Recognizing Periodicities on 3D Scanned Meshes Based on Indexed-ICP Algorithm

Daisuke Kondo, Tomohiro Mizoguchi, Satoshi Kanai

Graduate School of Information Science and Technology, Hokkaido University
Kita-14, Nishi-9, Sapporo 060-0814, Japan
Phone +81-11-706-6449 / FAX +81-11-706-7210
E-mail: d_kondo@sdm.ssi.ist.hokudai.ac.jp
tmizoguchi@sdm.ssi.ist.hokudai.ac.jp
kanai@ssi.ist.hokudai.ac.jp

Abstract: Recently, meshes of engineering objects have been easily acquired by 3D laser or high energy industrial X-ray CT scanning systems, which are then widely used in product development. When utilizing the scanned meshes in object inspection, re-design, and simulation, it is very important to use them to reconstruct CAD models. Engineering objects often exhibit translational and rotational periodicities for their functionality. Therefore, it is essential to recognize such periodicities when reconstructing 3D CAD models that contain compact data representations. However, previous methods for reconstructing CAD models have not focused on recognizing such periodicities. In this paper, we propose a new method for recognizing translational and rotational periodicities based on the indexed-ICP algorithm. Our method robustly extracts a set of periodically displaced regions and the parameters defining the periodicity, such as translational basis vectors or a rotational axis and a basis angle, from scanned meshes. We demonstrate the effectiveness of this method from experiments using 3D laser and X-ray CT scanned meshes of engineering objects.

Key words: reverse engineering, CAD/CAM, periodicity, ICP

1- Introduction

3D laser scanning systems are widely used in product development to acquire geometric point cloud data from real-world engineering objects. More recently, high energy industrial X-ray CT scanning systems, which have been developed rapidly, have enabled users to quickly and non-destructively obtain 3D images of complex objects containing internal structures [S1]. The acquired data can be easily converted into a 3D mesh using well-known surface reconstruction algorithms, such as the marching cube [LH1]. Reconstructing CAD models from scanned meshes is very important to the inspection, re-design, and simulation of engineering objects.

The surface of engineering object often exhibit translational and rotational periodicities for their functionality. For example, as shown in Figure 1, in order to define 3D CAD models with translational and rotational periodicities, a 3D geometry of the base region of the periodicity, parameters defining periodicity, and translational / rotational "pattern" commands must be fed into the CAD system. A CAD model of a single base region can be reconstructed from its scanned mesh model using several well-known reverse engineering algorithms [CG1, KZ1]. However, previous reverse engineering methods for reconstructing CAD models [VF1, KF1, TO1, CG1, KZ1] have not focused on identifying the periodicities. Therefore, defining such CAD models automatically will require extracting a set of periodically displaced regions with the same geometries, as well as the parameters defining the periodicity, such as translational basis vectors or a rotational axis and a basis angle.

1.1 – Related works

Many algorithms have been proposed for recognizing periodicities in 2D images. Lin *et al.* [LW1] proposed an algorithm that recognizes a translational periodicity in a 2D texture based on the generalized Hough transform. Liu *et al.* [LC1] proposed an algorithm that recognizes a variety of periodicities, including translations, rotations, and reflections, based on the crystallographic theory. Müller *et al.* [MZ1] proposed an algorithm for extracting a translational periodicity from a 2D façade image in order to create a 3D model of a building by subdividing the image and evaluating the mutual information between the different subdivided images. However, such algorithms cannot be easily extended to recognize periodicities in 3D scanned meshes.

Periodicity recognitions have strong relationships with symmetry detections in the sense that they both find pairs of local shapes that can be matched to one another under certain transformations. Podolak *et al.* [PS1] proposed an algorithm to detect all possible planar reflective symmetries from a 3D

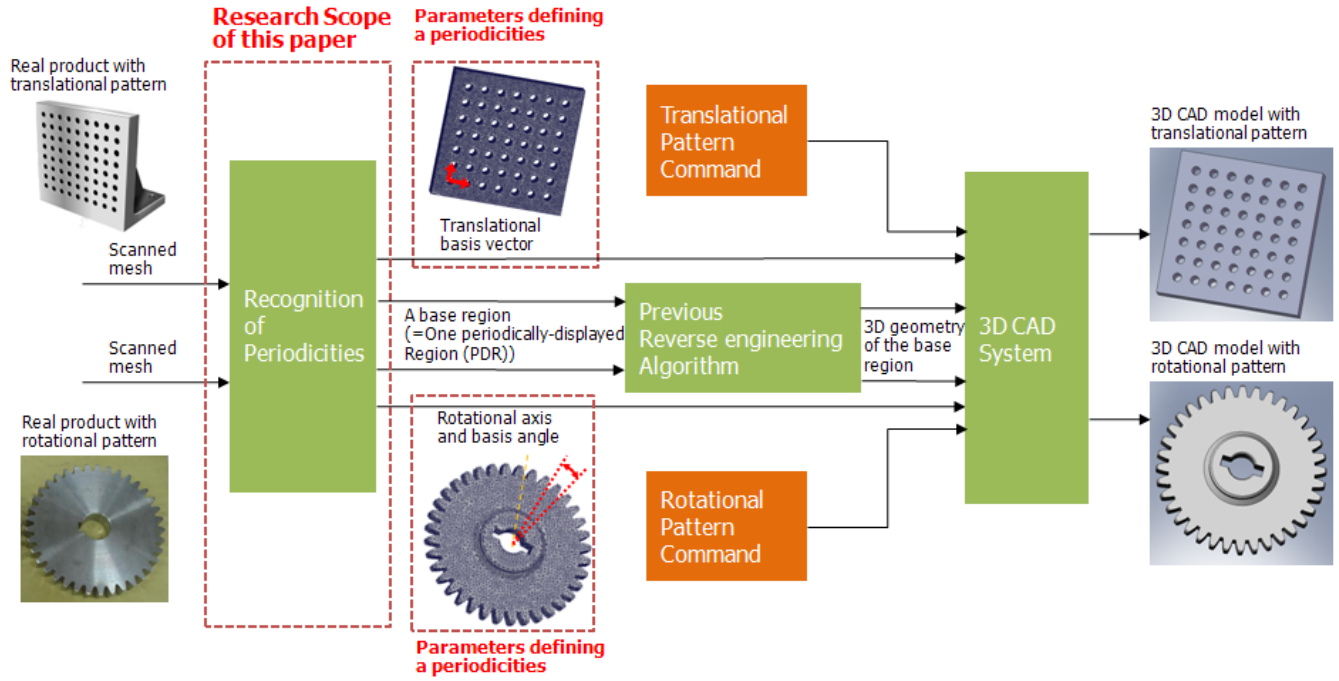


Figure 1: The significance of recognizing periodicities in engineering objects and the scope of our research

mesh based on the voting scheme. Mitra *et al.* [MG1] proposed an algorithm to detect partial approximate symmetries, including translation, rotation, reflection, and uniform scaling, from a 3D mesh. However, these algorithms only detect a pair of symmetric regions and their transformation, and thus cannot extract a set of periodic translations or rotation transformations that matches a single region to multiple regions at the same time.

Liu *et al.* [LM1] proposed an algorithm that extracts a single basis region from among the periodically displaced regions in a 3D mesh of a relief with user interaction. However, this algorithm cannot extract a set of periodically-displaced regions and the parameters defining the periodicity at the same time.

1.2 – Purpose and overview of our algorithm

As mentioned in section 1.1, no algorithm has yet been proposed that recognizes periodicities for the reconstruction of CAD models from 3D scanned meshes of engineering objects. Therefore, in this paper, we propose a new method for recognizing periodicities in scanned meshes based on the indexed-ICP algorithm. Our method extracts a set of periodically-displaced regions and the parameters defining the periodicity, such as translational basis vectors or a rotational axis and a basis angle from scanned meshes. It enables users to reconstruct 3D CAD models that contain compact data representations.

Our basic premise is that all of the periodically-displaced regions (PDRs) represent the same geometry, and that one of the PDRs can be matched to the other PDRs at the same time under a set of periodic translations or rotations.

