Improvement of Shape Accuracy
with Fitting Silhouette lines to a Sinogram

Trung Vuong Pham¹, Yutaka Ohtake¹, Hiromasa Suzuki¹

¹The University of Tokyo, Japan, E-mail: {pham, yu-ohtake, suzuki}@den.rcast.u-tokyo.ac.jp

Abstract
In recent years, the widespread and improvement of the performance of industrial X-ray CT scanners has led to an increasing demand for achieving accurate surface reconstruction of measured objects. However, it is often difficult to accurately reconstruct the shape of a measured object, which is caused by CT artifacts such as metal artifacts, beam hardening, and cone-beam artifacts. This paper proposes a novel method that reuses X-ray projection images, or a sinogram, to improve the accuracy of the reconstructed shape by correcting the positions of surface mesh vertices, which is done by fitting silhouette lines of the mesh to the sinogram. Reusing the sinogram has the potential application for improving the accuracy of the reconstructed mesh.

Keywords: 3D shape extraction, shape silhouette lines, shape correction.

1 Introduction
The blue arrows of Figure 1 shows the basic principle of the CT process used in this study. As a result of X-ray scanning, a sequence of 2D gray scale images captured from different angles, known as a sinogram, is obtained on the detector. On the basic of this sinogram, a CT volume is generated by a CT reconstruction algorithm [1]. Once the volumetric data is obtained, an isosurface mesh is typically extracted by using polygonization algorithms such as Marching Cubes [2]. When using this mesh for geometric dimensioning and tolerancing (GD&T), its geometrical accuracy is a critical parameter. However, in some cases, the isosurface mesh reconstruction of objects is not sufficiently accurate because the CT volume is not accurately generated. The sinogram is conventionally used only for reconstructing the CT volume. However, we propose a novel method for improving the accuracy of the reconstructed mesh by reusing the sinogram as shown in the red arrows of Figure 1.

Figure 1. Algorithm overview. The blue arrows show the conventional CT process and the red arrows indicate our proposed method.
The basic concept of the proposed method is to estimate errors of mesh vertices in the normal directions by comparing silhouette lines of the surface mesh with the sinogram. On the basis of the estimated errors, the mesh vertices are then moved along the normal directions. Moreover, we aim to achieve sub-pixel positional accuracy of the mesh vertices by interpolating the sinogram when computing fitting errors of the silhouette lines.

The paper is organized as follows. In Section 2, we introduce a detailed algorithm of the proposed method. In Section 3, we show our experimental results and discuss the current limitations of the method. Finally, we give an overall conclusion and outline directions for future work in Section 4.

2 Method

2.1 Silhouette lines extraction

The silhouette lines on a surface are defined as the set of points satisfying

$$d(p) \equiv n(p) \cdot (p - c) = 0,$$

where $p$ is a point on the surface, $n(p)$ is the unit normal vector at $p$, and $c$ is a perspective viewpoint [3]. In other words, the silhouette lines comprise the set of points on the surface whose normal vectors are perpendicular to the viewing direction (Figure 2(a)). Considering the focal point of the X-ray source as the viewpoint, the procedure of obtaining the silhouette is as follows:

1. For each mesh triangle consisting of vertices $p_A$, $p_B$, and $p_C$, the dot products $d(p_A)$, $d(p_B)$, and $d(p_C)$ are computed, respectively. Then the silhouette points are located where $d(p) = 0$ by linearly interpolating the values of $d$ on the triangle. For example, silhouette points $a$ and $b$ in Figure 2(b) are located as such that

$$a = \frac{|d(p_A)|p_B + |d(p_B)|p_A}{|d(p_A)| + |d(p_B)|}, \quad b = \frac{|d(p_A)|p_C + |d(p_C)|p_A}{|d(p_A)| + |d(p_C)|}.$$  

Next, line segments are created by connecting the points to form the silhouette lines (Figure 2(c)). Normal vectors of silhouette points are also computed for the next step.

