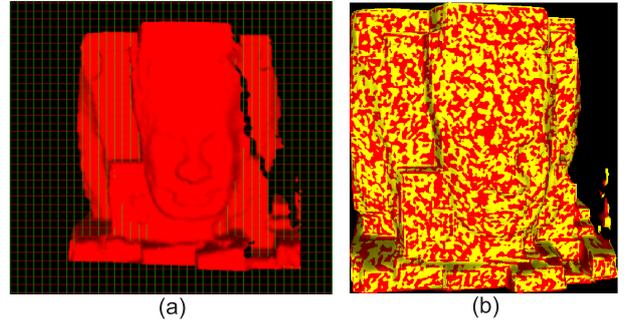


Figure 5: Data used in this evaluation. In (a) in this figure, the size of each unit square is 0.1 by 0.1 [m]. Here, positive axes of x-axis and y-axis are respectively set to the right and upper direction, and the positive direction of z-axis is set to the front direction, perpendicular to this figure.

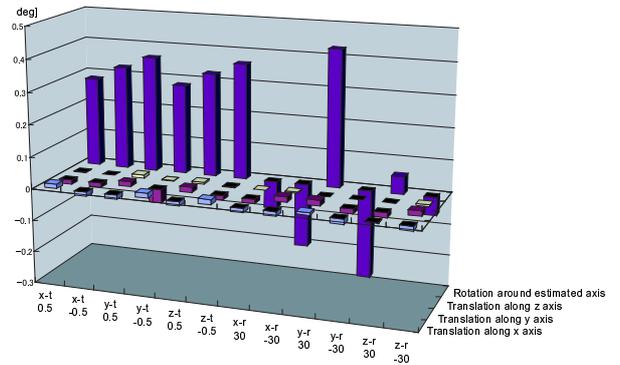


Figure 6: Estimation errors in translation and rotation for each initial position.

form is known a priori, but whose parameters are unknown.

Our goal is to simultaneously determine these deformation, translation, and rotation parameters by comparing the target data to transform with its corresponding data. We do this using an ICP framework: translation and rotation parameters are determined in a minimization paradigm. If we fix the translation and rotation parameters, determination of the deformation parameter becomes an iterative shape matching problem. Thus, we can handle all parameter determinations in a unified minimization framework.

We extend the parameter estimation of the registration formulation to add the shape parameter by extending the objective function in equation (12). Therefore, $z_{ij}(\mathbf{p})$ in equation (12) is transformed into:

$$z_{ij}(\mathbf{p}) = \|\mathbf{R}(\mathbf{q})\mathbf{g}(\mathbf{x}_i, \mathbf{k}) + \mathbf{t} - \mathbf{y}_{jil}\|^2, \quad (19)$$

where $\mathbf{p} = (\mathbf{t}, \mathbf{q}, \mathbf{k})$,
 $\mathbf{g}(\mathbf{x}_i, \mathbf{k})$: deformation function of point \mathbf{x}_i with respect to parameter \mathbf{k} .

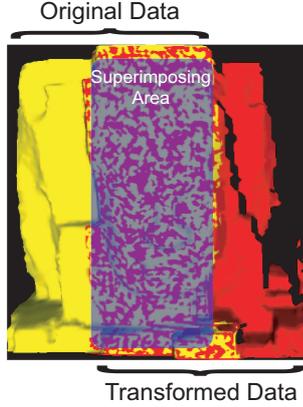


Figure 7: Partial data used to compare the registration result of ours and [16]

Our rigid-body registration is designed to be robust, and here we adopt the same strategy as in Section 3. In this extended framework, we consider the registration of multiple data sets. The algorithm in this framework is the same as shown in Section 3.1.

In the ICP based registration algorithm, the acquisition of the valid initial parameter is important for the optimal registration result. In our implementation, the initial transformation parameter is set manually, using GUI, with accuracy good enough to reach a true optimum.

4.2 Minimization of Objective Function for Parameter Estimation

Summing up our strategy in this framework as a numerical formulation, the minimization function is as follows:

$$E(\mathbf{p}) = \sum_i \sum_j \rho(z_{ij}(\mathbf{p})), \quad (20)$$

where

$$\mathbf{p} = (\mathbf{t}, \mathbf{q}, \mathbf{k}), \quad (21)$$

$$z_{ij}(\mathbf{p}) = \|\mathbf{R}(\mathbf{q})\mathbf{g}(\mathbf{x}_i, \mathbf{k}) + \mathbf{t} - \mathbf{y}_{ji}\|^2, \quad (22)$$

$$\rho(z_{ij}(\mathbf{p})) = \log\left(1 + \frac{1}{2}z_{ij}(\mathbf{p})\right), \quad (23)$$

- \mathbf{t} : translation vector,
- $\mathbf{R}(\mathbf{q})$: rotation matrix corresponding to quaternion q ,
- \mathbf{k} : shape parameter,
- \mathbf{x}_i : i th point in the data set of interest,
- $\mathbf{g}(\mathbf{x}_i, \mathbf{k})$: deformation function of point \mathbf{x}_i with respect to parameter \mathbf{k} ,
- \mathbf{y}_{ji} : the corresponding point of x_i in the j th measured data.