Step1: Extraction of periodically-displaced regions and their planar parameterization (Section 2)

Our algorithm first estimates the mesh principal curvatures at each vertex based on a local quadratic polynomial surface fitting and detects any sharp edges [VS1,MD1]. Then, it segments meshes into regions that are bounded by sharp edges. Next, it uses the voting scheme and pairwise ICP algorithm to select a set of PDRs from among the segmented regions using the fact that all PDRs represent the same geometry. It then fits a plane for a set of barycenters of the selected PDRs. Finally, it projects these barycenters onto the plane and calculates the 2D parameters of the projected barycenters.

Step2: Extraction of initial parameters defining a periodicity and assignment of indices to PDRs on a 2D plane (Section 3)

At the beginning of this phase, the user selects either a translational or a rotational pattern for the recognition. If a translational pattern is selected, our algorithm extracts initial translational basis vectors based on Lin's method [LW1] and then assigns indices that specify the multiple of the translational basis vectors for each PDR. Alternatively, if a rotational pattern is selected, the algorithm extracts an initial rotational axis and a basis angle based on a modification of Lin's method [LW1] and then assigns an index that specifies the multiple of the rotational basis angle for each PDR.

Step3: Extraction of optimal parameters defining a periodicity on 3D mesh (Section 4)

The final step extracts optimal parameters defining the periodicity, such as translational basis vectors or a rotational axis and a basis angle based on the proposed indexed-ICP algorithm. Our algorithm calculates the parameters that match a single PDR to others using a 3D transformation specified by the indices of the PDRs assigned in Step 2 and the translational basis vectors or the rotational axis and the basis angle. Because our algorithm uses only vertex coordinates of the mesh and does not require any additional

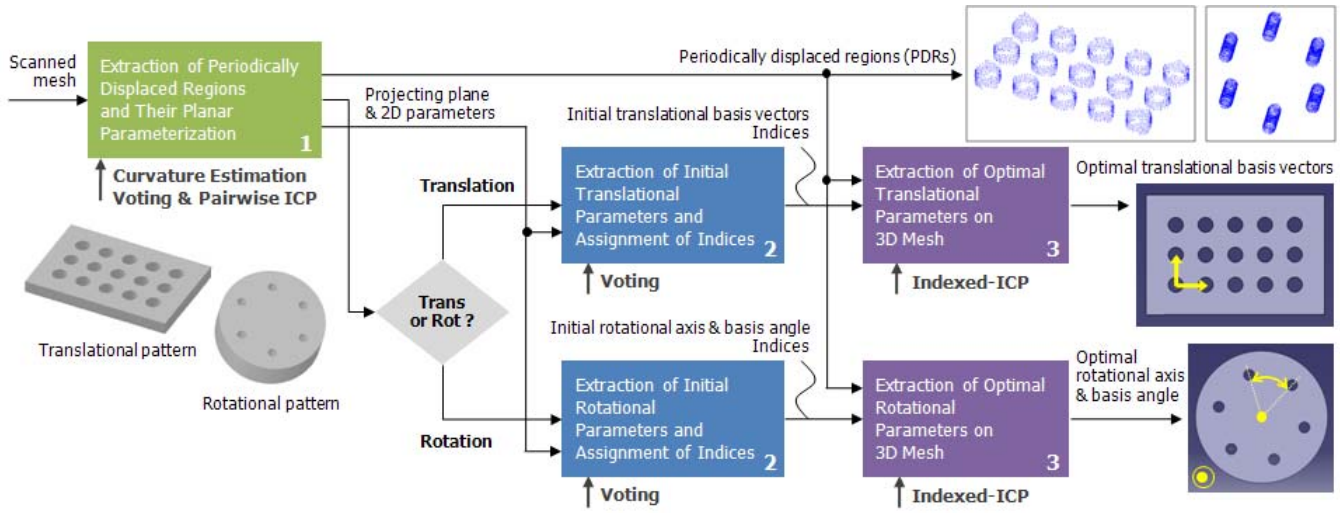


Figure 2: An overview of our algorithm

information in this step, such as curvatures or normal, the algorithm can robustly extract parameters from noisy scanned meshes. Even if the projecting plane obtained in Step1 contains some calculation error due to the scanning noise, our indexed ICP algorithm in Step3 compensates by reducing this error.

In this research, we assumed that there were more regions with periodicities in the scanned mesh, than without. We also assumed that the 3D geometries of all PDRs were roughly identical, and did not include any major flaws due to measurement fault. We also deal only with periodicities that could be created by a single 3D CAD system modeling operation. So far, our algorithm has only deal with periodically-displaced regions on planar surface.

2- Extraction and planar parameterization of periodically-displaced regions

2.1 – Curvature estimation and sharp edges extraction

To estimate mesh principal curvatures robustly on scanned meshes, our algorithm first fit the quadratic polynomial surface $h(u,v)$ in Eq. (1) for to set of vertices $N(i)$ around each vertex x_i .

$$h(u,v) = a_0u^2 + a_1v^2 + a_2uv + a_3u + a_4v + a_5 \qquad (1)$$

$N(i)$ was defined as the set of vertices topologically connected to the central vertex x_i , including x_i , within the specified Euclidean distance satisfying Eq. (2):

$$\|\mathbf{x}_j - \mathbf{x}_i\| < W \cdot l_{i,avg}, \qquad (2)$$

where $l_{i,avg}$ is the average length of the edges connected to x_i and W is the parameter specifying the neighboring size. We chose W=3.0 for all 3D scanned meshes with measurement noise based on our previous experiments. We then calculated the principal curvatures $\kappa_{i,\max}$ and $\kappa_{i,\min}$ at the corresponding point on $h(u,v)$ of x_i . Detailed algorithms of these processes

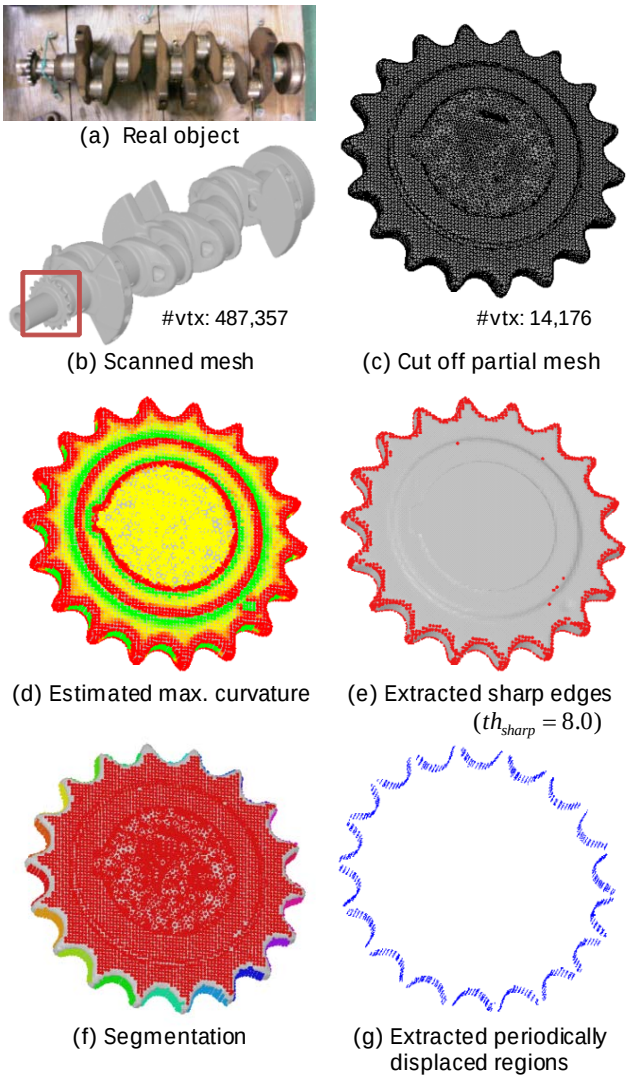


Figure 3: Extraction of periodically displaced regions

are precisely described in [VS1,MD1]. Next, our algorithm classified a vertex x_i if it satisfied the condition in Eq. (3) [VS1,MD1].

