COMPUTER VISION AND IMAGE UNDERSTANDING
Vol. 61, No. 3, May, pp. 295–307, 1995

A Robust Method for Registration and Segmentation of Multiple Range Images

TAKESHI MASUDA*

Image Understanding Section, Machine Understanding Division, Electrotechnical Laboratory, 1-1-4 Umezono, Tsukuba, Ibaraki 305, Japan

AND

NAOKAZU YOKOYA†

Nara Institute of Science and Technology, 8916-5 Takayama-cho, Ikoma, Nara 630-01, Japan

Received May 15, 1994; accepted January 11, 1995

Registration and segmentation of multiple range images are important problems in range image analysis. We propose a new algorithm of range data registration and segmentation that is robust in the presence of outlying points (outliers) like noise and occlusion. The registration algorithm determines rigid motion parameters from a pair of range images. Our method is an integration of the iterative closest point (ICP) algorithm with random sampling and least median of squares (LMS or LMedS) estimator. The segmentation method classifies the input data points into four categories comprising inliers and 3 types of outliers. Finally, we integrate the inliers obtained from multiple range images to construct a data set representing an entire object. We have experimented with our method both on synthetic range images and on real range images taken by two kinds of rangefinders. The proposed method does not need preliminary processes such as smoothing or trimming of isolated points because of its robustness. It also offers the advantage of reducing the computational cost.

1. INTRODUCTION

The problem of registration and segmentation of multiple range images is one of the most important problems in range image understanding, especially in the applications of multiple-view integration, motion analysis, object tracking, automatic model generation, and model-based object recognition. This problem has been investigated by many researchers [1].

Faugeras and Hebert [2] and Horn [3] employed quaternions to estimate rigid motion parameters from paired points. They solved the minimization of a cost function for a rigid motion in a quadratic form of a unit quaternion, which is a 4D vector that determines a 3D rotation matrix. This method is theoretically clear and also practical, but the correspondence must be established before the method is applied.

Correspondence is achieved through the use of features that are invariant to congruent transformations, such as curvatures [4]. However, curvatures are computed from the first and second derivatives, and thus they are easily affected by the noise contained in the range image. Kawai *et al.* [5] obtained correspondence using several features of segmented regions. Their method is not suitable for objects consisting of curved surfaces which are difficult to segment. All the correspondence-based approaches suffer from disappearance of features producing false correspondence caused by occlusion or noise.

Horn and Harris [6] developed constraint equations for a range image sequence as an extension of the optical flow equation. Their method, which does not need to build correspondence, extracts a small rigid motion that is uniform in the image. However, it cannot treat a scene with large motion because it uses the time differential of the range image sequence. Yamamoto *et al.* [7] derived a similar method by introducing the deformable net model to treat nonrigid motion. These approaches need well-segmented range images containing a smooth surface so that no occlusion happens.

* Visiting research scientist until June 1995 at Autonomous Systems Laboratory, Institute for Information Technology, National Research Council of Canada, Montreal Road, Building M-50, Ottawa, Ontario, Canada K1A 0R6. E-mail addresses: (permanent) tmasuda@etl.go.jp; (until June 1995) tmasuda@autsrv.iitsg-nrc.ca.

† E-mail address: yokoya@is.aist-nard.ac.jp.

1077-3142/95 $6.00

Besl and McKay [8] proposed the *iterative closest point* (ICP) algorithm. Each iteration of the algorithm contains the establishment of point correspondences and the rigid motion estimation using a unit quaternion. They treated two 3D shapes: a data shape and a model shape. Each data point in the data shape corresponds to the closest model point in the model shape. Rigid motion estimation using quaternions is then applied using these correspondences, and the data shape is then transformed according to the computed motion parameters. The correspondence is updated between the transformed data shape and the model shape. These procedures are iterated until the two shapes converge. This method works well if every data point has its corresponding model point. However, the method is affected by outliers such as noise and occlusion.

The *least median of squares* (LMS or LMedS) estimator gives an estimation that minimizes the median of squared residuals, while the standard *least squares* (LS) estimator gives an estimation that minimizes the sum of squared residuals. The LMS estimator is more robust than the LS estimator in that the LMS estimator is not affected by outliers of up to 50%. The concept of the breakdown point is used to formalize the evaluation of robustness [9].

The LMS estimator is not continuous and cannot be solved in an analytic form. A stochastic technique such as *random sampling* should be integrated with it [9–11]. This is a technique that takes subsamples from the given data set at random; at least one subsample including no outliers may be selected in several trials. Motion estimation is expected to be successful if a *good* subsample is selected. The probability that one obtains a good estimate depends on both the size of subsamples and the number of trials of taking subsamples.

This paper proposes a new method of registration and segmentation of range images, which is based on techniques in the field of robust statistics in order to treat range images containing outliers such as occlusion and noise. The method enables us to estimate the motion parameter in a robust way, to extract a segment without outliers, and finally to integrate reliable data from multiple views.

Our registration method determines a rigid uniform motion between two range images in a robust way by integrating the ICP algorithm with the random sampling and the LMS estimator [12]. The method iterates the process of random sampling (Section 3.2) and estimation of motion parameters and evaluation of the estimation. The estimation is done by the ICP algorithm (Section 3.3) including motion estimation using quaternions (Section 3.4). The evaluation is based on comparison between two range images (Section 3.5). This iteration makes up the LMS estimation as a whole. We determine the motion that gives the best evaluation through the iteration.

Our segmentation method [13] uses the results of the comparison between two range images (Section 3.6). Each point in the range data is classified in one of four categories: *Occluded, Unpaired, Outlier,* and *Inlier.* A point that is hidden by the other points on the same pixel is an *Occluded* point. A point that has no paired point in another range image on the sample pixel is an *Unpaired* point. The other points are separated into *Inliers* and *Outliers* by thresholding the distance between paired points.

We apply these registration and segmentation methods to each successive pair of multiple range images of a moving object and obtain the motion parameters and inlier points. We then integrate them to make a set of data that represents the object (Section 3.7).

The main advantage of the proposed method is its robustness. This enables us to obtain reliable results even from noisy range images. We experimented on several raw range images without any preliminary processes such as smoothing or removal of isolated points. Another advantage is the reduction of the computational cost, which is derived from the random sampling. Even though out algorithm invokes several trials of the ICP algorithm, the computational cost is still reduced.

Nomenclature

$R = \{\mathbf{r}(\mathbf{u}) \mid \mathbf{u} \in U\}$	Range image whose image coordinate is $\mathbf{u}$ and whose regular domain is U
$\mathbf{r}(\mathbf{u}) = (r_1(\mathbf{u}), r_2(\mathbf{u}), r_3(\mathbf{u}))$	Point in the range image on the pixel $\mathbf{u}$
$\mathbf{u} = (u_1, u_2) \in U$	Image coordinate
$\mathbf{r}^{-1}(\mathbf{x}) = (r_1^{-1}(\mathbf{x}), r_2^{-1}(\mathbf{x}))$	Projection of the range image
$\mathbf{T} = \{\mathbf{R}, \mathbf{t}\}$	3-D motion with a 3×3 rotation matrix $\mathbf{R}$ and a 3D translation vector $\mathbf{t}$
$\mathbf{T}(\mathbf{x}) = \mathbf{R}\mathbf{x} + \mathbf{t}$	Transformation of point x
$\mathbf{T}(X) = \{\mathbf{R}\mathbf{x}_i + \mathbf{t}\}$	Transformation of point set X
$R^{\mathrm{I}} = \{\mathbf{r}^{\mathrm{I}}(\mathbf{u}) \mid \mathbf{u} \in U^{\mathrm{I}}\}$	First range image for the registration
$R^{\mathrm{II}} = \{\mathbf{r}^{\mathrm{II}}(\mathbf{u}) \mid \mathbf{u} \in U^{\mathrm{II}}\}$	Second range image for the registration
$\mathbf{T}_{\mathrm{LMS}}$	Motion estimated by LMS
$\mathbf{T}_0 = \{\mathbf{I}, \mathbf{0}\}$	Identical transformation
$\mathrm{LMS}(R^{\mathrm{I}}, \mathbf{T}, R^{\mathrm{II}})$	LMS motion estimation between R^{I} and R^{II} with the initial motion $\mathbf{T}$
$P^{\mathrm{I}}_{\mathrm{RS}} \leftarrow \mathrm{RS}(R^{\mathrm{I}}, N_S)$	Randomly sampled set of N_S points from R^{I}