Again, we compute the parameter \mathbf{p} to satisfy the following equation:

$$\mathbf{p}_{opt} = \arg \min_{\mathbf{p}} E(\mathbf{p}). \quad (24)$$

To calculate \mathbf{p} , we also use the gradient-based solution. The descent gradient is computed as follows:

$$\begin{aligned} \frac{\partial E}{\partial \mathbf{p}} &= \sum_i \sum_j \frac{\partial \rho(z_{ij})}{\partial z_{ij}} \cdot \frac{\partial z_{ij}}{\partial \mathbf{p}} \\ &= \sum_i \sum_j w(z_{ij}) z_{ij} \frac{\partial z_{ij}}{\partial \mathbf{p}}, \end{aligned} \quad (25)$$

$$\text{where } w(z_{ij}) = \frac{1}{z_{ij}} \cdot \frac{\partial \rho(z_{ij})}{\partial z_{ij}},$$

$$\frac{\partial z_{ij}(\mathbf{p})}{\partial \mathbf{p}} = 2(\mathbf{R}(\mathbf{q})\mathbf{g}(\mathbf{x}_i, \mathbf{k}) + \mathbf{t} - \mathbf{y}_{ji}) \frac{\partial (\mathbf{R}(\mathbf{q})\mathbf{g}(\mathbf{x}_i, \mathbf{k}) + \mathbf{t} - \mathbf{y}_{ji})}{\partial \mathbf{p}} \Bigg|_{q_i}. \quad (26)$$

Using this descent gradient, the conjugate gradient is adopted similarly in Section 3.

5 Shape Parameter Estimation of Mathematical Model

5.1 Mathematical Model : Revolution Surface of Catenary

As a main topic in this chapter, we estimate the shape parameter of certain mathematical model made of plaster in order to examine its manufacturing accuracy (Figure 8). This model is a cultural asset; it was manufactured in Germany at the end of the 19th century for educational purposes. It has been displayed in our university museum.

This object has no documentation, and we are interested in identifying the shape parameters the makers used in manufacturing it. We wish to estimate deformation parameters by applying our extended registration framework algorithm to both measured data sets and the data set computed by mathematical formula, in order to evaluate the manufacturing accuracy of the plaster model.

Using our estimated parameters, we also wish to remake more accurate model for comparison, because both historians and the mathematicians are interested in the level of manufacturing skill extant in those days. Our target is the model that is called “revolution surface of catenary”.

5.2 Mathematical Formula and Experimental Result

The surface generated by rotating a 2D catenary is shown in Figure 8-(1). Such a surface always has azimuthal symmetry. Besides scale parameter (l), there are two parameters

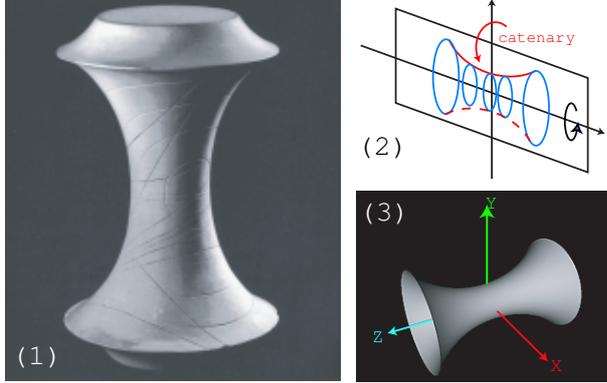


Figure 8: (1) A Mathematical model and (2) its ideal representation used in our experiment. It has the constant negative curvature on all points of their surface. (Data Informant: Prof. Toshitake Kohno (Graduate School of Mathematical Sciences, The University of Tokyo.))

(a, b) involved in the generation of such surfaces. The numerical formula is as follows:

$$\text{shape parameter } \mathbf{k} = (a, b, l) \quad (0 < b \leq a),$$

$$\mathbf{g}(\mathbf{x}_i, \mathbf{k}) = (l\phi(v) \cos u, l\phi(v) \sin u, l\psi(v)), \quad (27)$$

$$\text{where } 0 \leq u < 2\pi, \quad -a \cdot \sinh^{-1}\left(\frac{a}{b}\right) \leq v \leq a \cdot \sinh^{-1}\left(\frac{a}{b}\right),$$

$$\phi(v) = b \cosh\left(\frac{v}{a}\right), \quad \psi(v) = \int_0^v \sqrt{1 - \frac{b^2}{a^2} \sinh^2\left(\frac{t}{a}\right)} dt. \quad (28)$$

In the following, let

$$f(a, b, t) = \sqrt{1 - \frac{b^2}{a^2} \sinh^2\left(\frac{t}{a}\right)}. \quad (29)$$

Therefore,

$$\psi(v) = \int_0^v f(a, b, t) dt. \quad (30)$$

In a rigid-body transformation as shown in equation (3), point of interest \mathbf{x}_i is updated to \mathbf{x}'_i , and \mathbf{x}'_i to \mathbf{x}''_i at the next step, according to estimated parameters as follows:

$$\mathbf{x}'_i = \mathbf{R}\mathbf{x}_i + \mathbf{t}, \quad (31)$$

$$\mathbf{x}''_i = \mathbf{R}'\mathbf{x}'_i + \mathbf{t}'. \quad (32)$$

In the pattern of the target model, the calculated data are determined only by a shape parameter. In other words, $\mathbf{g}(\mathbf{x}_i, \mathbf{k})$ is actually $\mathbf{g}(\mathbf{k})$: \mathbf{x}_i is not needed. In this case, updating is usually performed as follows:

$$\mathbf{x}'_i = \mathbf{R}\mathbf{g}(\mathbf{x}_i, \mathbf{k}) + \mathbf{t}, \quad (33)$$

$$\mathbf{x}''_i = \mathbf{R}'\mathbf{R}\mathbf{g}(\mathbf{x}_i, \mathbf{k} + \mathbf{k}') + \mathbf{t} + \mathbf{t}'. \quad (34)$$

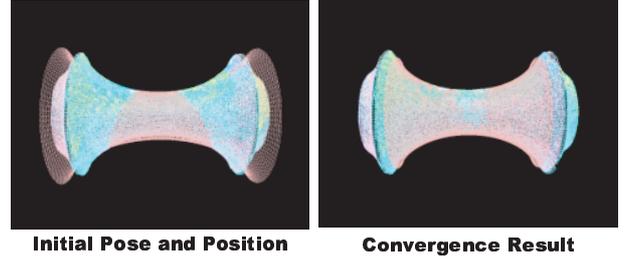


Figure 9: The initial state and result of the parametric data.