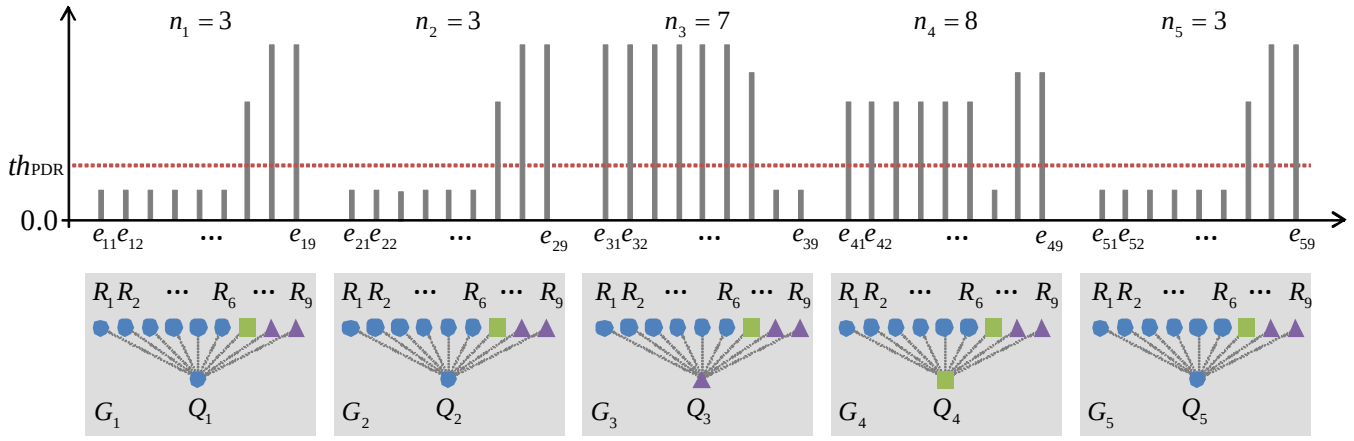


Figure 4: Selection of periodically-displaced regions by pairwise ICP algorithm

$$\frac{1}{|\kappa_{i,\max}|} < th_{sharp} \cdot l_{i,avg} \qquad (3)$$

The threshold th_{sharp} had to be set according to the geometry and the resolutions of the mesh. Intuitively, a larger th_{sharp} would induce a smaller number of sharp vertices, and vice versa. In our previous study [MD1], we found that this curvature estimation worked well even for noisy CT or laser scanned meshes.

Figure 3 shows an example of this process. The object in Figure 3(a) was scanned with an X-ray CT scanning system, and the mesh in Figure 3(b) was reconstructed from the scan. We then cut off the scanned mesh to obtain the partial mesh in Figure 3(c) and applied the algorithm to the partial. Figure 3(d) shows the estimated maximum curvatures and Figure 3(e) shows the extracted sharp edges.

2.2 – Extraction of periodically displaced regions

2.2.1 – Segmentation

Our algorithm next segmented a mesh into a set of regions $\{R_i\}$. A region is defined as a set of topologically-connected vertices that do not include sharp vertices. Example segmentation results are shown in Figure 3(f).

2.2.2 – Selection of PDRs

After the segmentation, our algorithm selected a set of periodically-displaced regions $\{PDR_j\}$ from $\{R_i\}$ in order to identify the periodicity. Each PDR_j represented the same geometry and their areas were therefore approximately equal. Under this assumption, our algorithm selected a PDR_j first by a voting scheme of the region's areas and next with a pairwise ICP algorithm [BM1,CM1]. We approximated the areas of the regions by the sums of the triangular areas that were connected to the vertices in these regions.

Selection of PDRs by voting scheme

First, our algorithm created voting bins which were divided by N_{area} and whose maximum value was the sum of the triangular areas in the mesh. Each region then cast a vote to its corresponding bin based on the sum of the triangular areas. Finally, the regions that cast votes to the bin with the most votes were selected as candidate PDRs. We set $N_{area} = 50$ for all meshes in this paper.

Selection of PDRs using a pairwise ICP algorithm

The voting scheme for the regions' areas mentioned in the previous section reduced the number of PDR candidates. However, non-PDRs still remained after this scheme is applied. Therefore, our method next used the pairwise *Iterative closest point* (*ICP*) algorithm [BM1,CM1] to accurately select PDRs from the remaining regions $\{R_j \,|1 \le j \le n^{rem}\}$. Here, we assumed that most of the remaining regions were PDRs. This method still works well even if we cannot identify all PDRs which should be recognized. The ICP algorithm matched pairs of regions so that the sums of the distances between the corresponding points were minimized. Details are described in section 4.1. All PDRs represented the same geometry, and pair of PDRs could therefore be matched by the ICP algorithm so as to minimize their average distances. However, when a PDR and a non-PDR were paired, the distance increased. Based on this fact, our method applied the following procedures for selecting PDRs.

Randomly select five regions $\{Q_i\}$ among $\{R_j\}$, and for each Q_i, create a set of pairs $G_i = \{\langle Q_i, R_j \rangle | \; 1 \le j \le n^{rem}\}$.

1. For each $\langle Q_i, R_j \rangle$, apply the ICP algorithm to match them and compute the averaged distance e_{ij} between the corresponding points.
2. For each G_i, count the number of false pairs n_i whose e_{ij} is more than the threshold th_{PDR}. We set th_{PDR} so that it became proportional to the averaged edge length l_{avg} of the mesh, such as $th_{PDR} = \tau_{PDR} l_{avg}$. We usually set $\tau_{PDR} = 1.0$.
3. If $n_i > \alpha n^{rem}$, verify that Q_i is not a PDR and then do not apply the following procedure to G_i. Otherwise, select Q_i as a PDR and then evaluate e_{ij} for each pair $\langle Q_i, R_j \rangle$ in G_i. If e_{ij} is less than th_{PDR}, select R_j as a PDR. Here we set $\alpha = 0.5$ and $\tau_{PDR} = 1.0$.

As a result of this process, a set of periodically-displaced regions $\{PDR_k\}$ were selected accurately, as shown in Figure 3(g). Figure 4 shows an example of this process. In this example, six PDRs (●) and three non-PDRs (■and ▲) are apparent in the mesh. First, three PDRs and two non-PDRs

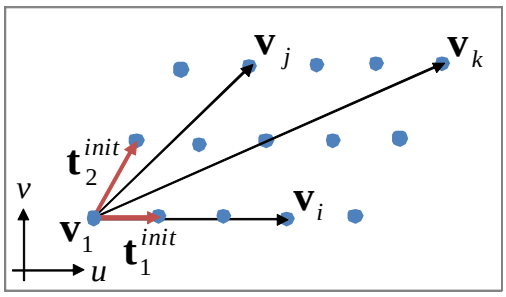


Figure 5: Extraction of initial translational parameters

(■ and ▲) were selected as $\{Q_i\}$, and then for each pair, $\langle Q_i, R_j \rangle$, e_{ij} was computed by applying the ICP algorithm. Since n_3 and n_4 were larger than the threshold αn^{rem}, Q_3 and Q_4 were designated as non-PDRs. Conversely, $\{R_i \mid 1 \le i \le 6\}$ were selected as PDRs.

3.2 – Margin

Next, our algorithm calculated the barycenter B_j of each PDR_j and fitted a least-squares plane to the set of B_j. Each B_j was then projected onto the plane and their parameters (u_j, v_j) were calculated on the plane. Axes *u* and *v* on the plane were arbitrarily defined so as to be orthogonal to one another.

3- Extraction of periodically-displaced regions and their planar parameterization

At the beginning of this step, we selected either a translational or a rotational pattern for periodicity recognition. Then, our algorithm calculated the initial parameters of the periodicity of the selected pattern.

A single PDR can be matched to others one at a time under a set of periodic translations or rotations. Therefore, a projected barycenter can also be approximately matched to others under the same transformations.

Under this assumption, our algorithm first calculated the initial parameters of a periodicity using Lin's method, described in 3.1. It then used the initial parameters to assign indices to each PDR. The advantage of this algorithm was its tolerance for the irregular sampling of projected barycenters, which were caused by differences in mesh vertex distribution.