$\mathbf{T}_{ICP,n}$	nth estimated motion by the ICP algorithm
$ICP(P^{I}_{RS}, \mathbf{T}, R^{II})$	Motion estimation between P^{I}_{RS} and R^{II} with the initial motion $\mathbf{T}$
$MS(R^{I}, \mathbf{T}, R^{II})$	Motion evaluation using the median of squares of residuals comparing with $\mathbf{T}(R^{I})$ and R^{II}
$p(\varepsilon, N_S, N_T)$	Probability that at least one subsample of N_S points of only inliers is drawn during N_T trials
$P^{I} = \{\mathbf{p}^{I}_i\}$	Point set used in the ICP algorithm
$P^{II} = \{\mathbf{p}^{II}_i\}$	Point set paired to P^{I} by the closest correspondence operation
$C(P^{I}, R^{II})$	Closest correspondence operation
$Q(P^{I}_{RS}, P^{II})$	Motion estimation using the quaternion representation
τ	Threshold signifying the tolerance of the convergence
σ	Scale of the input data
$\mathbf{r}^{I'}$	Point transformed from $\mathbf{r}^{I}$
$r(\mathbf{v})$	Residual on the pixel $\mathbf{v}$
$MS(R^{I}, \mathbf{T}, R^{II})$	Median of squares evaluation of $\mathbf{T}$ comparing with R^{I} and R^{II}
$\{R^{I}[0], R^{I}[1], R^{I}[2], R^{I}[3]\}$	Segmented result of R^{I}
$Seg(R^{I}; \mathbf{T}(R^{I}), R^{II})$	Segmentation of R^{I} comparing with $\mathbf{T}(R^{I})$ and R^{II}
θ	Threshold used in segmentation
$\hat{\sigma}$	Estimated standard deviation
$R^{t}(1 \le t \le N_R)$	Multiview N_R range images

2. RANGE IMAGES AND RIGID MOTION PARAMETERS

A range image is a set of 3D coordinates $R = \{\mathbf{r}(\mathbf{u}) \mid \mathbf{u} \in U\}$, where each element is given by

$$\mathbf{r}(\mathbf{u}) = (r_1(\mathbf{u}), r_2(\mathbf{u}), r_3(\mathbf{u})), \tag{1}$$

where a point in the 2D image plane is denoted as $\mathbf{u} = (u_1, u_2) \in U$. The set U denotes the regular domain that is a set of coordinates of pixels on which 3D coordinates of measured points are given. Our method uses the projection representing the integer coordinates

$$\mathbf{r}^{-1}(\mathbf{x}) = (r_1^{-1}(\mathbf{x}), r_2^{-1}(\mathbf{x})) \tag{2}$$

on the 2D image plane U. This signifies the location on which a 3D point $\mathbf{x} = (x_1, x_2, x_3)$ is projected. This specifies the geometry of a rangefinder, by which a range image is generated from a scene. The function $\mathbf{r}^{-1}$ satisfies $\mathbf{u} = \mathbf{r}^{-1}(\mathbf{r}(\mathbf{u}))$. We describe the actual projections of rangefinder systems employed in our experiments in Section 4.

We represent a set of 3D rigid motion parameters by a 3D congruent transformation that consists of a 3×3 rotation matrix $\mathbf{R}$ and a 3D translation vector $\mathbf{t}$, namely $\mathbf{T} = \{\mathbf{R}, \mathbf{t}\}$. A rigid motion of a 3D point $\mathbf{x}$ is given by a linear transformation as

$$\mathbf{T}(\mathbf{x}) = \mathbf{R}\mathbf{x} + \mathbf{t}. \tag{3}$$

The transformation of a set of points $X = \{\mathbf{x}_i\}$ is represented by $\mathbf{T}(X) = \{\mathbf{R}\mathbf{x}_i + \mathbf{t}\}$.

3. REGISTRATION OF RANGE IMAGES USING LMS ESTIMATION

3.1. *Overview of the Registration Algorithm*

Range data registration uses LMS estimation to determine the rigid motion parameters $\mathbf{T}_{LMS}$ between two range images $R^{I} = \{\mathbf{r}^{I}(\mathbf{u}) \mid \mathbf{u} \in U^{I}\}$ and $R^{II} = \{\mathbf{r}^{II}(\mathbf{u}) \mid \mathbf{u} \in U^{II}\}$, where the sets U^{I} and U^{II} denote the regular domains of both range images. The motion $\mathbf{T}_{LMS}$ is initially set to the identical transformation $\mathbf{T}_0 = \{\mathbf{I}, \mathbf{0}\}$ and updated to the motion that is evaluated "best ever" in each trial. At the end of all trials, $\mathbf{T}_{LMS}$ satisfies the LMS condition. *Our registration process* $LMS(R^{I}, \mathbf{T}_0, R^{II})$ *is carried out as follows:*

1. Initialization: $\mathbf{T}_{LMS} \leftarrow \mathbf{T}_0$.
2. We repeat Steps 3–6 taking n from 1 to N_T, where N_T signifies the number of trials.
3. A set of N_S points P^{I}_{RS} is extracted from R^{I} at random: $P^{I}_{RS} \leftarrow RS(R^{I}, N_S)$.
4. The point set P^{I}_{RS} is used by the ICP algorithm with the second image R^{II} to estimate the motion parameters: $\mathbf{T}_{ICP,n} \leftarrow ICP(P^{I}_{RS}, \mathbf{T}_{LMS}, R^{II})$.
5. The estimated motion is evaluated by $MS(R^{I}, \mathbf{T}_{ICP,n}, R^{II})$.
6. If $MS(R^{I}, \mathbf{T}_{ICP,n}, R^{II}) < MS(R^{I}, \mathbf{T}_{LMS}, R^{II})$, then $\mathbf{T}_{LMS} \leftarrow \mathbf{T}_{ICP,n}$ is updated.

The resultant motion $\mathbf{T}_{LMS}$ satisfies $MS(R^{I}, \mathbf{T}_{LMS}, R^{II}) = \min_{1 \le n \le N_T} MS(R^{I}, \mathbf{T}_{ICP,n}, R^{II})$, and this is the result of the LMS motion estimation between R^{I} and R^{II}.

3.2. *Random Sampling*

Random sampling is a technique that tries to get a good estimate by using a good subsample that contains only inliers. We apply this technique to the first range image R^{I}, and draw a set of N_S points P^{I}_{RS} from R^{I} at random.

We evaluate the probability of getting a good subsample by considering random sampling from a probabilistic viewpoint. Let ε be the rate of outlying points (outliers) such as noise and points that is occluded, disappeared or appeared. The probability that one draws an inlier in the random sampling is $1 - \varepsilon$. Further, the probability that one picks up a subsample containing N_S points that are all inliers is $(1 - \varepsilon)^{N_S}$. The probability that at least one subsample is composed of only inliers in N_T trials is evaluated as

$$p(\varepsilon, N_S, N_T) = 1 - (1 - (1 - \varepsilon)^{N_S})^{N_T}. \quad (4)$$

This is the probability that the random sampling process finds a good subsample of only inliers, and this probability obviously increases if ε decreases to zero, N_S decreases, or N_T increases.

3.3. *ICP Algorithm*

The ICP algorithm [8] estimates rigid motion parameters between paired 3-D shapes without preliminary correspondences. This algorithm is composed of two procedures: one generates temporary correspondences and the other estimates the motion using a unit quaternion from the point correspondences. These two procedures are iterated until the two given shapes are registered by the estimated transformation.