In the case of updating pattern as shown in equation (32), the descent gradient is obtained by evaluating the rigid-body rotation at an identity quaternion (\mathbf{q}_I) for all parameters. This numerical representation is as follows:

$$\frac{\partial z_{ij}(\mathbf{p})}{\partial \mathbf{p}} = \begin{bmatrix} 2(\mathbf{g}(\mathbf{x}_i, \mathbf{k}) + \mathbf{t} - \mathbf{y}_{ji}) \\ -4\mathbf{g}(\mathbf{x}_i, \mathbf{k}) \times (\mathbf{t} - \mathbf{y}_{ji}) \\ 2(\mathbf{g}(\mathbf{x}_i, \mathbf{k}) + \mathbf{t} - \mathbf{y}_{ji}) \frac{\partial(\mathbf{g}(\mathbf{x}_i, \mathbf{k}))}{\partial \mathbf{k}} \end{bmatrix}. \quad (35)$$

But otherwise, the updated rigid-body rotation is not reflected if it is evaluated at \mathbf{q}_I for all parameters. So we evaluate the rigid-body rotation at \mathbf{q}_i just for the rigid-body rotation parameter as shown in the following:

$$\frac{\partial z_{ij}(\mathbf{p})}{\partial \mathbf{p}} = \begin{bmatrix} 2(\mathbf{R}\mathbf{g}(\mathbf{x}_i, \mathbf{k}) + \mathbf{t} - \mathbf{y}_{ji}) \\ -4\mathbf{g}(\mathbf{x}_i, \mathbf{k}) \times (\mathbf{t} - \mathbf{y}_{ji}) \\ 2(\mathbf{R}\mathbf{g}(\mathbf{x}_i, \mathbf{k}) + \mathbf{t} - \mathbf{y}_{ji}) \mathbf{R} \frac{\partial(\mathbf{g}(\mathbf{x}_i, \mathbf{k}))}{\partial \mathbf{k}} \end{bmatrix}. \quad (36)$$

The 3D shape of the plaster model was captured using a VIVID 900 (KonicaMinolta) range finder. The data sets were initially aligned using a manual process via Graphic User Interface (GUI). Initial shape parameter was also manually estimated. Figure 9 shows the registration result, and it was well-behaved and convergent. The shape parameters were estimated as follows:

$$a = 0.0568, \quad b = 0.0237, \quad l = 0.996.$$

5.3 Evaluation

Our estimation is affected by various kinds of errors: range data measurement errors; initial registration errors; and the errors in the manually input initial shape parameter. We have already reported how the accuracy of our estimated parameter depended on such errors by using synthesized data computed using known parameters and adding Gaussian noise [27].

Additionally, here we investigated the combined effects of an initial translation, rotation, and specified shape parameter. The initial shape parameter (a, b, l) of the calculated data was set to five steps around each truth value. In



Figure 10: Reproduced metallic mathematical model.

particular, a , b , and l were set to 0.03, 0.04, 0.05, 0.06, 0.07, to 0.01, 0.015, 0.02, 0.025, 0.03, and to 0.7, 0.85, 1.0, 1.15, 1.3. Initial translation and rotation were exclusively set to three steps as follows: translation to 0.01, 0.02, and 0.03 [m] along x and z axes, and rotations of 10, 20, and 30 [deg] around the x axis. This results in 124 deformation cases ($125 (5^3)$ minus 1 (0.05, 0.02, 1.0: the truth value)). There are 9 translation and rotation cases, so there are 1116 (124×9) cases to investigate. Altogether, therefore, we investigated 1249 ($124 + 9 + 1116$) cases.

Of these 1249 cases, 991 result in the correct registration. Judging from these results, a registration tends to fail if there is too much difference between the initial and truth values. These data sets are obviously different in their shape and position; these differences might be easy to cancel because the user can immediately recognize a deficiency and re-run the algorithm after improving the initial shape parameter and position estimates.

5.4 Reproduction of Mathematical Model

Using our algorithm of shape parameter estimation, another mathematical model of Dini's Surface was reproduced in metal by Yamada Seiki Co.,ltd. [28] under the supervision of an artist, Mr. Hiroshi Sugimoto [29]. Yamada Seiki Co.,ltd successfully generated the 3D Shape of the original model with high accuracy (Figure 10), and Mr. Sugimoto held an exhibition of the work at the Mori Art Museum at Roppongi Hills.

In this way, our algorithm can create CAD (computer-aided design) primitives and compressed 3D shape data faithful to the original shape, and as a result, we can refine or alter the shape as desired.

6 Registration for Range Data Obtained by Floating Laser Range Sensor

6.1 Floating Sensing System

To obtain 3D measurement data for large objects, a laser range sensor (LRS) mounted on a tripod is often used. Unfortunately, it often happens that some part of a large object is invisible from the ground. In order to scan these invisible faces, a scaffold might be built nearby. However, this involves time and expense, and moreover, some surfaces might still not be visible due to space limitations for this scaffolding, lack of a viable superstructure, and so forth.

We have developed a novel 3D measurement system [30]. Our system digitizes objects from the air while being suspended beneath a balloon. Although our system is free from high frequency vibration like that caused by helicopter engines, there still remains low frequency movement due to the floating balloon which distorts the data. However, this movement can be modeled as simple trajectory by regarding the movement as the swing of a pendulum.