3.1 – Extraction of initial translational parameters and assignment of indices to PDRs

In translational periodicities, a set of projected barycenters $\{PB_i\}$ form an approximate parallelogram grid on the projecting plane, as shown in Figure 5. This grid can be spanned by one or two translational basis vectors. Our algorithm used Lin's method [LW1] to find these vectors and assign indices to each PDR_i. This algorithm is summarized as follows:

Calculate the translational basis vectors:

1. Regard the projected barycenters $\{PB_i\}$ as a set of 2D vectors $\{\mathbf{v}_i = (u_i, v_i) 1 \le i \le n\}$ originating from $\mathbf{v}_1 = (0,0)$. Here n is the number of PDRs. Then, create a 2D accumulate array $S(i, j)$ and initialize entries of the array $S(i, j) = 0$ for $1 \le i, j \le n$. For each vector $\mathbf{v}_j$, where $2 \le j \le n$, perform the following procedures:

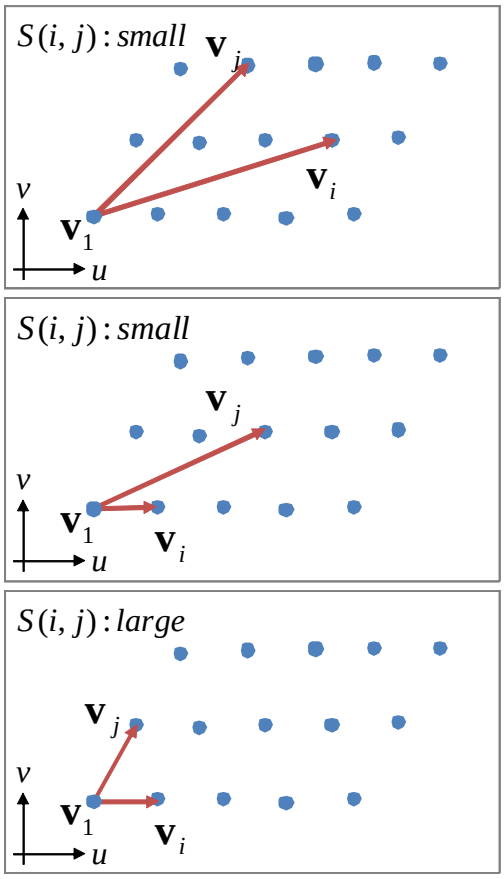


Figure 6: Examples of the $S(i, j)$

2-1. For each vector $\mathbf{v}_k$, where $2 \le k \le n$ and which is *collinear* to $\mathbf{v}_j$, compute the value of a using $a = |\mathbf{v}_k| / |\mathbf{v}_j|$. Let a' be the round integer of a.

2-2. Update the value of $S(1, j)$ using the following scoring rule:

$$S(1, j) = S(1, j) + \frac{1 - 2|a - a'|}{|\mathbf{v}_j|}$$

2. For each *non-collinear* pair of vectors $\mathbf{v}_i$ and $\mathbf{v}_j$, where $2 \le i < j \le n$, perform the following procedures as shown in Figure 5:

3-1. For each vector $\mathbf{v}_k$, where $2 \le k \le n$, compute the value of a and b using the following equation:

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} u_i & u_j \\ v_i & v_j \end{bmatrix}^{-1} \begin{bmatrix} u_k \\ v_k \end{bmatrix},$$

where a and b are coefficients of the linear combination of $\mathbf{v}_i$ and $\mathbf{v}_j$ for $\mathbf{v}_k$. Let a' and b' be the round integers of a and b respectively.

3-2. Update the value of $S(i, j)$ using the following scoring rule:

$$S(i, j) = S(i, j) + \frac{1 - 2\max(|a - a'|, |b - b'|)}{\max(|\mathbf{v}_i|, |\mathbf{v}_j|)}$$

3. Let the entry with the highest score in the array S located at $(\hat{i}, \hat{j})$. Set $\mathbf{u}_1 = \mathbf{v}_{\hat{i}}$, $\mathbf{u}_2 = \mathbf{v}_{\hat{j}}$, $\mathbf{u}_3 = \mathbf{u}_1 + \mathbf{u}_2$, and $\mathbf{u}_4 = \mathbf{u}_1 - \mathbf{u}_2$. Select the pair of vectors from $\langle \mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_4 \rangle$ with the smallest and the second smallest length. Call this pair of vectors $\mathbf{t}_1^{init}$ and $\mathbf{t}_2^{init}$.

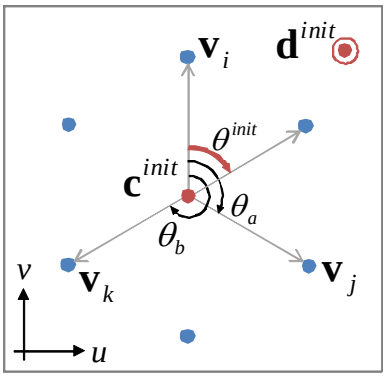


Figure 7: Extraction of initial rotational parameters

4. Output $\mathbf{t}_1^{init}$ and $\mathbf{t}_2^{init}$ as the initial translational basis vectors and then stop the process.

For each pair of vectors $\mathbf{v}_i$ and $\mathbf{v}_j$, if both vectors were short and located near a point of the parallelogram grid, their corresponding score $S(i, j)$ was high, as shown in Figure 6.

Index the PDRs:

For each vector $\mathbf{v}_i = (u_i, v_i)$, where $2 \le i \le n$, compute the value of c and d using the following equation:

$$\begin{bmatrix} c \\ d \end{bmatrix} = \begin{bmatrix} \mathbf{t}_1^{init} & \mathbf{t}_2^{init} \end{bmatrix}^{-1} \begin{bmatrix} u_i \\ v_i \end{bmatrix},$$

where c and d are coefficients of the linear combination of $\mathbf{t}_1^{init}$ and $\mathbf{t}_2^{init}$ for $\mathbf{v}_i$. Let c' and d' be the round integers of c and d respectively. Then, the indices of PDR_i are computed as (c', d').

3.2 – Extraction of initial rotational parameters and assignment of indices

In rotational periodicities, a set of projected barycenters $\{PB_i\}$ forms an approximate radial grid on the plane, as shown in Figure 7. The grid is spanned by a rotational basis angle around a center of rotation. To find this angle and the center, and to assign an index to each PDR_i, our algorithm used a modified version of Lin's algorithm [LW1]. This algorithm is summarized as follows:

Calculate the initial rotational axis:

1. Calculate the initial directional vector $\mathbf{d}^{init}$ of the rotational axis as the normal vector of the projecting plane.
2. For each triplet $\langle PB_i, PB_j, PB_k \rangle$ of the projected barycenters, calculate two perpendicular bisectors l_{ij} and l_{ik} and then compute the intersectional points $\mathbf{c}_{ijk}$ of l_{ij} and l_{ik}.
3. Calculate the averaged point of all $\mathbf{c}_{ijk}$ and regard it as both the center of the circle on the plane and the initial point $\mathbf{c}^{init}$ on the rotational axis.

Calculate the initial rotational basis angle:

1. Regard the projected barycenters as a set of vectors $\{\mathbf{v}_i = (u_i, v_i) \,|\, 1 \le i \le n\}$ originated by $\mathbf{c}^{init}$. Then create a 2D accumulate array $S(i, j)$ and initialize entries of the array $S(i, j) = 0$ for $1 \le i, j \le n$.

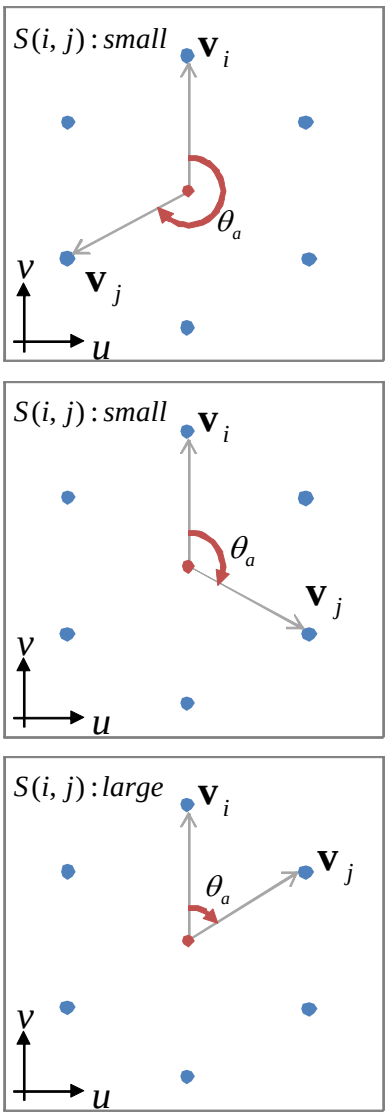


Figure 8: Examples of the $S(i, j)$

2. For each pair of vectors $\mathbf{v}_i$ and $\mathbf{v}_j$, where $2 \le i < j \le n$, perform the following procedures:
   2-1. For each vector $\mathbf{v}_k$, where $2 \le k \le n$, compute the value of a using $a = \theta_b / \theta_a$, where $\theta_a = ang(\mathbf{v}_i, \mathbf{v}_j)$ and $\theta_b = ang(\mathbf{v}_i, \mathbf{v}_k)$. Let a' be the round integer of a.
   2-2. Update the value of $S(i, j)$ using the following scoring rule:

$$S(i, j) = S(i, j) + \frac{1 - 2|a - a'|}{\sqrt{ang(\mathbf{v}_i, \mathbf{v}_j)}}$$

3. Let the entry with the highest score in the array S locate at $(\hat{i}, \hat{j})$ and let the initial rotational basis angle be $\theta^{init} = ang(\mathbf{v}_{\hat{i}}, \mathbf{v}_{\hat{j}})$.
4. Output θ^{init} as the initial rotational basis angle and then stop the process.