2. In order to obtain the 2D silhouette lines and their normal vectors on the detector plane, perspective projection is performed for the 3D silhouette lines and their normal vectors determined in Step 1.

![Image](2.png)

Figure 2. Silhouette lines extraction. (a) The concept of silhouette lines extraction. (b) Silhouette points computed by linear interpolation. (c) Silhouette lines created by connecting the line segments.

2.2 Fitting-error computation

We estimate the planar errors by comparing the silhouette line on the detector with the projection image which has a same projection angle. According to the aforementioned definition, the silhouette lines correspond to the edges of the projection image; therefore, the fitting errors are the differences between the edges and the silhouette lines. Moreover, because of the inherent property of the projection image, the gradient of the projection value at every edge point changes abruptly (Figure
3(b)). On the basis of this property, the edge point is considered as an intersection of two fitting lines of sampling points adjacent in the normal direction of each silhouette point (Figure 3(c)).

The procedure for fitting-error computation is as follows: First, we put the silhouette lines on the detector and the projection image having the same projection angle in the same coordinate (Figure 4(a)). Then, \( q \) pixels close to each silhouette point in the normal vector direction are selected, and \( h \) sampling points for each pixel are generated (Figure 4(b)). Projection values of sampling points are computed by bilinear interpolation. Assuming that the number of sampling points is \( n \), we slide a point of interest \( x = k \) in the range of \([3, n - 3]\) to divide the sampling points into two subsets: \([1, k]\) and \([k + 1, n]\). For each group, we fit a straight line to the data by using the least square method [4]. Assuming that the fitting lines are \( y_1 = a_1 x + b_1 \) and \( y_2 = a_2 x + b_2 \), we then find the best-fit position of the point of interest \( k \) where \( S(k) \), the sum of residuals, is minimal (Figure 4(c)).

\[
S(k)^2 = \sum_{i=1}^{k} (y_1 - (a_1 x_i + b_1))^2 + \sum_{j=k+1}^{n} (y_2 - (a_2 x_j + b_2))^2
\]  

(3)

Next, we find the intersection of the two best fitting lines and consider the difference between the intersection and the corresponding silhouette point as the fitting error \( \varepsilon \) as follows, where \( \delta \) is the pixel size of the projection image:

\[
\varepsilon = \frac{\delta}{h} \left( \frac{b_2 - b_1}{a_2 - a_1} - \frac{q}{2} \right). 
\]  

(4)

Figure 4. (a) Silhouette lines and projection image in the same coordinate. (b) Sampling points close to a silhouette point in the normal vector direction. (c) Computation of the fitting error.
2.3 Computation of the polygonal mesh error

In this step, we apply back-projection of the fitting error to the polygonal mesh to obtain the positional error at each vertex. For each projection image, the back-projection error of each silhouette point \( \varepsilon \) can be computed as follows, where \( \varepsilon_{det} \) is the fitting error on the detector, \( L \) is the distance between the X-ray and the object, \( D \) is the distance between the X-ray and the detector, \( \lambda \) is the angle between the focal axis and the view vector, \( ||c - p|| \) is the distance between the X-ray and the silhouette point (Figure 4):

\[
\varepsilon = \varepsilon_{det} \frac{L}{D} \cos \lambda = \varepsilon_{det} \frac{L^2}{D ||c - p||}.
\]  

(5)

We assign the back-projection error to the vertex pair consisting in the mesh edge that contains a silhouette point. The polygonal mesh is sequentially rotated and compared with the equivalent projection image to compute the corresponding errors of the mesh vertices. We repeat the abovementioned steps until all projection images have been compared. The overall estimated error of each mesh vertex \( E(p) \) is a mean value of its total error and its number of repetition \( M(p) \) during the error-mapping process:

\[
E(p) = \frac{1}{M(p)} \sum_{i=1}^{k} \varepsilon_i(p).
\]  

(6)

Figure 5. Back-projection of the error. Figure 6. Computation of a smoothing vector.