In our algorithm, the set of points $P^{\mathrm{I}}_{\mathrm{RS}}$ selected by the random sampling and the second range image R^{II} are given for the ICP algorithm. The motion estimate of the ICP algorithm is initialized by the motion $\mathbf{T}_{\mathrm{LMS}}$ in Section 3.1 that is evaluated best during the preceding trials. Each $\mathbf{p}^{\mathrm{I}}_i \in P^{\mathrm{I}}$ is temporarily paired to the closest point $\mathbf{p}^{\mathrm{II}}_i \in R^{\mathrm{II}}$, which satisfies

$$\|\mathbf{p}^{\mathrm{I}}_i - \mathbf{p}^{\mathrm{II}}_i\| = \min_{\mathbf{r}^{\mathrm{II}} \in R^{\mathrm{II}}} \|\mathbf{p}^{\mathrm{I}}_i - \mathbf{r}^{\mathrm{II}}\|, \quad (5)$$

and a set of points $P^{\mathrm{II}} = \{\mathbf{p}^{\mathrm{II}}_i\}$ $(1 \le i \le N_S)$ is determined. We refer to this operation as

$$P^{\mathrm{II}} = \mathrm{C}(P^{\mathrm{I}}, R^{\mathrm{II}}). \quad (6)$$

The point sets P^{I} and P^{II} are used to estimate rigid motion parameters $\mathbf{T}_{\mathrm{ICP}}$ using the quaternion representation. The point set P^{I} is updated to $\mathbf{T}_{\mathrm{ICP}}(P^{\mathrm{I}}_{\mathrm{RS}})$ and these procedures are iterated until P^{I} and P^{II} converge. The ICP algorithm $\mathrm{ICP}(P^{\mathrm{I}}_{\mathrm{RS}}, \mathbf{T}_{\mathrm{LMS}}, R^{\mathrm{II}})$ is summarized as follows:

1. $k \leftarrow 1$, $P^{\mathrm{I}} \leftarrow \mathbf{T}_{\mathrm{LMS}}(P^{\mathrm{I}}_{\mathrm{RS}})$, $d_0 \leftarrow \infty$.
2. The point correspondence is established: $P^{\mathrm{II}} \leftarrow \mathrm{C}(P^{\mathrm{I}}, R^{\mathrm{II}})$.
3. The motion between paired points is estimated using the quaternion representation: $\mathbf{T}_{\mathrm{ICP}} \leftarrow \mathrm{Q}(P^{\mathrm{I}}_{\mathrm{RS}}, P^{\mathrm{II}})$.
4. We apply the estimated motion to the point sets: $P^{\mathrm{I}} \leftarrow \mathbf{T}_{\mathrm{ICP}}(P^{\mathrm{I}}_{\mathrm{RS}})$.
5. $d_k \leftarrow d(P^{\mathrm{I}}, P^{\mathrm{II}})$.
6. Steps 2–5 are repeated, increasing $k \leftarrow k + 1$, until $d_{k-1} - d_k < \tau\sigma$.

The motion $\mathbf{T}_{\mathrm{ICP}}$ is the result of the ICP algorithm.

The iteration terminates if $d_{k-1} - d_k < \tau\sigma$, where $d_k = d(P^{\mathrm{I}}, P^{\mathrm{II}})$ signifies the sum of mean squared errors between $\mathbf{p}^{\mathrm{I}}_i$ and $\mathbf{p}^{\mathrm{II}}_i$. The threshold value τ represents the tolerance of the convergence. The scale factor σ makes the convergence condition dimensionless. It represents the approximate size of $P^{\mathrm{I}}_{\mathrm{RS}}$ and is defined as the square root of the trace of the covariance matrix of $P^{\mathrm{I}}_{\mathrm{RS}}$. For the ICP algorithm, it has been proven that the sequence of mean squared errors d_k decreases monotonically [8].

The procedure that is most costly in terms of computation is the establishment of point correspondence. The computational cost of the original ICP algorithm is $O(N(R^{\mathrm{I}})N(R^{\mathrm{II}}))$, where $\mathrm{N}(\cdot)$ signifies the number of points of the range data. The proposed method reduces it to $O(N_S N(R^{\mathrm{II}}))$, and it becomes $O(N_T N_S N(R^{\mathrm{II}}))$ with N_T trials. Our algorithm was able to achieve the estimate under the condition $N_T N_S < N(R^{\mathrm{I}})$ in the experiment (Section 4).

The original ICP algorithm avoids the problem of local minima by using many initial motion estimates. We have employed another approach to avoid local minima by putting the ICP algorithm into a stochastic optimization technique using random sampling and the LMS estimation. In the ICP algorithm used in our method, the estimation starts from the motion $\mathbf{T}_{\mathrm{LMS}}$ that is evaluated best ever, and if the motion $\mathbf{T}_{\mathrm{ICP}}$ resulting from the ICP algorithm is evaluated better than $\mathbf{T}_{\mathrm{LMS}}$, then $\mathbf{T}_{\mathrm{LMS}}$ is updated to $\mathbf{T}_{\mathrm{ICP}}$ (Section 3.1).

3.4. *Rigid Motion Estimation Using Quaternion Representation*

We describe the determination of the 3D rigid motion parameters $\mathbf{T}$ between paired point sets P^{I} and P^{II} that minimize the cost function

$$\frac{1}{N_S} \sum_{i=1}^{N_S} \|\mathbf{p}^{\mathrm{II}}_i - \mathbf{T}(\mathbf{p}^{\mathrm{I}}_i)\|^2. \quad (7)$$

The technique for solving this minimization problem using the quaternion representation was originally introduced by Faugeras and Hebert [2] and Horn [3]. Besl and McKay summarized it in their ICP algorithm [8]. There is also another effective technique for solving this problem, which is based on singular value decomposition [14].

A unit quaternion is a 4D vector $\mathbf{q} = (q_0, q_1, q_2, q_3)$, where $q_0 \ge 0$ and $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$. A 3×3 rotation

matrix $\mathbf{R}(\mathbf{q})$ is derived from a unit quaternion $\mathbf{q}$ as

$$R(\mathbf{q}) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 + q_2^2 - q_1^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 + q_3^2 - q_1^2 - q_2^2 \end{bmatrix}. \quad (8)$$

The quaternion representation makes the minimization of Eq. (7) equivalent to the maximization of a quadratic form of a unit quaternion. The optimal rotation matrix is given by the unit quaternion that is the unit eigenvector corresponding to the maximum eigenvalue of the 4×4 matrix whose components are generated from the cross-covariance matrix between the given pairs of point sets. The optimal translation vector is obtained by the difference between the transformed centroids of the first point set and the second one. This method uses only the centroids and the cross-covariance matrix of the given pair of point sets, and it produces a solution if more than three pairs of points are given.

3.5. *Evaluation of Estimated Motion*

We have described the motion estimation by the ICP algorithm, which uses the points selected by the random sampling. Now we discuss the method for determining the best motion parameters.

A point of the first range image $\mathbf{r}^{\mathrm{I}}(\mathbf{u})$ is transformed by the motion $\mathbf{T}$ to

$$\mathbf{r}^{\mathrm{I}'}(\mathbf{u}) = \mathbf{T}(\mathbf{r}^{\mathrm{I}}(\mathbf{u})) \quad (\mathbf{u} \in U^{\mathrm{I}}). \quad (9)$$

The projection of the transformed 3D point on the image plane is changed from $\mathbf{u}$ to $\mathbf{v}$ which is given by

$$\mathbf{v} = \mathbf{r}^{-1}(\mathbf{r}^{\mathrm{I}'}(\mathbf{u})). \quad (10)$$

If more than one point is projected on the same pixel, the point closest to the viewpoint is selected. This simulates how rangefinder systems measure objects. This reprojected data set is a transformed range image $\mathbf{T}(R^{\mathrm{I}})$, whose regular domain is $U^{\mathrm{I}'}$. The squared residual at a pixel $\mathbf{v}$ is given by

$$r(\mathbf{v})^2 = \|\mathbf{r}^{\mathrm{I}'}(\mathbf{v}) - \mathbf{r}^{\mathrm{II}}(\mathbf{v})\|^2. \quad (11)$$

This residual is calculated for every pixel in U^{II} (Fig. 1). We define the residual to be infinitely large for the pixels in $C(U^{\mathrm{I}'}) \cap U^{\mathrm{II}}$, where C signifies the complement on which the residual cannot be determined. We use the median of these residuals to evaluate the motion $\mathbf{T}$, which