For each pair of vectors $\mathbf{v}_i$ and $\mathbf{v}_j$, if the vectors were located near the vertices of a radial grid and if their angle was small, their corresponding score was high, as shown in Figure 8.

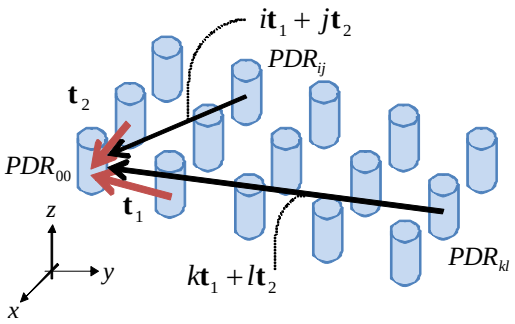


Figure 9: Indexed-ICP algorithm for Translational pattern

Index the PDRs:
For each vector $\mathbf{v}_i = (u_i, v_i)$, where $\mathbf{v}_i = (u_i, v_i)$, compute the value of c using $c = \theta_c / \theta^{init}$, where $\theta_c = ang(\mathbf{v}_i, \mathbf{v}_1)$. Let c' be the round integer of c. Then, the PDR_i index is computed as c'.

4- Extraction of optimal parameters of periodicity from 3D meshes based on indexed-ICP algorithm

4.1 – ICP algorithm

The *Iterative closest point* (*ICP*) *algorithm* was first proposed by Besl and McKay [BM1] and Chen and Medioni [CM1] for matching a pair of scanned data $\langle X, Y \rangle$, and many variants have since been proposed [RL1]. In the ICP algorithm, the optimal transformation can be found where the matching error between corresponding points is minimized. If we denote such a transformation $\mathbf{x}_i' = \mathbf{t}_{XY} + \mathbf{R}_{XY}\mathbf{x}_i$ as $\mathbf{T}_{XY} = \langle \mathbf{t}_{XY}, \mathbf{R}_{XY} \rangle$, where $\mathbf{x}_i \in X$, it can be calculated by the following procedure:

1. **Initialize:** Compute the initial transformation $\mathbf{T}_{XY}^{init}$, and set $itr = 0$.
2. **Find closest points:** For each point $\mathbf{x}_i^{itr}$ in the current position of the data *X*, find the point $\mathbf{y}_{c(i)}^0$ in *Y* that is closest to $\mathbf{x}_i^{itr}$.
3. **Compute a transformation:** Compute the transformation $\mathbf{T}_{XY}^{itr}$ so that the sum of matching errors between corresponding points is minimized. The objective function to be minimized is:

$$F^{itr} = \sum_{i=1}^{N} \left\| \mathbf{y}_{c(i)}^0 - \mathbf{x}_i^{itr+1} \right\|^2. \qquad (4)$$

4. **Update points and calculate error:** Update points using the current transformation $\mathbf{T}_{XY}^{itr}$ such that $\mathbf{x}_i^{itr+1} = \mathbf{t}_{XY}^{itr} + \mathbf{R}_{XY}^{itr}\mathbf{x}_i^0$, then calculate the current average matching error as $E^{itr} = (F^{itr})^{1/2}/N$.
5. **Termination condition:** If $E^{itr} - E^{itr+1} > \varepsilon$, update $itr \leftarrow itr + 1$ and repeat the process from step 2. Otherwise, output the current transformation $\mathbf{T}_{XY}^{itr}$ as the optimal $\mathbf{T}_{XY}^{opt}$, and stop the process.

This ICP algorithm can calculate a transformation that matches a single pair of scanned data. However, it cannot extract a set of periodic transformations simultaneously that match single data to multiple data which are periodically displaced in groups. Therefore, we propose an indexed-ICP algorithm, which calculates such sets of periodic transformations.

4.2 – Our indexed-ICP algorithm

In this section, we describe our indexed-ICP algorithm which extracts optimal parameters for defining translational or rotational periodicities from 3D scanned meshes.

4.2.1 – Extraction of optimal translational parameters

In translational cases, a single PDR can be matched to other PDRs at the same time under a set of periodic translations along the basis vectors $\mathbf{t}_1$ and $\mathbf{t}_2$. Here we denote PDR_{ij} as $PDR_{ij} = \{\mathbf{x}_{ij,k} \mid 0 \le i \le N_1, 0 \le j \le N_2, 1 \le k \le n_{ij}\}$, where N_1 and N_2 are the maximum indices for $\mathbf{t}_1$ and $\mathbf{t}_2$ respectively. The point $\mathbf{x}_{ij,k}$ can be matched to its corresponding point $\mathbf{x}_{00,c(k)}$ such that $\mathbf{x}_{00,c(k)} = i\mathbf{t}_1 + j\mathbf{t}_2 + \mathbf{x}_{ij,k}$, where i and j are the indices calculated in Step 3.1 that specify the multiples of the translational basis vectors. To find such vectors, our indexed-ICP algorithm for the translational pattern performs the following procedure:

1. **Initialize:** Set $\mathbf{t}_1^0 = -\mathbf{t}_1^{init}$, $\mathbf{t}_2^0 = -\mathbf{t}_2^{init}$ and $itr = 0$.
2. **Find closest points:** For each PDR_{ij}, not including PDR_{00}, perform the following process:
   For each point $\mathbf{x}_{ij,k}^{itr}$ in the current position of PDR_{ij}, find the point $\mathbf{x}_{00,c(k)}^0$ closest to $\mathbf{x}_{ij,k}^{itr}$ that satisfies the normal constraint in Eq.(5):

$$\cos^{-1}(\mathbf{n}_{00,c(k)}^0 \cdot \mathbf{n}_{ij,k}^{itr}) < th_{nrm}, \qquad (5)$$

   where $\mathbf{n}_{00,c(k)}^0$ and $\mathbf{n}_{ij,k}^{itr}$ are the outward unit normal vectors at $\mathbf{x}_{00,c(k)}^0$ and $\mathbf{x}_{ij,k}^{itr}$. We typically set $th_{nrm} = 15.0[\text{deg}]$.
3. **Compute translational vectors:** Compute the current translational basis vectors $\mathbf{t}_1^{itr}$ and $\mathbf{t}_2^{itr}$ by minimizing the sum of matching errors between corresponding points in Eq.(6):

$$F^{itr} = \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} \sum_{k=1}^{n_{ij}} \left\| \mathbf{x}_{00,c(k)}^0 - \{ i\mathbf{t}_1^{itr} + j\mathbf{t}_2^{itr} + \mathbf{x}_{ij,k}^0 \} \right\|^2. \qquad (6)$$

   This can be expressed as a parallel linear equation and can be solved for $\mathbf{t}_1^{itr}$ and $\mathbf{t}_2^{itr}$ using *Gauss-Jordan elimination* [PT1].
4. **Update points and calculate error:** Update the points using the current vectors $\mathbf{t}_1^{itr}$ and $\mathbf{t}_2^{itr}$, such that $\mathbf{x}_{ij,k}^{itr+1} = i\mathbf{t}_1^{itr} + j\mathbf{t}_2^{itr} + \mathbf{x}_{ij,k}^0$. Then, calculate the average error as $E^{itr+1} = (F^{itr})^{1/2} / N_{all}$, where N_{all} is the sum of the number of vertices in $\{PDR_{ij}\}$.