Our system consists of two main processes: scanning and registration. For the 3D scanning of visible surfaces from the ground, we use an LRS mounted on a tripod on the ground, as usual. To scan facets invisible from the ground, such as the rooftop of a building, we have developed and tested a Floating Laser Range Sensor (FLRS). The FLRS data contains distortion caused by the swing motion of the balloon during scanning, but our extended registration framework can be applied to remove this to rectify the data.

6.2 Floating Laser Range Sensor

Our FLRS system consists of a scanner unit, a controller and a personal computer (PC). These three units are suspended below a balloon.

Our scanner unit includes a laser range finder especially designed to be hung from a balloon. Our design requirements were that the unit be compact and lightweight enough to be carried by a balloon, and that it be fast enough to minimize the influence of the balloon's normal swing.

The scanner unit includes a spot laser radar unit and two mirrors. We chose to use the LARA25200 supplied by Z+F Inc. as a laser radar unit because of its high sampling rate (maximum 625,000 [points/sec]).

6.3 Inter- and Intra-Scanning Registration

6.3.1 Assumption and Formulation of FLRS Motion

In order to align data sets from the FLRS, we distinguish between two different types of movement, "inter-scanning"

and ‘‘intra-scanning’’. Inter-scanning movements provide different views of a scene, and are equivalent to a series of rigid-body transformations. But the FLRS moves during the acquisition of each range data set; this intra-scanning movement of the sensor distorts the measurement data. Our extended registration framework enables the rectification of this distortion; we can represent this motion as a deformation parameter.

The motion of FLRS during scanning depends on the following:

- Its initial velocity
- Its initial angular velocity
- Any acceleration generated by external force
- Any angular acceleration generated by external moments

We can ignore the influence of translation and angular accelerations because our FLRS needs only one second to scan a frame. Therefore, we consider FLRS movement to have constant velocity in translation and rotation, without changing its rotation axis during a frame. Under this assumption we set up the deformation equation in equation (19).

Figure 11 shows positional relationship in intra-scanning registration. Here, \mathbf{O}_u means the origin of the camera coordinate system for the case in which FLRS does not move during a scan, and \mathbf{O}_d means the origin of the camera coordinate system at the time τ_i . (Note that $0 \leq \tau_i \leq 1$; one measurement can require up to a second.)

Assuming that the FLRS moves during scanning, τ_i is the time elapsed since the first point was captured. Then FLRS acquires a 3D point \mathbf{a}_u in the camera coordinate system u (i.e., \mathbf{a}_d in the camera coordinate system d). Because the measurement point is actually recorded in the camera coordinate system u , the translation vector and rotation matrix from the coordinate system u to the coordinate system d at time τ_i are $\mathbf{v}_u\tau_i$ and $\mathbf{R}_{d \leftarrow u}(\tau_i)$, respectively.

If the corresponding point of \mathbf{a}_u is \mathbf{b}_u in the world coordinate system w , then the error in this registration can be represented as

$$z_i = \|\mathbf{R}_{w \leftarrow u}\mathbf{a}_u + \mathbf{T}_u - \mathbf{b}_u\|^2. \quad (37)$$

If the FLRS moves during acquisition, the measurement point is captured from the origin of the camera coordinate \mathbf{O}_d at the measurement time τ_i . Therefore in the coordinate system u ,

$$\mathbf{a}_u = \Delta\mathbf{R}_{d \leftarrow u}^{-1}(\tau_i)\mathbf{a}_d + \mathbf{v}_u\tau_i. \quad (38)$$

Notice that $\mathbf{R}_{d \leftarrow u}^{-1}(\tau_i)$ is equal to $\mathbf{R}_{u \leftarrow d}(\tau_i)$.

Substituting equation (38) for equation (37),

$$z_i = \|\mathbf{R}_{w \leftarrow u}(\Delta\mathbf{R}_{d \leftarrow u}^{-1}(\tau_i)\mathbf{a}_d + \mathbf{v}_u\tau_i) + \mathbf{T}_u - \mathbf{b}_u\|^2. \quad (39)$$

In this case, the geometric function $\mathbf{g}(\mathbf{x}_i, \mathbf{k})$ is represented as follows:

$$\mathbf{g}(\mathbf{x}_i, \mathbf{p}_{intra}) = \Delta\mathbf{R}_{d \leftarrow u}^{-1}(\tau_i)\mathbf{x}_i + \mathbf{v}\tau_i, \quad (40)$$

where \mathbf{p}_{intra} include the state of the intra-rotation axis, its angular velocity, and the intra-translation (\mathbf{v}).

In addition to the parameters of the rigid-body transformation $\mathbf{R}_{w \leftarrow u}$, \mathbf{T}_u , we have to estimate the deformation parameter of $\Delta\mathbf{R}_{d \leftarrow u}^{-1}(\tau_i)$, \mathbf{v}_u .

Intra-rotation is represented by the description of the rotation axis and angular velocity, but these parameters cannot be obtained in the same way as the rigid-body rotation solution which involved a quaternion derivative. In case of rigid-body rotation, the rotation axis description is first calculated, and then the amount of rotation around this calculated axis can be determined by the quaternion normalization. This rigid-body rotation is common to the whole data. But intra-rotation does change with the time τ_i at each i -th point, namely, it must be represented as a function with respect to τ_i .