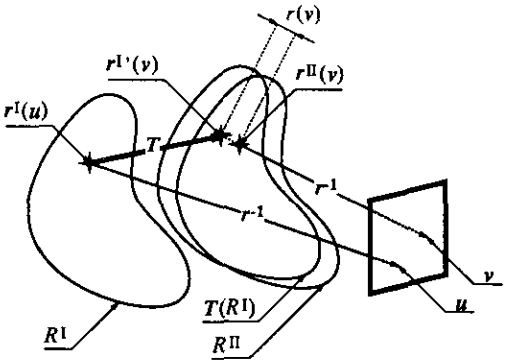


FIG. 1. Evaluation of the estimated motion parameters. A point $\mathbf{r}^{\mathrm{I}}(\mathbf{u})$ in the image R^{I} on the pixel $\mathbf{u}$ is transformed by the estimated motion $\mathbf{T}$ to the point $\mathbf{r}^{\mathrm{I}'}(\mathbf{v})$ of $\mathbf{T}(R^{\mathrm{I}})$ on $\mathbf{v}$. The residual $r(\mathbf{v})$ is determined as the distance between $\mathbf{r}^{\mathrm{I}'}(\mathbf{v})$ and $\mathbf{r}^{\mathrm{II}}(\mathbf{v})$, both of which are on the same pixel $\mathbf{v}$. The median of the residuals is used as the evaluation of the motion.

is represented as

$$\mathrm{MS}(R^{\mathrm{I}}, \mathbf{T}, R^{\mathrm{II}}) = \sqrt{\underset{\mathbf{v} \in U^{\mathrm{II}}}{\mathrm{med}}\, r(\mathbf{v})^2}. \quad (12)$$

The ICP algorithm estimates the motion $\mathbf{T}_{\mathrm{ICP},n}$ ($1 \le n \le N_{\mathrm{T}}$) using the points selected by the random sampling in each iteration. From the LMS estimation result, we find the motion $\mathbf{T}_{\mathrm{LMS}}$ corresponding to the least MS evaluation among $\mathbf{T}_{\mathrm{ICP},n}$.

In this evaluation, the transformation of the first range image requires computational cost of $O(N(R^I))$. Then we need to compute distances between two 3D points within the region $U^{\mathrm{I}'} \cap U^{\mathrm{II}}$, on which we can determine the residuals. This computational cost is given by $O(N(U^{\mathrm{I}'} \cap U^{\mathrm{II}}))$. A sorting process of these residuals is also required, whose computational complexity is discussed in [15].

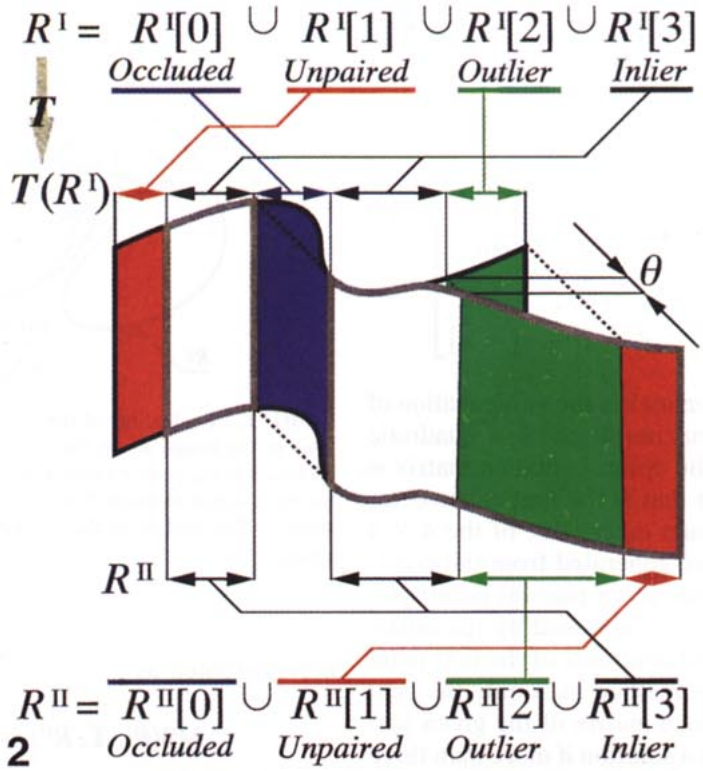
3.6. *Segmentation*

We segment the first range image R^{I} into 4 categories by comparing the transformed range image $\mathbf{T}(R^{\mathrm{I}})$ with the second range image R^{II}. We represent this segmentation of R^{I} as

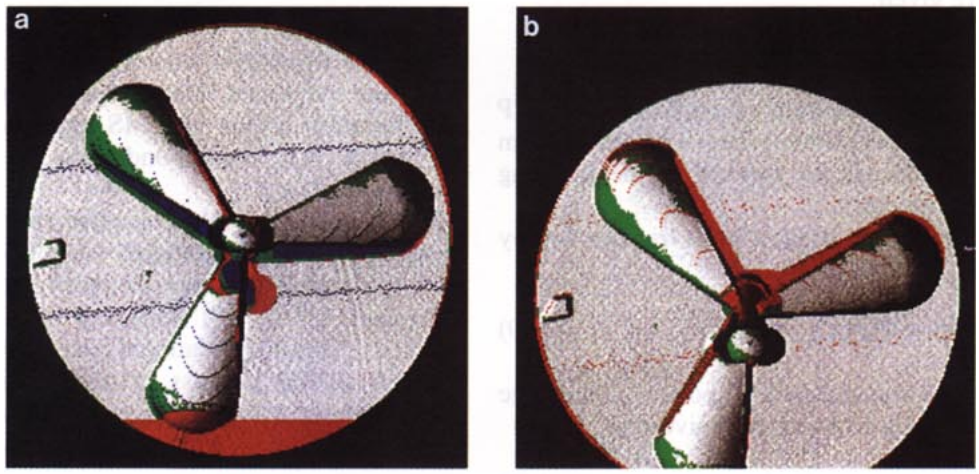
$$\{R^{\mathrm{I}}[0], R^{\mathrm{I}}[1], R^{\mathrm{I}}[2], R^{\mathrm{I}}[3]\} \leftarrow \mathrm{Seg}(R^{\mathrm{I}}; \mathbf{T}(R^{\mathrm{I}}), R^{\mathrm{II}}). \quad (13)$$

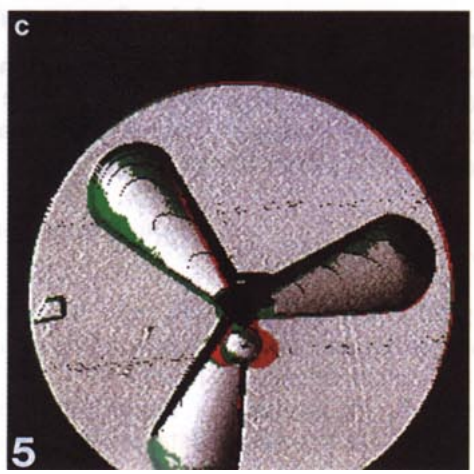
We refer to points classified in these categories as *Occluded*, *Unpaired*, *Outlier*, and *Inlier* points, respectively.

Points hidden behind the other points in the same range data are classified in $R^{\mathrm{I}}[0]$ (*Occluded* points). Other


$R^{\mathrm{I}} = R^{\mathrm{I}}[0] \cup R^{\mathrm{I}}[1] \cup R^{\mathrm{I}}[2] \cup R^{\mathrm{I}}[3]$
Occluded
Unpaired
Outlier
Inlier
T
$T(R^{\mathrm{I}})$
θ
R^{II}
$R^{\mathrm{II}} = R^{\mathrm{II}}[0] \cup R^{\mathrm{II}}[1] \cup R^{\mathrm{II}}[2] \cup R^{\mathrm{II}}[3]$
Occluded
Unpaired
Outlier
Inlier

2


a
b


c

5

points are in front and they are mapped onto the image plane. A point is classified as being in $R^{\mathrm{I}}[1]$ (*Unpaired* point) if it disappears. In this case, the transformed point is mapped onto the pixel on which no point of R^{II} is mapped. The points that are neither *Occluded* nor *Unpaired* are split by a threshold θ. A point is classified as being in $R^{\mathrm{I}}[2]$ (*Outlier*) if the residual $r(\mathbf{v})$ is larger than θ. Points in $R^{\mathrm{I}}[0]$, $R^{\mathrm{I}}[1]$, and $R^{\mathrm{I}}[2]$ may move differently or be unreliable; the remainder are classified as belonging to $R^{\mathrm{I}}[3]$ (*Inliers*). Thus each point in the first range image is classified into one of these four categories (Fig. 2).