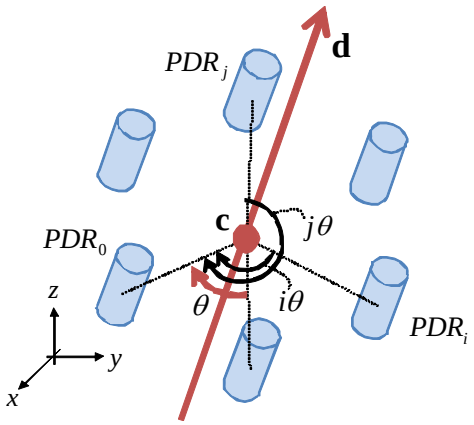


Figure 10: Indexed-ICP algorithm for rotational pattern

5. **Termination condition:** If $E^{itr} - E^{itr+1} > \varepsilon^{trans}$, update $itr \leftarrow itr + 1$ and repeat the process from Step 2. Otherwise, output the optimal translational basis vectors as $\mathbf{t}_1^{opt} = -\mathbf{t}_1^{itr}$ and $\mathbf{t}_2^{opt} = -\mathbf{t}_1^{itr}$, and stop the process. We set the tolerance ε^{trans} proportional to the average edge length l_{avg} such that $\varepsilon^{trans} = \alpha l_{avg}$ typically with $\alpha = 0.001$.

4.2 – Extraction of optimal rotational parameters

In rotational cases, a single PDR can be matched to other PDRs simultaneously under a set of periodic rotations around an axis $\mathbf{a} = \langle \mathbf{d}, \mathbf{c} \rangle$, where $\mathbf{d}$ is the axis directional vector and $\mathbf{c}$ is an arbitrary point on the axis. Here, we denote PDR_i as $PDR_i = \{\mathbf{x}_{i,k} \,|\, 0 \le i \le N, , 1 \le k \le n_i\}$. The point $\mathbf{x}_{i,k}$ can be matched to its corresponding point $\mathbf{x}_{0,c(k)}$ such that $\mathbf{x}_{0,c(k)}$, where i is the index calculated in Step 3.2 that specifies the multiple of the basis angle. To find such an axis and basis angle, our indexed-ICP algorithm for the rotational pattern performs the following procedure:

1. **Initialize:** Set $\mathbf{d}^0 = \mathbf{d}^{init}$, $\mathbf{c}^0 = \mathbf{c}^{init}$, $\theta^0 = -\theta^{init}$ and $itr = 0$.
2. **Find closest points:** For each PRR_i, where $1 \le i \le N$, perform the following process:
   For each point $\mathbf{x}_{i,k}^{itr}$ in the current position of PRR_i, find the point $\mathbf{x}_{0,c(k)}^0$ closest to $\mathbf{x}_{i,k}^{itr}$ that satisfies the normal constraint in Eq.(7):

$$\cos^{-1}(\mathbf{n}_{0,c(k)}^0 \cdot \mathbf{R}_i(i\theta^{itr}, \mathbf{a}^{itr})\mathbf{n}_{i,k}^{itr}) < th_{nrm}, \qquad (7)$$

   where $\mathbf{n}_{0,c(k)}^0$ and $\mathbf{n}_{i,k}^{itr}$ are the outward unit normal vectors at $\mathbf{x}_{0,c(k)}^0$ and $\mathbf{x}_{i,k}^{itr}$. (We typically set $th_{nrm} = 15.0$[deg].)
3. **Compute rotational axis and basis angle:** Compute the directional vector $\mathbf{d}^{itr}$, the point on the axis $\mathbf{c}^{itr}$, and the basis angle θ^{itr} by minimizing the sum of matching errors between corresponding points in Eq.(8):

$$G^{itr} = \sum_{i=1}^{N-1} \sum_{k=1}^{n_i} \left\| \mathbf{x}_{0,c(k)}^0 - \mathbf{R}(i\theta^{itr}, \mathbf{a}^{itr})\mathbf{x}_{i,k}^0 \right\|^2. \qquad (8)$$

   This non-linear equation can be solved for $\mathbf{d}^{itr}$, $\mathbf{c}^{itr}$, and θ^{itr} using *Levenberg-Marquardt algorithm* [PT1].
4. **Update points and calculate error:** Update the points using the current rotational parameters $\mathbf{d}^{itr}$, $\mathbf{c}^{itr}$ and θ^{itr}, such that $\mathbf{x}_{i,k}^{itr+1} = \mathbf{R}(i\theta^{itr}, \mathbf{a}^{itr})\mathbf{x}_{i,k}^0$. Then calculate the average error as $E^{itr+1} = (G^{itr})^{1/2} / N_{all}$, where N_{all} is the sum of the number of vertices in $\{PDR_i\}$.
5. **Termination condition:** If $E^{itr} - E^{itr+1} > \varepsilon^{rot}$, update $itr \leftarrow itr + 1$ and repeat the process from Step 2. Otherwise, output the optimal rotational parameters as $\mathbf{d}^{opt} = \mathbf{d}^{itr}$, $\mathbf{c}^{opt} = \mathbf{c}^{itr}$ and $\theta^{opt} = -\theta^{itr}$, and stop the process. (We set ε^{rot} such that $\varepsilon^{rot} = \beta l_{avg}$, typically with $\beta = 0.001$.)

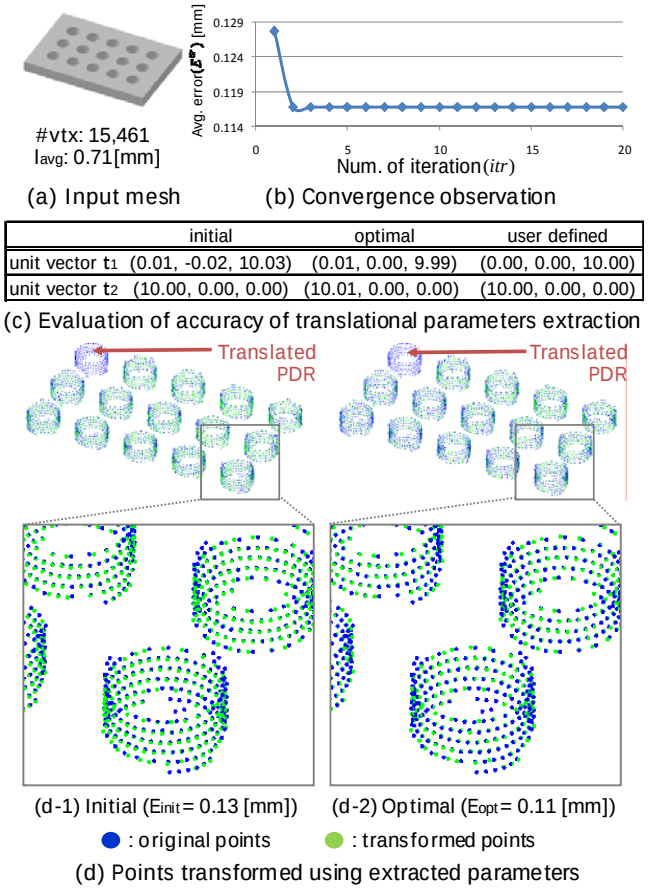
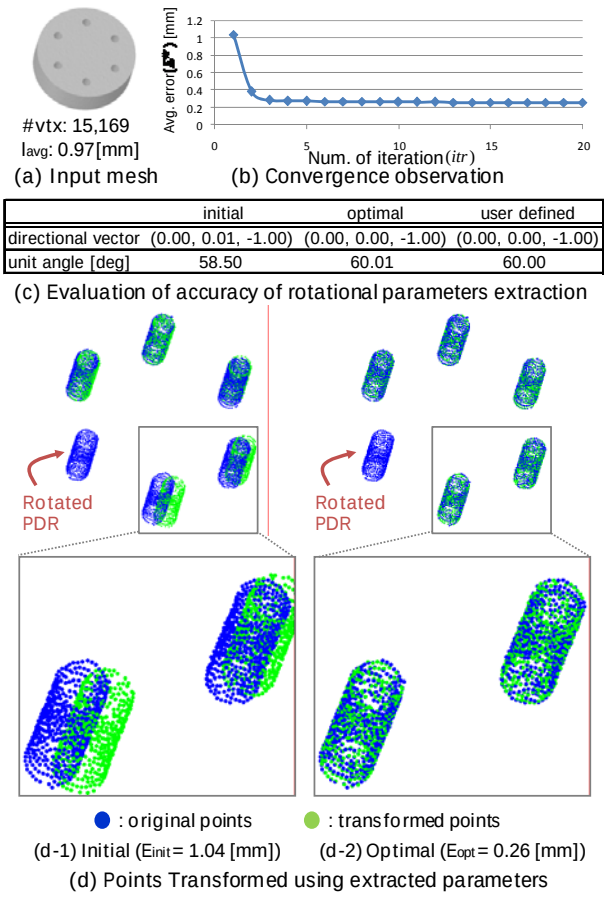