To remedy this problem, we represent $\Delta\mathbf{R}_{d \leftarrow u}^{-1}(\tau_i)$, by allowing \mathbf{m} and ω be the rotation axis and angular velocity respectively, as follows:

$$\Delta\mathbf{R}_{d \leftarrow u}^{-1}(\tau_i) = \Delta\mathbf{R}_{d \leftarrow u}^{-1}(\mathbf{m}, \omega\tau_i) = \begin{bmatrix} (1 - \cos \omega\tau_i)m_x^2 + \cos \omega\tau_i & (1 - \cos \omega\tau_i)m_x m_y - (\sin \omega\tau_i)m_z & (1 - \cos \omega\tau_i)m_z m_x + (\sin \omega\tau_i)m_y \\ (1 - \cos \omega\tau_i)m_x m_y + (\sin \omega\tau_i)m_z & (1 - \cos \omega\tau_i)m_y^2 + \cos \omega\tau_i & (1 - \cos \omega\tau_i)m_y m_z - (\sin \omega\tau_i)m_x \\ (1 - \cos \omega\tau_i)m_z m_x - (\sin \omega\tau_i)m_y & (1 - \cos \omega\tau_i)m_y m_z + (\sin \omega\tau_i)m_x & (1 - \cos \omega\tau_i)m_z^2 + \cos \omega\tau_i \end{bmatrix}, \quad (41)$$

where

$$\mathbf{m} = (m_x, m_y, m_z) \quad \text{and} \quad \|\mathbf{m}\| = 1. \quad (42)$$

6.3.2 Parameter Gradient of Objective Function

In this inter- and intra-registration, we likewise cannot update the geometric point in the same way as in the case of rigid-body transformation. If a 3D point \mathbf{x} is iteratively changed to \mathbf{x}' and \mathbf{x}'' by a rigid-body transformation (\mathbf{R}, \mathbf{t}) and (\mathbf{R}', \mathbf{t}'), the relationship between \mathbf{x} and \mathbf{x}' is:

$$\mathbf{x}' = \mathbf{R}\mathbf{x} + \mathbf{t}, \quad (43)$$

$$\mathbf{x}'' = \mathbf{R}'\mathbf{x}' + \mathbf{t}'. \quad (44)$$

However, assuming that a 3D point \mathbf{x} is iteratively changed to \mathbf{x}' and \mathbf{x}'' by this inter- and intra-registration ($\mathbf{R}, \Delta\mathbf{R}, \mathbf{v}, \mathbf{T}$) and ($\mathbf{R}', \Delta\mathbf{R}', \mathbf{v}', \mathbf{T}'$), the equation above is not valid. because \mathbf{x}' and \mathbf{x}'' expand to the following:

$$\mathbf{x}' = \mathbf{R}(\Delta\mathbf{R}\mathbf{x} + \mathbf{v}\tau) + \mathbf{T}, \quad (45)$$

$$\mathbf{x}'' = \mathbf{R}'\mathbf{R}\{\Delta\mathbf{R}'\mathbf{x} + (\mathbf{v} + \mathbf{v}')\tau\} + (\mathbf{T} + \mathbf{T}'). \quad (46)$$

In rigid-body transformation, the descent gradient is obtained by evaluating the rigid-body rotation at an identity quaternion (q_I) for all parameters, but that method cannot be used here because of the updating problem as described above. Specifically, the updated rigid-body rotation is not reflected if it is evaluated at q_I for all parameters. So we evaluate the rigid-body rotation at q_I only for rigid-body rotation parameters.

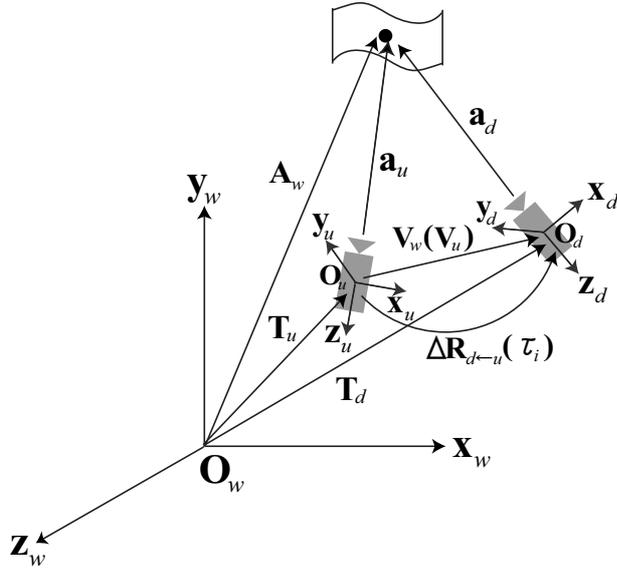


Figure 11: Positional relationships in the case of intra-scanning registration. O_w , is the origin of the world coordinate system, O_u the camera coordinate system origin when the measurement starts, and O_d is the camera coordinate origin when the measurement is finished. Capital and small letters are respectively concerned with the world coordinate and each camera coordinate shown by their subscript.

To derive the descent gradient in the non-rigid case, we replacing $\mathbf{R}_{w \leftarrow u}$, $\Delta \mathbf{R}_{d \leftarrow u}^{-1}(\tau_i)$, \mathbf{a}_d , \mathbf{v}_u , τ_i , \mathbf{T}_u , and \mathbf{b}_u with \mathbf{R} , $\Delta \mathbf{R}(\tau_i)$, \mathbf{x}_i , \mathbf{v} , τ_i , \mathbf{T} , and \mathbf{y}_{ji} in equations (39) and (40) to obtain the following:

$$z_i = \|\mathbf{R}\mathbf{g}(\mathbf{x}_i, \mathbf{p}_{intra}) + \mathbf{T} - \mathbf{y}_{ji}\|^2, \quad (47)$$

$$\mathbf{g}(\mathbf{x}_i, \mathbf{p}_{intra}) = \Delta \mathbf{R}(\mathbf{m}, \omega \tau_i) \mathbf{x}_i + \mathbf{v} \tau_i, \quad (48)$$

where

$$\mathbf{p}_{intra} = (\mathbf{v}, \mathbf{m}, \omega). \quad (49)$$

6.4 Experiment

As an experiment on an actual case, we executed our algorithm on the data of the Bayon temple. In this experiment, we aligned the corresponding data captured by our FLRS and Cyrax 2500. The latter data set was scanned from the stable ground so that there was no movement during scanning, and we assume that it is sufficiently reliable. The details of the data set and the registration process are respectively shown in Figures 12 and 13. You can see that our algorithm aligned and fitted the FLRS's data well in comparison to the Cyrax2500's data.