We also apply this segmentation to the second range image, R^{II}. This is represented as

$$\{R^{\mathrm{II}}[0], R^{\mathrm{II}}[1], R^{\mathrm{II}}[2], R^{\mathrm{II}}[3]\} \leftarrow \mathrm{Seg}(R^{\mathrm{II}}; \mathbf{T}(R^{\mathrm{I}}), R^{\mathrm{II}}). \quad (14)$$

In this case, $R^{\mathrm{II}}[0]$ contains originally occluded points that may occur in some kind of range image. Unpaired points in $R^{\mathrm{II}}[1]$ have appeared by the estimated motion. Outliers $R^{\mathrm{II}}[2]$ and inliers $R^{\mathrm{II}}[3]$ are paired to those of the first image, $R^{\mathrm{I}}[2]$ and $R^{\mathrm{I}}[3]$ respectively; thus $N(R^{\mathrm{I}}[2]) = N(R^{\mathrm{II}}[2])$ and $N(R^{\mathrm{I}}[3]) = N(R^{\mathrm{II}}[3])$.

The threshold value θ is determined as $2.5\hat{\sigma}$, where $\hat{\sigma}$ signifies the standard deviation of residuals, estimated in a robust way as

$$\hat{\sigma} = 1.4826\mathrm{MS}(R^{\mathrm{I}}, \mathbf{T}_{\mathrm{LMS}}, R^{\mathrm{II}}). \quad (15)$$

The factor 1.4826 is derived from the assumption that the distribution of the residuals is Gaussian [9, 11].

3.7. *Integration of Multiple-View Range Images*

We apply the registration and segmentation algorithm to multiple-view range images $R^t (1 \le t \le N_{\mathrm{R}})$, and we construct an integrated range image using the reliable points extracted from these range images. We obtain a set of motion parameters and inliers by the registration and segmentation of each pair of successive two range images. LMS estimation is applied for R^t and R^{t+1} instead of R^{I} and R^{II}, respectively, and the motion $\mathbf{T}_{\mathrm{LMS},t}$ is determined and inliers $R^t[3]$ and $R^{t+1}[3]$ are extracted. The inlier $R^t[3]$ is stored to be integrated, and in the next registration, the random sampling is applied to only $R^{t+1}[3]$ instead of the whole image, and the motion $\mathbf{T}_{\mathrm{LMS},t}$ is used as the initial motion in estimating $\mathbf{T}_{\mathrm{LMS},t+1}$ instead of the identical transformation (Section 3.1):

1. Initialization: $\mathbf{T}_{\mathrm{LMS},0} \leftarrow \mathbf{T}_0$.
2. Steps 3–5 are repeated, taking t from 1 to $N_{\mathrm{R}} - 1$.
3. Registration: $\mathbf{T}_{\mathrm{LMS},t} \leftarrow \mathrm{LMS}(R^t, \mathbf{T}_{\mathrm{LMS},t-1}, R^{t+1})$.
4. Segmentation of R^t: $\{R^t[0], R^t[1], R^t[2], R^t[3]\} \leftarrow \mathrm{Seg}(R^t; \mathbf{T}_{\mathrm{LMS},t}(R^t), R^{t+1})$.
5. Segmentation of R^{t+1}: $\{R^{t+1}[0], R^{t+1}[1], R^{t+1}[2], R^{t+1}[3]\} \leftarrow \mathrm{Seg}(R^{t+1}; \mathbf{T}_{\mathrm{LMS},t}(R^t), R^{t+1})$.

Finally an integrated data set is obtained as a collection of inliers with the motion given by

$$R^{N_{\mathrm{R}}}[3] \cup \mathbf{T}_{\mathrm{LMS},N_{\mathrm{R}}-1}(R^{N_{\mathrm{R}}1}[3] \cup \cdots \mathbf{T}_{\mathrm{LMS},2}(R^2[3] \cup \mathbf{T}_{\mathrm{LMS},1}(R^1[3])) \cdots).$$

4. EXPERIMENTAL RESULTS

We tried the proposed method on synthetic range images and two different kinds of real range data. One of the real range data sets was obtained by the NRCC (National Research Council of Canada) rangefinder and the other by that at the ETL (Electrotechnical Laboratory). All the programs used in this experiment were written in C++ language and compiled with the g++ compiler. We use the programs shown in [15] to compute eigenvalues and eigenvectors of symmetric matrices in motion estimation using a quaternion. Computational time is measured on a single-processor workstation (HP Apollo 9000 Model 735) whose performance is 40 MFLOPS on the double-precision Linpack benchmark. In the experiment, we use $\tau = 10^{-8}$ for the termination of iteration in the ICP algorithm.

4.1. *Determination of the Number of Samples*

The number of samples N_{S} is a parameter that we can choose in this algorithm. We have shown the probability $p(\varepsilon, N_{\mathrm{S}}, N_{\mathrm{T}})$ that the random sampling selects at least one point set of only inliers in Section 3.2. This probability becomes large if N_{S} is small. However, a small N_{S} makes the ICP estimation less reliable because a sample of few points can easily fit into any part of the shape. This implies that a small N_{S} value usually requires a larger N_{T} value for obtaining good estimates. The probability that ICP estimation can produce the correct estimate using a set of N_{S} inliers cannot be predicted because it depends

FIG. 2. Segmentation of range images. The first image R^{I} is transferred by the estimated motion $\mathbf{T}$ and compared with the second image R^{II}. Each point of both images is segmented into one of four categories: *Occluded* point if it is occluded by the other point in the same image, *Unpaired* point if there is no point on the same pixel on the other image, *Inlier* point if the residual is smaller than the threshold θ, and *Outlier* point in other cases.

FIG. 5. Registration and segmentation results for (a) R^{I}, (b) R^{II}, and (c) R^{I} applied with the determined motion $\mathbf{T}_{\mathrm{LMS}}$, which is considered to be similar to R^{II}. Each pixel except *Inlier* is colored according to the category in which the point on the pixel is classified. Blue is for *Occluded* points, red is for *Unpaired* points, and green is for *Outliers*.

on both the object shape and the property of noise in the input range image. On the contrary, if N_S is large, N_T should also be large because it becomes very hard to draw a sample of only inliers. The estimated result with large N_S cannot be much better than that of the original ICP algorithm that uses all the points.

The minimum possible N_S for which the ICP algorithm can produce a solution is 3. If the estimate is unreliable then a large N_T is necessary. On the contrary, if we consider the worst case for the LMS estimation $\varepsilon = 0.5$, we need $N_T = 382$ trials for $N_S = 7$ to make the probability $p(\varepsilon, N_S, N_T)$ 95% or more. A good value of N_S is between 3 and 7, and we have chosen $N_S = 5$ for this experiment.

We limit the number of trials by $N_T = 200$, and $p(\varepsilon = 0.5, N_S = 5, N_T = 200) = 99.8\%$. The ICP algorithm does not always produce a correct solution even if all the points in the sample are inliers. The probability that we obtain a correct estimate must be lower. Larger N_T makes the estimation residual smaller but requires longer computational time. If we do not need an accurate estimate, we can use small N_T. We can also specify the acceptable residual value or the computational time to terminate the iteration of trials.

4.2. *Synthetic Range Images*

We first show experimental results on synthetic range images to demonstrate the effectiveness and robustness of our algorithm. The synthesized range image is composed of an ellipsoid whose shape is given by the equation $\frac{4}{9}x^2 + y^2 + \frac{1}{4}z^2 = 50^2$ and a hyperboloid of one sheet whose shape is given by the equation $-x^2 + y^2 + 4z^2 = 50^2$, both of which have their centers at (128, 128, 1000). The object is orthogonally projected on the image plane whose size is 256 × 256. Figure 3a illustrates this synthesized range image by coding the height value as the gray level at each pixel. This image was used for the first input image for the registration, and we generated other range images of the same object on which we applied rotations of 15° (Fig. 3b), 30°(Fig. 3c), and 45°(Fig. 3d), which are used as the second input image for the registration.

For each motion, we also did experiments on the distorted images which are generated by adding random values distributed uniformly from −128 to 128 to depth values on various amounts of pixels in both of the input images. For example, Fig. 3e and 3f are a pair of range images with 15° rotation and 20% synthetic noise. The registration results for various motions and various amounts of noise are shown in Table 1.