Figure 11: Evaluation of the accuracy of translational parameter extraction

5- Results

5.1 – Verification of parameters extraction results

We have evaluated the accuracy of the parameters of periodicity obtained using our proposed algorithm. For the evaluation, we used the meshes in Figures 11(a) and 12(a), which were created by triangulating the CAD models using the FEM meshing tool. Then, we added artificial noise to these meshes by moving each vertex along its normal direction by a Gaussian-distributed random distance with 5% standard deviation proportional to the average mesh edge length. The meshes in Figures 11(a) and 12(a) contained about 20,373 and 15,169 vertices, and their average edge lengths were 0.70 and 0.97 mm, respectively.



	initial	optimal	user defined
directional vector	(0.00, 0.01, -1.00)	(0.00, 0.00, -1.00)	(0.00, 0.00, -1.00)
unit angle [deg]	58.50	60.01	60.00

(c) Evaluation of accuracy of rotational parameters extraction

Fig. 12: Evaluation of the accuracy of rotational parameter extraction

Figures 11(b-d) and 12(b-d) show the accuracy evaluation results for the extracted translational and rotational parameters, respectively. As shown in Figures 11(b) and 12(b), our indexed-ICP algorithm gradually reduced the average error E^{itr} between corresponding points. Figures 11(c) and 12(c) show the extracted parameters. The initial parameters were appropriately optimized so as to be similar to the correct user-defined parameters. As shown in Figures 11(d) and 12(d), we translated and rotated one PDR against the others according to the extracted periodicity parameters and their indices. This demonstrates that a PDR can be accurately transformed to another according to the assigned indices and extracted parameters. The average errors were reduced from 0.13 mm to 0.11 mm, and from 1.04 mm to 0.26 mm, respectively, by the indexed-ICP algorithm.

5.2 – Experimental results for scanned meshes

We applied our proposed algorithm to 3D laser and X-ray CT scanned meshes of real-world objects. All of the experiments were processed on a Core2 Quad 2.4GHz CPU. Figure 13 shows the results of the translational periodicity recognition. We first scanned the LEGO block in Figure 13(a) using a 3D laser scanning system and then reconstructed the mesh in Figure 13(b) from part of the scanned data. The average edge length of the mesh was 0.93 mm. As shown in Figure 13(d), our method was able to accurately extract all of the PDRs without extracting any non-PDRs. Figure 13(c) shows the

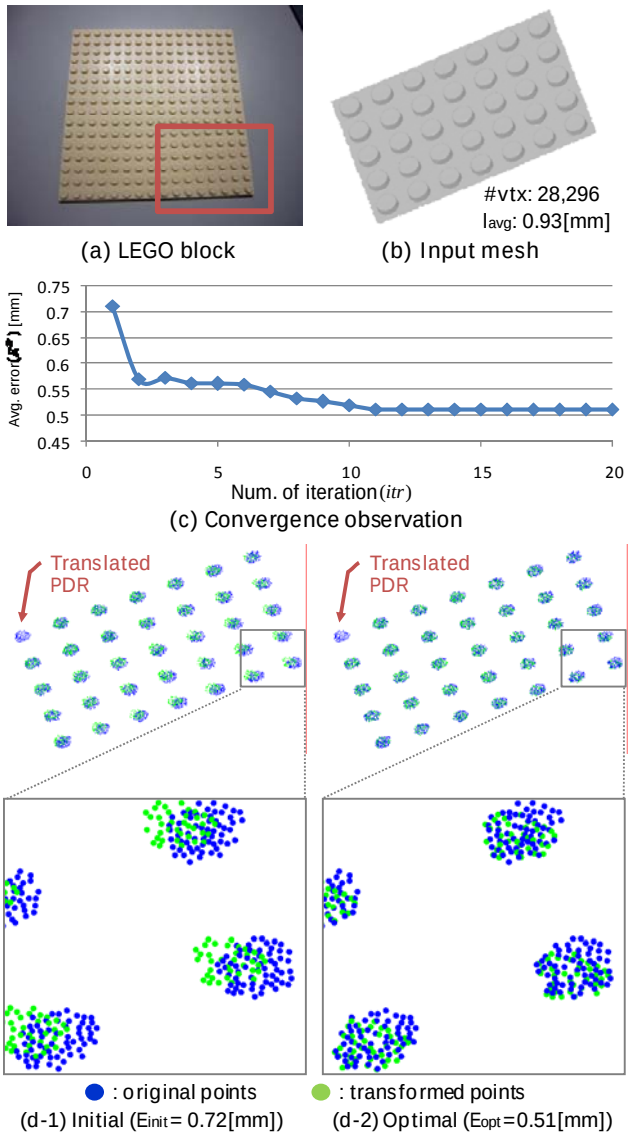


Fig 13: Laser scanned mesh results

convergence observation of the indexed-ICP. The average matching error between corresponding points was gradually reduced from 0.72 to 0.51 mm. Figure 13(d) shows the points transformed according to the extracted initial and optimal parameters defining the translational periodicity with indices. This shows that PDRs in the cylindrical region were more appropriately transformed so that pairs of PDRs were closer together, according to the optimal parameters, than the initial ones. Additionally, it shows that our indexed-ICP algorithm was able to accurately extract the translational parameters. In this model, each PDR contained about 70 vertices and the total running time was 4.84 sec. Figures 14 and 15 show the results of the rotational periodicity recognition using the X-ray CT scanned meshes. The mesh in Figure 14 is the same as that used in Figure 3. The average edge lengths were about 1.02 and 1.00 mm, respectively. As shown in Figures 14(b) and 15(d), our method was able to accurately extract all of the PDRs without extracting any non-PDRs. Figures 14(a) and 15(c) show the average matching errors observed in the indexed-ICP. The errors were gradually reduced to satisfactorily small values with reference to the average edge