2.4 Shape correction

We correct the mesh shape by moving its vertices along the direction of the normal vectors in the amount of the overall estimated error of each mesh vertex.

\[
p \leftarrow p + E(p)n(p).
\]  

(7)

where \( E(p) \) and \( n(p) \) are the overall estimated error and the normal vector of vertex \( p \), respectively. However, the movement of the vertices in only the normal direction may result in triangles with flipped orientation. To solve this problem, we make the vertex distribution uniform while simultaneously moving the vertices. Because vertex movement on the tangent plane has no affect on the geometry of the mesh surface, we can smooth the surface by moving the vertices in the median
direction of the normal and the tangent vectors [5]. The computation of the smoothing vector is shown in Figure 6, where $L$ is a Laplacian vector at the mesh vertex. As a result, we correct the mesh shape by moving its vertices as follows, where $E(p)n(p)$ is the movement in the normal direction and $L(p) - (L(p) \cdot n)n$ is the movement in the tangential direction:

$$p \leftarrow p + E(p)n(p) + L(p) - (L(p) \cdot n)n.$$

(8)

2.5 Summary of the Algorithm

The proposed algorithm can be summarized as follows:

1. Extract the silhouette lines of the polygonal mesh considering the focal point of the X-ray source as a viewpoint. Then, perform the perspective projection of the silhouette line on the detector.
2. Compute the fitting error by comparing the silhouette line on the detector with the respective projection image having the same projection angle.
3. Apply back-projection of the fitting error to the polygonal mesh to obtain the estimated error at each vertex.
4. Correct the mesh shape by moving mesh vertices in the median direction of normal vectors in the amount of the estimated errors and the tangent vectors in the amount of the Laplacians.

Furthermore, we apply the Block Iterative Method to accelerate shape correction of the surface mesh [6]. In particular, we divide the sinogram $S$ into independent subsets of the sinogram $S_1, S_2, \ldots, S_N$ that satisfy following condition (Figure 7):

$$S = \bigcup_{n=1}^{N} S_n, S_l \cap S_{l'} = \{\emptyset\} \text{ for } l \neq l'.$$

(9)

For each subset, we correct the mesh shape with the above four steps by using respective projection images. The mesh shape, which was corrected by using the current subset, is then applied as an input mesh for the next subset. We iteratively correct the mesh shape until all subsets are applied (Figure 8).

3 Result and discussion

3.1 Experiment conditions

The proposed method was applied to both simulation data and actual scan data. The simulation data was obtained by using the Scorpion XLab software for cone-beam X-ray CT scanner simulation. The
actual scanned data was obtained by using the Carl Zeiss Metrotom 800 cone-beam X-ray CT scanner. The experiment conditions are shown in Table 1. The sinogram of the Sphere is the simulation data and the remains are actual scanned data. Moreover, the number of adjacent pixels of each silhouette point is $q = 4$, the sampling rate of each pixel is $h = 4$, and the number of ordered subsets of the sinogram is $N = 3$ in all cases.

Table 1. Experiment conditions

<table>
<thead>
<tr>
<th></th>
<th>Sphere (Fe)</th>
<th>Rod (Ti)</th>
<th>Step Cylinder (Al)</th>
<th>Block (Al)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage (kV)</td>
<td>120</td>
<td>100</td>
<td>125</td>
<td>80</td>
</tr>
<tr>
<td>Current (mA)</td>
<td>0.5</td>
<td>0.15</td>
<td>0.15</td>
<td>0.1</td>
</tr>
<tr>
<td>Number of pixel</td>
<td>500×500</td>
<td>728×920</td>
<td>728×920</td>
<td>728×920</td>
</tr>
<tr>
<td>Pixel size (mm)</td>
<td>0.03</td>
<td>0.12</td>
<td>0.11</td>
<td>0.04</td>
</tr>
<tr>
<td>Projection numbers</td>
<td>500</td>
<td>800</td>
<td>800</td>
<td>700</td>
</tr>
</tbody>
</table>

3.2 Results

Figure 6 shows the result of the sphere. Due to the cone-beam artifact, the reconstructed CT volume contains partial volume artifacts locating in the vicinity of the poles (Figure 6(a)). We evaluated the error by comparing both initial and corrected shape with the perfect sphere (Figure 6(b)). Shape improvement was clearly visible. The radius of the corrected sphere was 4.995 mm, whereas that of the initial sphere was 4.96 mm. Moreover, sphericity of the corrected and initial spheres were 0.1 and 0.055, respectively. Note that mesh vertices locating in the vicinity of the poles were not used in computing the radii and the sphericities.