One can observe that estimation without noise is successful, and estimate with noise of 30% for 15° rotation, 20% for 30° rotation, and 10% for 45° fail. According to the segmentation results with application of the actual motion, the rate of outliers ε of the clear data was 23% in 15° rotation, 36% in 30° rotation, and 48% in 45° rotation; these are mainly classified as occluded points. It is reasonable that the registration process failed under the condition that the sum of all kinds of outliers including occlusion and noise exceeded 50% the pixels in the entire image.

It can be shown that our algorithm works correctly if the number of outlying points such as occlusion and noise does not exceed half of the input image. The CPU time required for registration was from 5 to 8 min in the above experiments.

4.3. *NRCC Range Images*

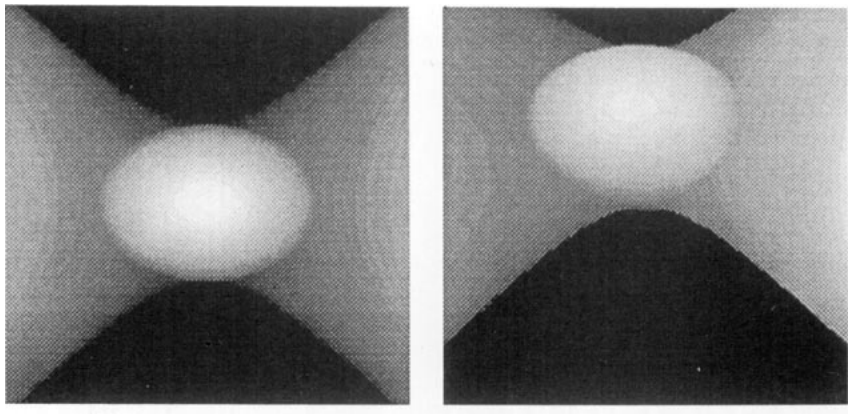
The NRCC range image is a scalar-valued 256 × 256 array containing vertical height values. The stored value at $\mathbf{u}$ is used for $r_3(\mathbf{u})$ in Equation (1) in millimeters. The others are given by $r_1(\mathbf{u}) = u_1 \times 0.5$ mm and $r_2(\mathbf{u}) = u_2 \times 0.5$ mm. The rangefinder projection $\mathbf{r}^{-1}$ in Eq. (2) is given by the orthographic projection. The height resolution is reported as 50 μm [16].

Figure 4 illustrates a pair of range images, R^{I} and R^{II}, with shading. The object is a model of a gripper for robots that is a construct of a plane, three supports, and a rod with a spherical head. Occlusion is caused by the projections, such as the rod and the supports, and truncation happens at the upper limit of the measurable domain. The coordinate system on the image plane is taken as follows: the bottom left corner is the origin (0, 0), the u_1 axis is taken from the left side to the right side, and the u_2 axis is from the bottom to the top.

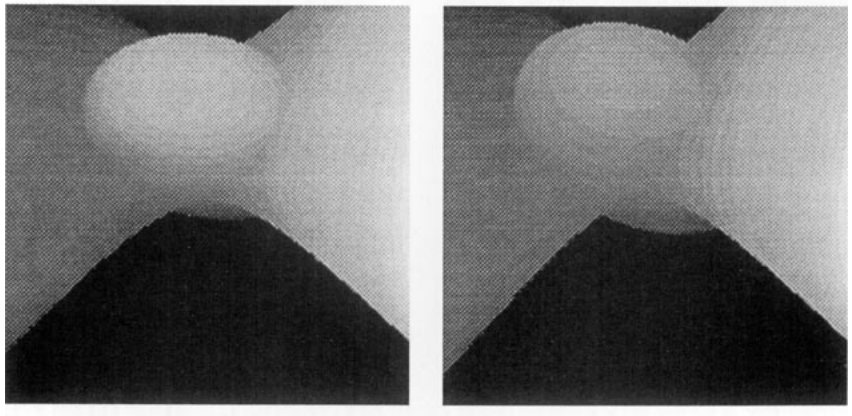
We show the registration and segmentation result for $N_S = 5$ and $N_T = 200$ that minimized the median of residuals to 191 μm. This median residual was within the possible error range of the NRCC rangefinder (horizontal ± 250 μm and vertical ±50 μm). The extracted motion parameters were

$$\left\{\begin{bmatrix} 1.000 & 0.000 & 0.001 \\ -0.000 & 0.986 & -0.170 \\ -0.001 & 0.170 & 0.986 \end{bmatrix}, \begin{bmatrix} 0.32\text{mm} \\ 5.83\text{mm} \\ -11.0\text{mm} \end{bmatrix}\right\}. \quad (16)$$

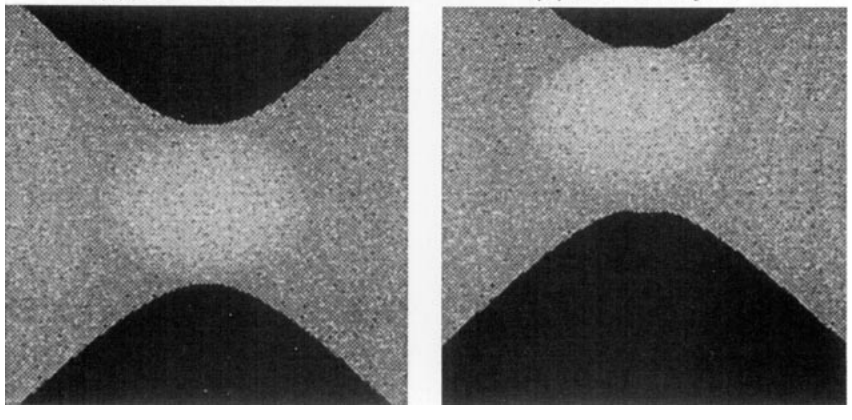
The angle of rotation was 9.74° and the axis was (1.000, 0.008, −0.002). The range image generated from the first image by this motion parameters is illustrated with shading in Fig. 5, where the segmentation result is overlaid. The threshold θ was set to 0.708 mm, and the first range image, containing 42,791 points, was segmented into *Occluded*, *Unpaired*, *Outliers*, and *Inliers* classes of 1917, 2995, 3213, and 34,666 points respectively. The second range image, containing 40,472 points, was segmented into 0, 2593, 3213, and 34,666 points, respectively. It can be seen that *Outliers* exist around the rod head and supports that stick out. The CPU time for registration and segmentation was about 6 min.



(a) Original position (b) Rotated by 15°



(c) Rotated by 30° (d) Rotated by 45°



(e) With 20% noise (f) Rotated by 15° with 20% noise

FIG. 3. Synthetic range images.

4.4. *ETL Range Images*

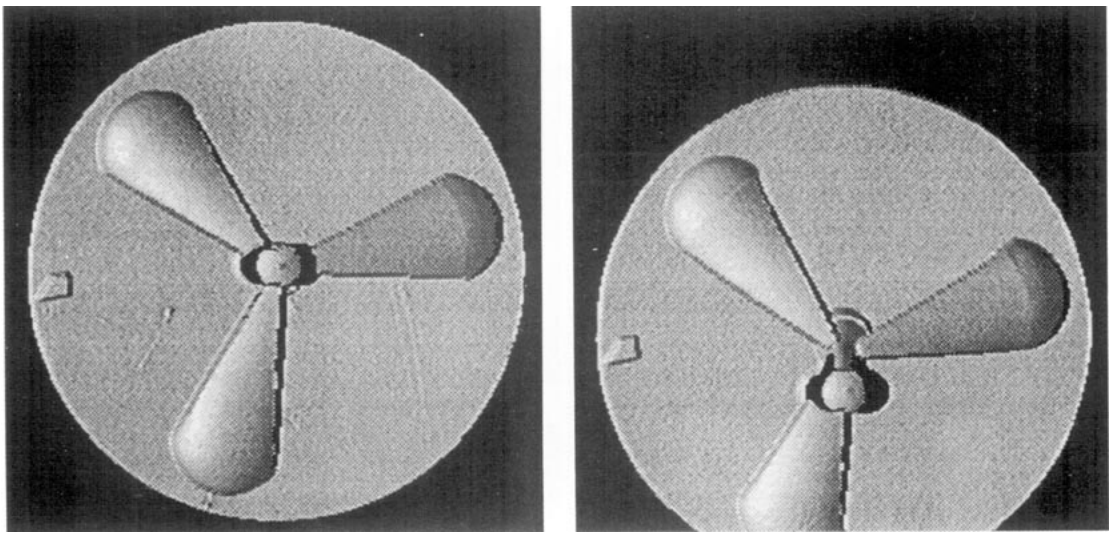
The ETL range image is a vector-valued 512 × 480 array containing 3D coordinates in meters at each pixel that directly gives $\mathbf{r}(\mathbf{u})$. The 3D coordinates are taken approximately as the first coordinate r_1 being from the left to the right, r_2 being depth, and r_3 being from the bottom to the top. The projection on the image plane $\mathbf{r}^{-1}$ is given by a composition of a 3D congruent transformation and a projective transformation. The precision of the measurement of the ETL rangefinder system is about 0.8 mm [17].