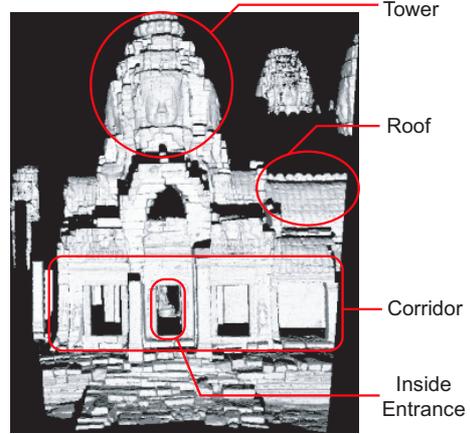


Figure 12: Detail of Bayon Data. This range data is the partial shape of the Bayon temple in Cambodia.

6.5 Evaluation

To evaluate the accuracy of the algorithm, we aligned the two data sets used in the accuracy evaluation of our rigid-body transformation parameter (See Section 3.3.3). One data set is regarded as a data set without distortion, and the other as being deformed according to equation (40). This latter data set is equivalent to the quality of a data set obtained by FLRS. We considered the position difference (inter-translation and inter-rotation: rigid-body translation and rotation) of our two data sets.

The initial intra-transformation (intra-translation and intra-rotation) of pseudo-FLRS data was manipulated in five steps: exclusive translation to ± 0.5 , ± 0.25 , and 0 [m/s] in each axis, and exclusive rotation to ± 20 , ± 10 , and 0 [deg/s] around each axis. In this scenario we assume the rotation axis is known. The number of actual intra-transformation cases are therefore 168 because the number of only intra-translation, only intra-rotation, and combined intra-transformation is respectively 12 (4 (± 0.5 and ± 0.25) $\times 3$ (each axis)), 12 (4 (± 20 and ± 10) $\times 3$ (each axis)), 144 (12×12). (The effect of a varying initial rotation axis is investigated later.)

Next, the effects of inter-transformation (inter-translation and inter-rotation) are considered. The initial inter-transformation was manipulated in three steps: translation to ± 0.1 and 0 [m], and rotation to ± 5 and 0 [deg]. In this case, the number of inter-transformation is 12 because the number of only inter-translation and only inter-rotation is 6 (2 (± 0.5) $\times 3$ (each axis)) and 6 (2 (± 5) $\times 3$ (each axis)). Thus, the total number of evaluation pattern is 2169 (144 (only intra-transformations) $+ 12$ (only inter-transformations) $+ 144 \times 12$ (combination of intra- and inter-transformation)).

As a numerical result, we show 24 evaluation results

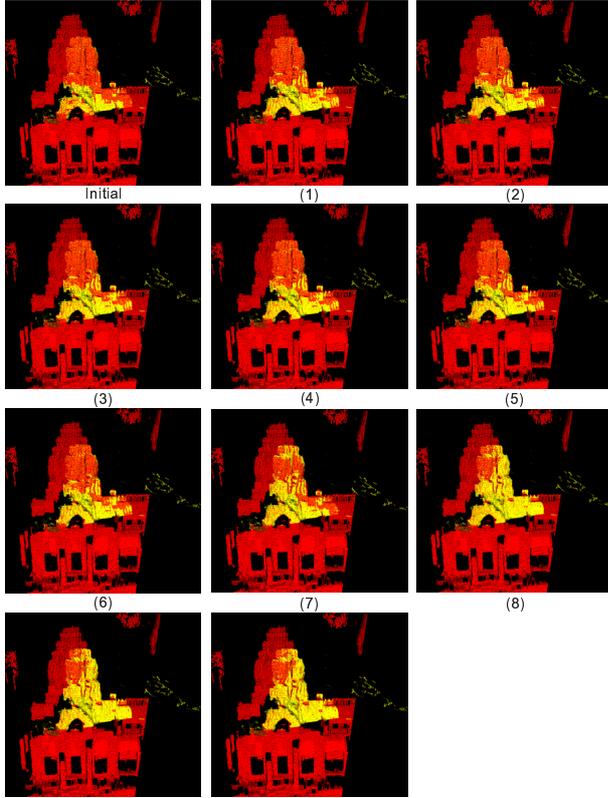


Figure 13: Range images in our registration process: A range image of FLRS (yellow) is aligned and fitted onto the corresponding range image of Cyrax 2500 (red) simultaneously.

on behalf of all the estimation results in Figures 14 and 15. Figures 14 and 15 respectively investigate the effects of only inter-transformation as well as only intra-transformation on parameter estimation. In these figures, “x-t” means translation along x axis, and “x-r” means rotation around x axis. Judging from these figures, the resulting errors of inter-translation, inter-rotation, intra-translation, and intra-rotation are respectively within $0.15 [m]$, $2.2 [deg]$, $0.11[m/s]$, and $1.9 [deg/s]$. It can be seen that inter-transformation errors tend to have more effect on parameter estimation than that of intra-transformation.