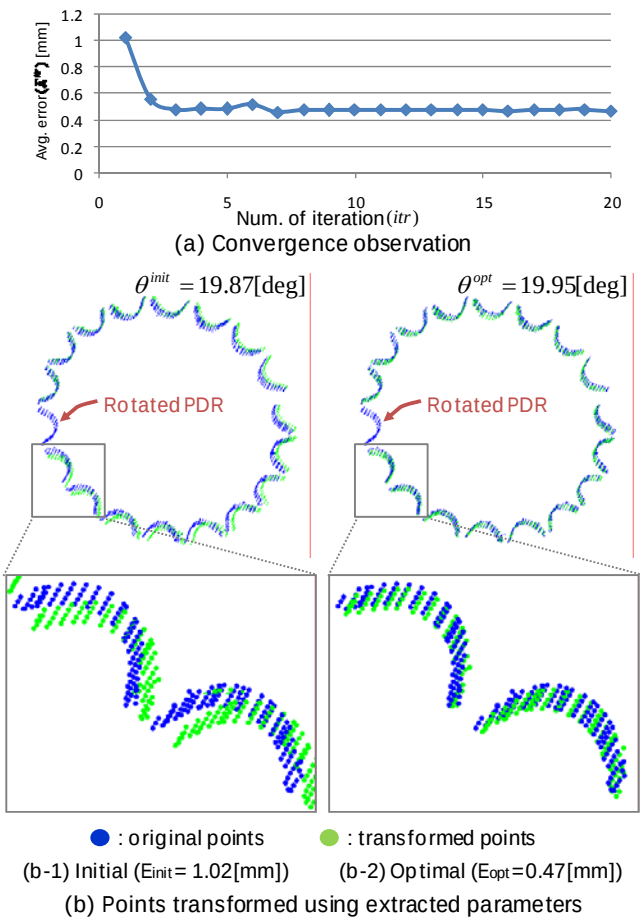


Fig. 14: X-ray CT scanned mesh results

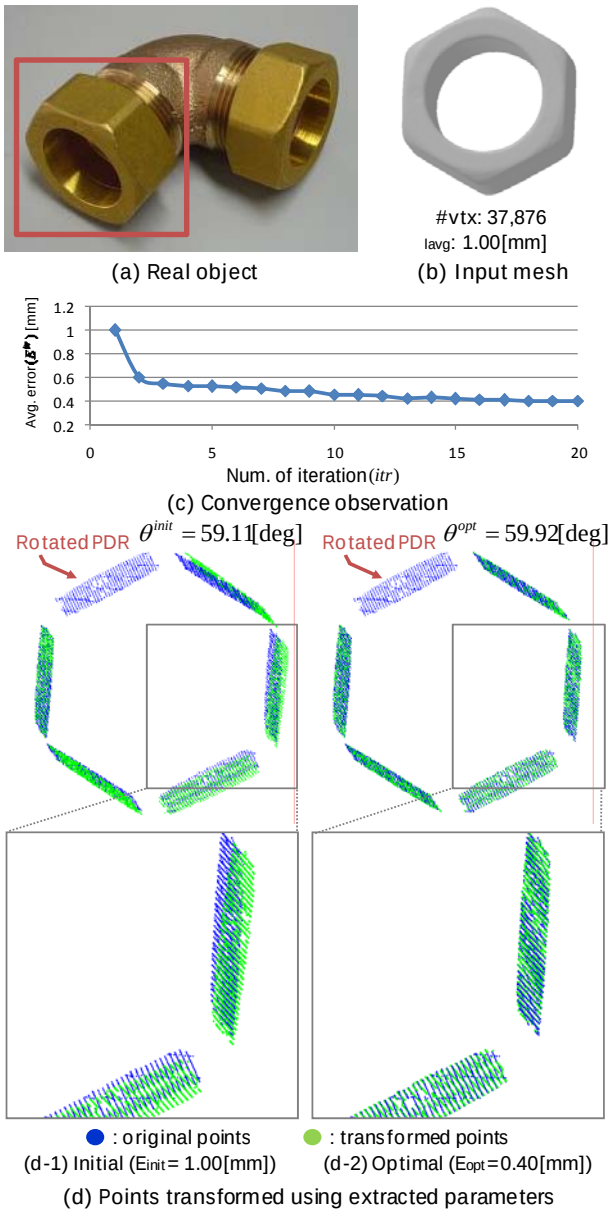


Fig 15: X-ray CT scanned mesh results

lengths. Figures 14(b) and 15(d) show the transformation results using the extracted initial and optimal rotational parameters with indices. These show that the PDRs were appropriately transformed so as to be close together according to the optimal parameters, and that our indexed-ICP algorithm was able to accurately extract the rotational parameters. In these models, each PDR contained about 100 and 1,300 vertices, and the total running times were 5.28 and 52.39 sec, respectively.

6- Discussion

Assumption. In this work, we assume that meshes are noisy, but the edge lengths of the meshes are relatively constant and therefore the vertices are approximately uniformly distributed on the surfaces. All the meshes in this paper follows this assumption. If meshes are reconstructed by marching cube based algorithms from the X-ray CT scanned data, they satisfy such properties (i.e. constant edge lengths and equal vertices distributions). Users may have obtained scanned meshes that do not follow the assumption. In such a case, users can remesh them and then can apply our method to recognize the periodicities.

Limitations. Our algorithm contains several limitations and the main two are as follows: First, it cannot detect periodicities if one correct PDR is over-segmented into several regions. We will overcome this limitation by finding spatially neighboring regions each of which belongs to a different periodicity but has similar periodic parameters, and then by appropriately combining them into a single PDR. Second, our method cannot detect periodicities from meshes that do not include sharp edges such as free-form shapes because such meshes cannot be appropriately segmented into regions.

7- Conclusion and future works

We have proposed a method for recognizing 3D periodicities based on the indexed-ICP algorithm in order to enable the reconstruction of CAD models from scanned meshes of engineering objects. By applying the proposed method to various CAD triangulated and scanned meshes, we demonstrated that our method could accurately and robustly recognize translational and rotational periodicities in meshes. In future works, we will attempt to improve the extraction component of the current algorithm for multiple regions. Our current method deals only with PDRs, each of which can be

segmented into a single region. However, in some meshes, a PDR can be segmented into multiple regions. For such cases, our method needs to first combine the multiple regions into a single set of regions and then apply the proposed method for periodicity recognition. Moreover, we will try to recognize periodicities on thin, plate-like meshes, such as mobile phone housings, where each PDR contains vertices only at regional boundaries.

Acknowledgments

This work was financially supported by a grant-in-aid under the Project No. 19360067. We would like to thank Ichiro Nishigaki, Tatsuro Yashiki and Noriyuki Sadaoka of HITACHI Co., Ltd., for providing the X-ray CT scanned meshes in Figures 3, 14, and 15. We would also like to thank Akira Manjome of the Hokkaido Industrial Research Institute for providing the 3D laser scanned mesh in Figure 13.

REFERENCES

[BM1] Besl P. and McKay N. A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2): 239-256, 1992

[CG1] Chaperon T. and Goulette F. Extracting Cylinders in full 3D Data Using a Random Sampling Method and the Gaussian Image. Proceeding of the Vision Modeling and Visualization Conference, 35-42, 2001

[CM1] Chen Y. and Medioni G. Object modeling by registration of multiple range images. Image and Vision Computing, 10(3): 145-155, 1992

[KF1] Ke Y., Fan S., Zhu W., Li A., Liu F. and Shi X. Feature-based reverse modeling strategies. Computer-Aided Design, 38(5): 485-506, 2006

[KZ1] Ke Y, Zhu W, Liu G. and Shi X. Constrained fitting for 2D profile-based reverse modeling, Computer-Aided Design, 38(2):104-114, 2005

[LC1] Liu, Y., Collins, T.T. and Tsin Y. A computational model for pattern perception based on frieze and wallpaper groups. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(3): 354-371, 2006

[LH1] Lorensen W. E. and Harvey E. C. ACM SIGGRAPH Computer Graphics, 21(4): 163-169, 1987

[LM1] Liu S., Martin R.R., Langbein F.C. and Rosin P.L. Segmenting Periodic Reliefs on Triangle Meshes. Lecture Notes in Computer Science, 4647, 290-306, 2007

[LW1] Lin H.C., Wang L.L. and Yang S.N. Extracting periodicity of a regular texture based on autocorrelation functions. Pattern Recognition Letters, 18: 433-443, 1997

[MD1] Mizoguchi T., Date H., Kanai S. and Kishinami T. Quasi-optimal mesh segmentation via region growing/merging. Proc. of ASME/DETC, No.35171. 2007

[MG1] Mitra N.J., Guibas L.J. and Pauly M. Partial and approximate symmetry detection for 3D geometry. ACM Transactions on Graphics, 25(3): 560-568, 2006

[MZ1] Müller P., Zeng G., Wonka P. and Gool L.V. Image-based procedural modeling of facades. ACM Transactions on Graphics, 26(3): No.85, 2007

[PS1] Podolak J., Shilane P., Golovinskiy A., Rusinkiewicz S. and Funkhouser T. A planar-reflective symmetry transform for 3D shapes. ACM Transactions on Graphics, 25(3): 549-559, 2006

[PT1] Press W.H., Teukolsky S.A. "Vetterling W.T. and Flannery B.P. Numerical Recipes in C++: The Art of Scientific Computing, Second Edition," Cambridge University Press, 1992

[RL1] Rusinkiewicz S. and Levoy M. Efficient variants of the ICP algorithm. Proc. of International Conference on 3D Digital Imaging and Modeling, 145-152, 2001

[S1] Suzuki H. Convergence engineering based on X-ray CT scanning technologies. Proc. of JSME Digital Engineering Workshop, 74-77, 2005

[TO1] Thompson W.B., Owen J.C. de St. Germain H.J., Stark S.R. and Jr. Henderson T.C. Feature-based reverse engineering of mechanical parts. IEEE Transactions on Robotics and Automation, 57-66, 1999

[VF1] Várady T., Facello M.A. and Terék Z. Automatic extraction of surface structures in digital shape reconstruction. Computer-Aided Design, 39(5): 379-388, 2007

[VS1] Vieira M. and Shimada K. Surface mesh segmentation and smooth surface extraction through region growing. Computer-Aided Geometric Design, 22(8): 771-792, 2005