Figure 6 shows six sequential input range images with shading. The object is a plaster statue and each frame is measured after the object is rotated horizontally by approximately 10°. No preliminary procedures of trimming noise or segmentation are applied to these original images. One can observe that something stable during the sequence is also measured on the right side of the object.

TABLE 1
Registration Results for Synthetic Images

Motion	Evaluation MS	Rotation Angle	Rotation Axis	Translation	Figures
Actual	0.464	15.00°	$\begin{bmatrix} -0.707 \\ -0.707 \\ 0.000 \end{bmatrix}$	$\begin{bmatrix} 183.0 \\ -135.0 \\ 34.1 \end{bmatrix}$	
Estimated without noise	0.464	15.00°	$\begin{bmatrix} -0.706 \\ -0.708 \\ 0.001 \end{bmatrix}$	$\begin{bmatrix} 183.3 \\ -134.8 \\ 34.0 \end{bmatrix}$	(a) (b)
Estimated with 10% noise	0.597	15.16°	$\begin{bmatrix} -0.704 \\ -0.710 \\ 0.001 \end{bmatrix}$	$\begin{bmatrix} 186.2 \\ -136.2 \\ 34.6 \end{bmatrix}$	
Estimated with 20% noise	6.020	14.55°	$\begin{bmatrix} -0.650 \\ -0.760 \\ 0.016 \end{bmatrix}$	$\begin{bmatrix} 193.9 \\ -115.8 \\ 29.6 \end{bmatrix}$	(e) (f)
Estimated with 30% noise	32.546	3.54°	$\begin{bmatrix} 0.027 \\ 0.062 \\ 0.998 \end{bmatrix}$	$\begin{bmatrix} 0.5 \\ 35.7 \\ 7.2 \end{bmatrix}$	
Actual	0.570	30.00°	$\begin{bmatrix} -0.707 \\ -0.707 \\ 0.000 \end{bmatrix}$	$\begin{bmatrix} 353.6 \\ -305.6 \\ 134.0 \end{bmatrix}$	
Estimated without noise	0.565	29.89°	$\begin{bmatrix} -0.707 \\ -0.707 \\ 0.004 \end{bmatrix}$	$\begin{bmatrix} 352.6 \\ -304.5 \\ 133.0 \end{bmatrix}$	(a) (c)
Estimated with 10% noise	0.997	29.90°	$\begin{bmatrix} -0.706 \\ -0.708 \\ 0.000 \end{bmatrix}$	$\begin{bmatrix} 353.0 \\ -304.1 \\ 133.2 \end{bmatrix}$	
Estimated with 20% noise	28.292	20.55°	$\begin{bmatrix} -0.350 \\ -0.937 \\ 0.122 \end{bmatrix}$	$\begin{bmatrix} 340.7 \\ -79.2 \\ 42.6 \end{bmatrix}$	
Actual	1.002	45.00°	$\begin{bmatrix} -0.707 \\ -0.707 \\ 0.000 \end{bmatrix}$	$\begin{bmatrix} 500.0 \\ -452.3 \\ 292.9 \end{bmatrix}$	
Estimated without noise	0.806	45.00°	$\begin{bmatrix} -0.708 \\ -0.706 \\ 0.000 \end{bmatrix}$	$\begin{bmatrix} 499.8 \\ -453.1 \\ 293.4 \end{bmatrix}$	(a) (d)
Estimated with 10% noise	42.493	30.54°	$\begin{bmatrix} -0.474 \\ -0.880 \\ -0.051 \end{bmatrix}$	$\begin{bmatrix} 457.2 \\ -201.6 \\ 133.0 \end{bmatrix}$	

Note. The lines designated by "Actual" show the actual motions applied to the object and their evaluation, which are supposed to be estimated. The following lines are the estimated results with or without added noise. The column of "Figures" refers to the figure numbers in Fig. 3 that were used as the inputs for the estimate.



(a) First range image (b) Second range image

FIG. 4. NRCC range images for the input.

These images contain impulse noise points caused by the rangefinder system of only about 0.5% of the data, but these noises distribute in a region 30 times larger than the object size and severely distort the object shape.

The registration and segmentation are done with N_S = 5 and N_T = 200, and the result is shown in Table 2. All the medians of residuals MS lie within the reported accuracy of the rangefinder, and these estimates agree with the conditions under which we obtained the input range images. The segmentation process extracts 293,791 inliers,

TABLE 2
Extracted Motion Parameters as the Result of Registration and the Number of Points as the Result of Segmentation

	Registration result				Segmentation result				
t	MS (mm)	Angle	Axis	Translation (mm)	$N(R^t)$	$N(R^t[0])$	$N(R^t[1])$	$N(R^t[2])$	$N(R^t[3])$
1	0.535	9.96°	$\begin{bmatrix} -0.014 \\ 0.005 \\ -1.000 \end{bmatrix}$	$\begin{bmatrix} -47.0 \\ 12.1 \\ 0.510 \end{bmatrix}$	93094	22407	19705	465	50517
2	0.590	9.86°	$\begin{bmatrix} -0.016 \\ 0.005 \\ -1.000 \end{bmatrix}$	$\begin{bmatrix} -46.4 \\ 12.5 \\ 0.860 \end{bmatrix}$	86569	20595	19277	641	46056
3	0.612	10.00°	$\begin{bmatrix} -0.009 \\ 0.008 \\ -1.000 \end{bmatrix}$	$\begin{bmatrix} -47.0 \\ 12.4 \\ 0.587 \end{bmatrix}$	84705	18804	19825	531	45545
4	0.622	10.08°	$\begin{bmatrix} -0.008 \\ 0.012 \\ 1.000 \end{bmatrix}$	$\begin{bmatrix} -47.3 \\ 12.5 \\ 0.490 \end{bmatrix}$	84859	14862	21967	627	47403
5	0.594	9.81°	$\begin{bmatrix} -0.012 \\ 0.005 \\ -1.000 \end{bmatrix}$	$\begin{bmatrix} -46.1 \\ 12.2 \\ 0.461 \end{bmatrix}$	88456	13581	21981	759	52135
6	—	—	—	—	93969	7587	33488	759	52135