For estimation evaluation of an intra-rotation axis, a data set in which intra-translation is set to $-0.25 [m]$ along z axis ($= (0, 0, 1)$) and intra-rotation is set to $-10 [deg]$ is aligned to the data set without distortion. Then, we set the initial intra-rotation axis to $(0.00, 0.50, 0.87 (= \frac{\sqrt{3}}{2}))$. After registration, estimated axis of intra-rotation is $(0.00, 0.48, 0.88)$. Therefore, the intra-rotation axis can be reasonably estimated by our method, with its resulting appearance visibly close to the truth (Figure 16). However, the initial estimate must be somewhat close to the truth or wrong convergence may

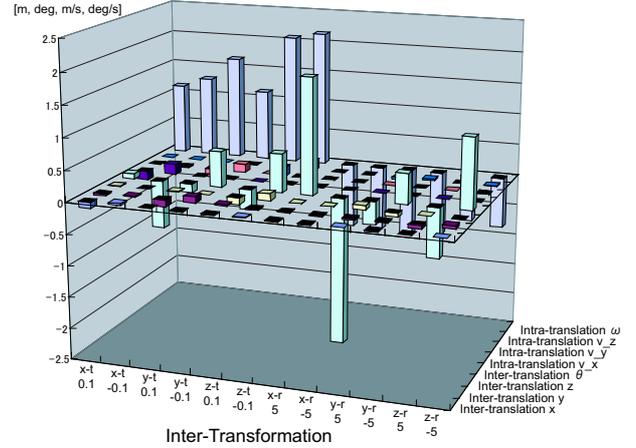


Figure 14: Estimation errors of parameters for each initial inter-transformation.

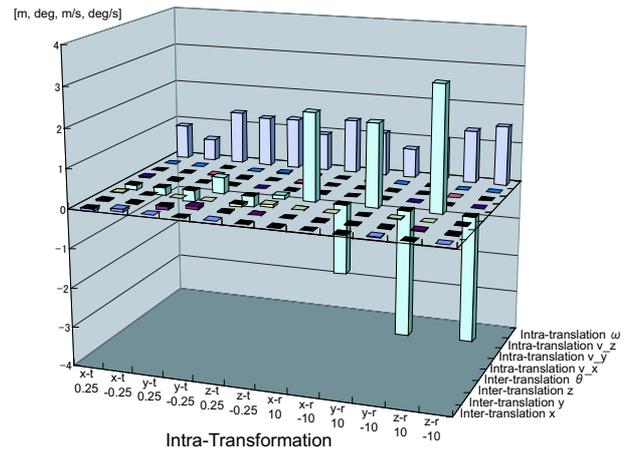


Figure 15: Estimation errors of parameters for each initial intra-transformation.

result.

Of 2196 cases, we considered 1860 cases resulted in good registration. Generally speaking, as long as the initial error of inter-translation, inter-rotation, intra-translation, and intra-rotation is respectively within $\pm 0.5 [m]$, $\pm 5 [deg]$, $\pm 0.25 [m/s]$, and $\pm 20 [deg/s]$, the registration result will be accurate, even if these effects are combined.

7 Conclusion and Future Work

In this thesis, we proposed the robust simultaneous registration of 3D shape data. This registration reduces the solution of a nonlinear equation to iteratively minimize the distance between a pair of corresponding 3D data sets. As a preparation for designing the registration algorithm, we analyzed

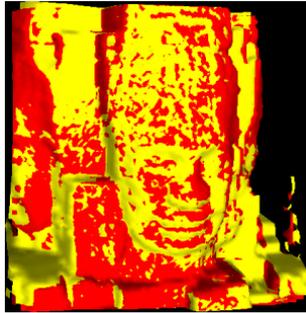


Figure 16: Registration result in the case which the different parameter of intra-rotation axis is initially set.

the merits and demerits of conventional methods, and to design the most accurate registration algorithm, we adopted the simultaneous ordering, all point matching, closest point-to-point distance, and M-estimator for outlier elimination.

To verify the robustness of our registration against the initial position and the measurement noise, we evaluated the estimation accuracy of registration parameter, comparing the registration result between our method and conventional registration. To summarize our implemented registration, it can align only the identical part of superposing 3D shape data robustly because of the use of M-estimator, and can restore even complex shape since simultaneous strategy is employed.

Moreover, we extended our registration, namely, rigid-body transformation, to enable registration among 3D data that can deform each other through some known mathematical formula. This extended method requires determining more parameters concerned with shape than just the six translation and rotation parameters. It can solve the rigid-body transformation and shape parameter in a unified manner. Here we assumed that the shape changes are strictly defined by some parameterized formulation derived from the deformation mechanisms.

We employed our extended registration to estimate the shape parameter from the shape measurement data of mathematical plaster models. We successfully estimated their parameters and reproduced a metallic replica model of the original Dini's Surface with high accuracy. We verified the estimation accuracy through a simulation-based experiment.

We also applied our extended registration to registration among range images obtained by the laser range sensor suspended beneath a balloon. In contrast with a conventional 3D sensing system, this registration needs to rectify the distortion due to the movement during measurement. We also evaluated the estimation accuracy of FLRS movement, and the applicable limitation.

In our future work following, we have some goals for improving our system. The first is to automatically de-

termine the initial pose and position among aligned data sets. This automatic determination would enable totally automatic registration among 3D data sets. The problem is how to lead the initial state for our system to work well.

The next goal is to align deformable data without rigid (undeformable) data. In this thesis, we always assumed the registration between rigid data and deformable data. This problem here may also be how to determine the initial pose, position, and shape parameter as described above.

The applications we proposed here are only a few of the possible applications, and we are trying to develop an application to generate the CAD primitives under the shape parameter estimated from the range image. This application will convert the range images into the properly approximated CAD data sets. The benefit of this application is to be able to compress the range images which usually consist of numerous 3D points and polygons. We intend to apply our framework widely to various classes of problem.