Figure 7 shows the results of actual scanned data for a titanium rod with a 2.5 mm radius. Due to the beam hardening artifact of the CT image (Figure 7(b)), the initial shape showed a conspicuous dent, which was not present in the corrected shape (Figure 7(c)). In particular, the radius of the defect part was 2.391 mm in the initial shape, and 2.497 mm in the corrected shape. However, the regular part of the corrected rod unexpectedly increased from 2.512 mm to 2.532 mm over that in the actual scanned data. This problem is a limitation of our proposed method.

Figure 8 shows the results of the actual scanned data for a step cylinder constructed of aluminum with an internal radius of 5.0 mm. Due to the variety in thickness of each step, the lower internal radius of the initial reconstructed shape was larger than the upper internal radius. We practically measured the radius of each step. The results of the upper, middle, and lower internal radii were 4.971 mm, 4.978 mm, and 4.998 mm, respectively. Conversely, the measurement results for the corresponding parts of
the corrected shape were 4.998 mm, 5.0 mm, and 5.006 mm. Therefore, the dimensions of the data were improved. However, the proposed method was ineffective in the aluminum block experiment (Figure 9). The corrected surface became slightly rough, and the dimensions of the block did not change significantly compared to the initial data. A possible reason is that it was difficult to detect the silhouette lines of the planar parts. As the result, such parts did not have sufficient information to be accurately corrected.

Figure 7. Results of rod experiment (radius = 2.5 mm). (a) Actual titanium rod. (b) Cross-section of CT volume. (c) Comparison of initial and corrected shape.

Figure 8. Results of step cylinder. (a) Aluminum step cylinder. (b) Cross-section of CT volume. (c) Error colormap of initial and corrected shape.

Figure 9. Results of block experiment. (a) Aluminum block. (b) Cross-section of CT volume. (c) Original data. (d) Correction data.
3.3 Discussion
The computation times are summarized in Table 2. Intel® Core™ i7-3520M CPU @ 2.90GHz Processor and 12.0 GB of RAM are used for the computation.

<table>
<thead>
<tr>
<th></th>
<th>Number of vertices</th>
<th>Number of triangles</th>
<th>Computation time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sphere</td>
<td>52,384</td>
<td>104,763</td>
<td>185 sec.</td>
</tr>
<tr>
<td>Rod</td>
<td>30,396</td>
<td>60,788</td>
<td>63 sec.</td>
</tr>
<tr>
<td>Cylinder</td>
<td>347,954</td>
<td>695,904</td>
<td>2243 sec.</td>
</tr>
<tr>
<td>Block</td>
<td>373,214</td>
<td>756,424</td>
<td>2627 sec.</td>
</tr>
</tbody>
</table>

The proposed method is applicable to objects in which the silhouette lines can be easily detected. In particular, objects having curves and pin angles, such as spheres and cylinders, can be accurately corrected because the silhouette lines of these parts were easily detected. Moreover, sub-pixel positional accuracy of the mesh vertices can be achieved. However, this method has two limitations: It is not applicable to parts in which the silhouette lines cannot be accurately detected. In addition, parts in which the initial reconstructed shapes show significant defects also cannot be accurately corrected.

4 Conclusion
This paper proposed a novel method for correcting the shape inaccuracy caused by CT artifacts by comparing shape silhouette lines and X-ray projection images. We demonstrated that the shape accuracies of spheres and cylinders are improved by applying the proposed method. In future work, we plan to improve the method so that it can be applied to a wide range of shapes and objects composed of multiple materials.

References