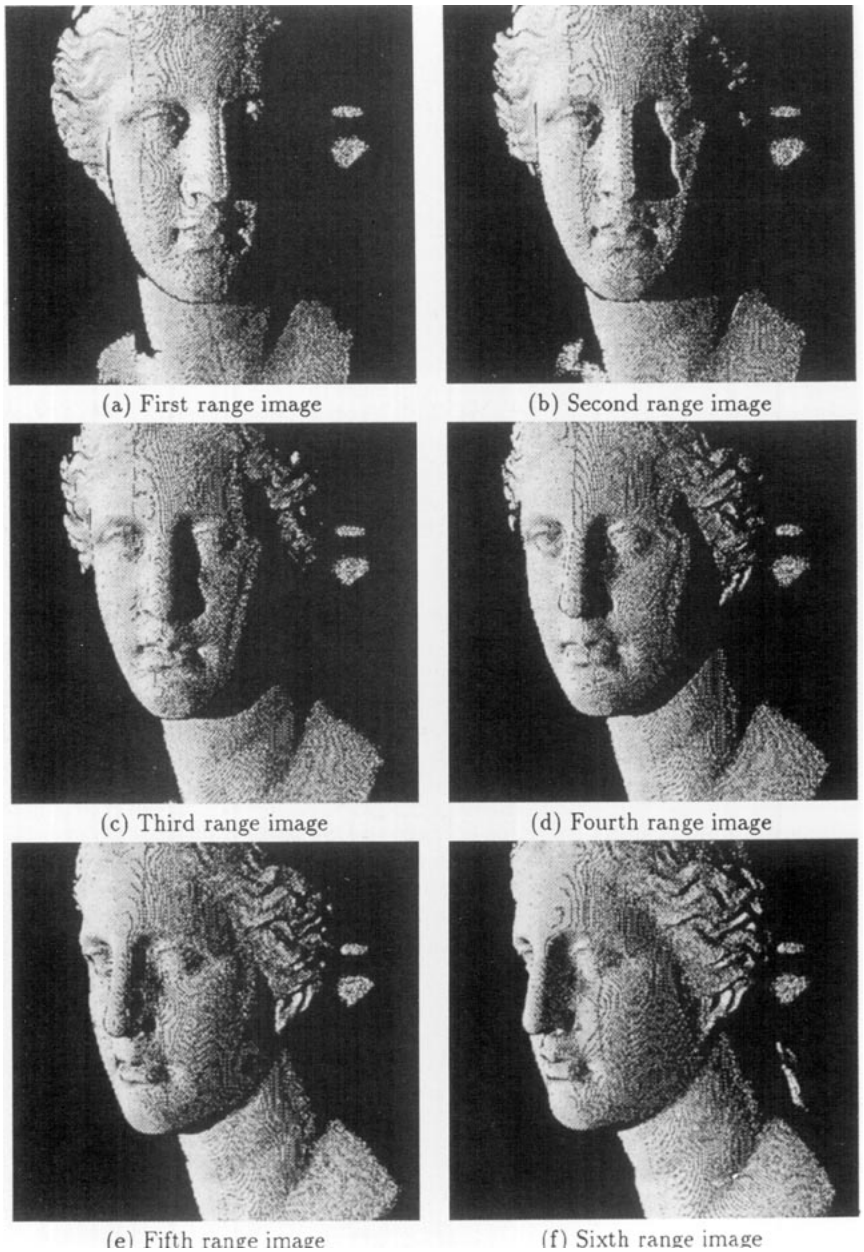


FIG. 6. A set of input ETL range images.

and the range image generated from an integrated data set of these inliers is shown in Fig. 7. The object is viewed from the same position as the third input image (Fig. 6c). One can observe that something measured on the right side was removed. Lack of data between the scanning lines, the shaded side of the face, and the region occluded by nose and chin were interpolated by the integration. The CPU time for each registration and segmentation was about 12 min. The calculation of the projection $\mathbf{r}^{-1}$ of the ETL range image requires matrix operations, whose CPU time is not negligible.

5. CONCLUSION

We have proposed a robust method for registration and segmentation of multiple range images. The registration method is an integration of the ICP algorithm with random sampling and the LMS estimator. The segmentation classifies each point in an input range image into one of four categories comprising inliers and three types of outliers. Finally, we construct a data set for the object using the inliers from multiple views.

The present method was robust even if the input data

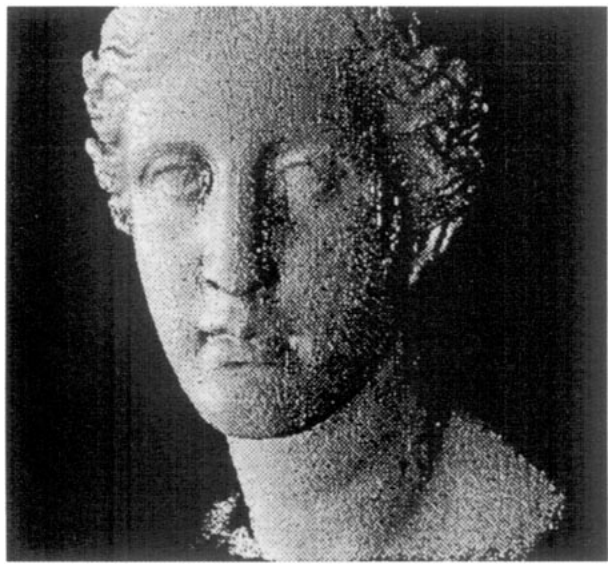


FIG. 7. Range image generated from six-view range data in Fig. 6.

contained up to 50% outliers. It also reduced the computational cost of the ICP algorithm.

We did experiments with the proposed method on a synthetic data set and raw range images measured by two different kinds of rangefinders. We chose a number of samples N_S, balancing between random sampling and the ICP algorithm. The experiments showed that the method could determine motion parameters to high enough accuracy for integration of multiple-view range data.

This method can be applied to problems such as motion-based segmentation, multiple-view integration, automatic model acquisition, motion analysis, object tracking, and model-based object recognition.

ACKNOWLEDGMENTS

This work was partially carried out under the collaboration program "Image Understanding Systems" between ETL and NRCC. The authors thank Nobuyuki Otsu, Kazuhiko Yamamoto, Jacques Domey, and Marc Rioux for making an effort to this project possible. Special thanks are due to Pierre Boulanger, Guy Godin, and Gerhard Roth at NRCC for providing us with their range images and many beneficial comments. Yoshihiro Kawai and other staff of the Computer Vision Section of ETL also offered us their range images and helped us to use the ETL rangefinder. The authors thank the staff of the Image Understanding Section of ETL and the reviewers for their comments.

REFERENCES

1. B. Sabata and J. K. Aggarwal, Estimation of motion from a pair of range images: A review, *CVGIP: Image Understanding*, **54**, 1991, 309–324.
2. O. D. Faugeras and M. Hebert, The representation, recognition, and locating of 3-D objects, *Int. J. Robotics Res.* **5**, 1986, 27–52.
3. B. K. P. Horn, Closed-form solution of absolute orientation using quite quaternions, *J. Opt. Soc. Am. A* **4**, 1987, 629–642.
4. N. Kehtarnavaz and S. Mohan, A framework for estimation of motion parameters from range images, *Comput. Vision Graphics Image Process.* **45**, 1989, 88–105.
5. Y. Kawai, T. Ueshiba, T. Yoshimi, and M. Oshima, Reconstruction of 3D objects by integration of multiple range data, *in* Proceedings 11th ICPR, *the Hague, August 1992*, Vol. I, pp. 154–157.
6. B. K. P. Horn and J. G. Harris, Rigid body motion from range image sequences, *CVGIP: Image Understanding* **53**, 1991, 1–13.
7. M. Yamamoto, P. Boulanger, J.-A. Beraldin, M. Rioux and J. Domey, Direct estimation of deformable motion parameters from range image sequence, *in Proceedings, 3rd ICCV, Osaka, Dec. 1990*, pp. 460–464.
8. P. J. Besl and N. D. McKay, A method for registration of 3-D shapes, *IEEE Trans. Pattern Anal. Mach. Intelligence* **14**, 1992, 239–256.
9. P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*, Wiley, New York, 1987.
10. G. Roth and M. D. Levine, Minimal subset random sampling for pose determination and refinement, in *Advances in Machine Vision: Strategies and Applications* (C. Archibald and E. Petriu, Eds.), pp. 1–21, World Scientific, Singapore, 1992.
11. P. Meer, D. Mintz, A. Rosenfeld, and D. Y. Kim, Robust regression methods for computer vision: A review, *Int. J. Comput. Vision* **6**, 1991, 59–70.
12. T. Masuda and N. Yokoya, Robust estimation of rigid motion parameters between a pair of range images, in *Proceedings, 8th Scandinavian Conference on Image Analysis, Tromsø, Norway, May* 1993, Vol. I, pp. 499–506.
13. T. Masuda and N. Yokoya, A robust method for registration and segmentation of multiple range images, in *Proceedings, 2nd CAD-Based Vision Workshop, Champion, PA, Feb.* 1994, pp. 106–113.
14. K. Kanatani, *Geometric Computation for Machine Vision*, Oxford Univ. Press, Oxford, 1993.
15. W. H. Press, S. A. Teukdsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C—The Art of Scientific Computing*, Second Ed., Cambridge Univ. Press, Cambridge, UK, 1992.
16. M. Rioux, F. Blais, J.-A. Beraldin and P. Boulanger, Range imaging sensors development at NRC laboratories, in *Proceedings, IEEE Workshop on Interpretation of 3D Scenes, Austin, TX Nov.* 1989, pp. 154–160.
17. T. Yoshimi and M. Oshima, Multi light sources range finder system, in *Proceedings, IAPR Workshop on CV, Tokyo, Oct. 1988*, pp. 245–248.