Acknowledgement

I'd like to thank Professor Toshitake Kohno (Graduate School of Mathematical Sciences, The University of Tokyo), Professor Yoshiaki Nishino, Ms. Mito Ikemizu (The University Museum, The University of Tokyo), Mr. Hiroshi Sugimoto (Artist), and Mr. Yasuhiro Yamada (Yamada Seiki Co.,Ltd.) for their cooperation in my research activities.

Reference

- [1] P.J. Besl and N.D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 2, pp. 239–256, February 1992.
- [2] G. Turk and M. Levoy. Zipped polygon meshes from range images. In *ACM SIGGRAPH Proceedings*, pp. 311–318, July 1994.
- [3] T. Masuda and N. Yokota. A robust method for registration and segmentation of multiple range images. *Computer Vision and Image Understanding*, Vol. 61, No. 3, pp. 295–307, 1995.
- [4] K. Pulli. Multiview registration for large data sets. In *Proceedings of the 2nd International Conference on 3D Digital Imaging and Modeling*, pp. 160–168, 1999 October.
- [5] G. Blais and M. Levine. Registering multiview range data to create 3d computer objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 8, pp. 820–824, 1995.

- [6] K. Higuchi, M. Herbert, and K. Ikeuchi. Building 3-d models from unregistered range images. In *Graphical Models and Image Processing*, Vol. 57, pp. 315–333, July 1995.
- [7] A.E. Johnson and M. Herbert. Surface matching for object recognition in complex 3-dimensional scenes. *Image and Vision Computing*, Vol. 16, No. 9/10, pp. 635–651, July 1998.
- [8] David Simon. *Fast and Accurate Shape-Based Registration*. PhD thesis, School of Computer Science, Carnegie Mellon University, 1996.
- [9] Y. Chen and G.G. Medioni. Object modeling by registration of multiple range images. *Image and Vision Computing*, Vol. 10(3), No. 3, pp. 145–155, 1992.
- [10] P. Neugebauer. Geometrical cloning of 3d objects via simultaneous registration of multiple range images. In *Proceedings of International Conference on Shape Modeling and Application*, pp. 130–139, March 1997.
- [11] T. Oishi, A. Nakazawa, R. Kurazume, and K. Ikeuchi. Fast simultaneous alignment of multiple range images using index images. In *Proceedings of the 5th International Conference on 3-D Digital Imaging and Modeling (3DIM 2005)*, pp. 476–483, June 2005.
- [12] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the icp algorithm. In *Proceedings of the 3rd International Conference on 3D Digital Imaging and Modeling*, pp. 145–152, May 2001.
- [13] J. Feldmar and N. Ayache. Rigid and affine registration of smooth surfaces using differential properties. In *Proceedings of Third European Conference on Computer Vision (ECCV'94)*, pp. 397–406, 1994.
- [14] K. Nishino and K. Ikeuchi. Robust simultaneous registration of multiple range images. In *Proceedings of the 5th Asian Conference on Computer Vision*, Vol. 2, pp. 455–461, January 2002.
- [15] A.E. Johnson and S. Kang. Registration and integration of textured 3-d data. In *Proceedings of International Conference on 3D Digital Imaging and Modeling*, pp. 234–241, May 1997.
- [16] Z.Y. Zhang. Iterative point matching for registration of free form curves and surfaces. *International Journal of Computer Vision*, Vol. 13, No. 2, pp. 119–152, October 1994.
- [17] Mark D. Wheeler and Katsushi Ikeuchi. Sensor modeling, probabilistic hypothesis generation, and robust localization for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 3, pp. 252–265, March 1995.
- [18] Mark D. Wheeler. *Automatic Modeling and Localization for Object Recognition*. PhD thesis, School of Computer Science, Carnegie Mellon University, 1996.
- [19] C. V. Stewart, C. L. Tsai, and A. Perera. A view-based approach to registration: Theory and application to vascular image registration. In *Proceedings of International Conference on Information Processing in Medical Imaging (IPMI)*, pp. 475–486, 2003.
- [20] A. Guéziec, X. Pennec, and N. Ayache. Medical image registration using geometric hashing. *IEEE Computational Science and Engineering, special issue on Geometric Hashing*, Vol. 4, No. 4, pp. 29–41, 1997.
- [21] Eric Bardinet, Laurent D. Cohen, and Nicholas Ayache. A parametric deformable model to fit unstructured 3d data. *Computer Vision and Image Understanding*, Vol. 71, No. 1, pp. 39–54, 1998.
- [22] P. R. Andresen and M. Nielsen. Non-rigid registration by geometry constrained diffusion. In *Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI'99)*, pp. 533–543, 1999.
- [23] Ryusuke Sagawa, Tomohito Masuda, and Katsushi Ikeuchi. Effective nearest neighbor search for aligning and merging range images. In *Proceedings of 4th International Conference on 3-D Digital Imaging and Modeling (3DIM 2003)*, October 2003.
- [24] E. Polak. *Computational Methods in Optimization*. New York: Academic Press, 1971.
- [25] David A.H. Jacobs. *The State of the Art in Numerical Analysis*. London: Academic Press, 1977.
- [26] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. New York: Springer-Verlag, 1980.
- [27] T. Masuda, Y. Hirota, K. Ikeuchi, and K. Nishino. Simultaneous determination of registration and deformation parameters among 3d range image. In *Proceedings of 5th International Conference on 3-D Digital Imaging and Modeling (3DIM 2005)*, June.
- [28] Yamada Seiki Co.,Ltd. <http://www.yamadaseiki.co.jp>.
- [29] Gallery Koyanagi. <http://www.gallerykoyanagi.com/>.
- [30] Y. Hirota, T. Masuda, R. Kurazume, K. Ogawara, K. Hasegawa, and K. Ikeuchi. Designing a laser range finder which is suspended beneath a balloon. In *Proceedings of 6th Asian Conference on Computer Vision*.