PotSAC: A Robust Axis Estimator for Axially Symmetric Pot Fragments

Je Hyeong Hong
KIST
jhh37@kist.re.kr

Young Min Kim
Seoul National University
youngmin.kim@snu.ac.kr

Koang-Chul Wi
Hanseo University
kcwi@hanseo.ac.kr

Jinwook Kim
KIST
zinook@kist.re.kr

Abstract

The task of virtually reassembling an axially symmetric pot from its fragments can be greatly simplified by utilizing the constraints induced by the pot's axis of symmetry. This requires accurate estimation of the axis for each sherd, whose 3D data typically contain gross outliers arising from surface artifacts, noisy surface normals and unfiltered data along the break surface. In this work, we propose a simple two-stage robust axis estimator, PotSAC, which is based on a variant of the random sample consensus (RANSAC) algorithm followed by robust nonlinear least squares refinement. Unlike previous work which have either compensated the axis estimation accuracy for robustness against outliers or vice versa, our method can handle the aforementioned outlier sources without compromising its accuracy. This is achieved by carefully designing the method to combine and extend the advantage of each key prior work. Experimental results on real scanned fragments demonstrate the effectiveness of our method, paving the way towards high quality reassembly of symmetric potteries.

1. Introduction

Axially symmetric pots are one of the most dominant sources of cultural heritage excavated in East Asia. However, they are found as a large volume of scattered broken pieces, which are gathered, cleansed and reassembled by restoration professionals. The manual process often handles tens of pieces with a considerable amount of time, and sometimes incur undesired damages on the fragments. Furthermore, the current practice of the manual method cannot scale to hundreds or thousands of pieces, as the number of possible configurations significantly increases. This has motivated researchers towards developing an automatic virtual reassembly algorithm given 3D scanned pot fragments [19, 15].

Given two pieces of fragments from the pottery, a naive reassembly tries full rigid transformation between them, which can be non-trivial without good pose initialization [2]. Fortunately, due to the innate rotational symmetry of these potteries, degrees of freedom can be significantly reduced at most to 2 (compared to 6) with a sign ambiguity along the axis. Existing virtual re-assembly works [19, 15] have attempted to take advantage of this symmetry by proposing their own axis estimation methods and simplifying the merging process.

Figure 1. Accurately estimated axes of symmetry can simplify the virtual reassembly of symmetric pots. The image above left shows a manually reassembled pottery, while the image above right shows the top view of a possible configuration after aligning individual axes of symmetry obtained using PotSAC. Our method provides highly accurate initialization for virtual reassembly of real symmetric-pot fragments.

In this paper, we propose PotSAC, a two-stage robust axis estimation method that takes in a noisy point cloud and surface normals of a pot fragment and outputs a feasible axis of symmetry in most cases. Our algorithm is largely inspired by the success of the random sample consensus (RANSAC) framework [6, 16] and robust nonlinear least squares optimization [17, 1] in computer vision. In devising our algorithm, we have carefully reviewed prior work (see Sec. 2) to selectively gather the advantage of each method's problem formulation. It will be shown in Sec. 5 that PotSAC can retrieve precise and visually convincing axes for a range of different sherds.

While our method is not the first work to incorporate RANSAC for axis estimation, previous work [15] have only applied it to a circle fitting-based approach, which discards potentially useful surface normal data and achieves lower accuracy. We believe this is the first work presenting a

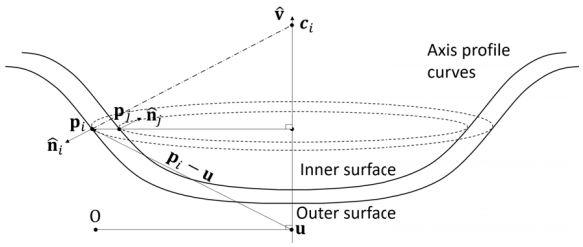


Figure 2. An illustration of the underlying geometry of a symmetric pot. $\hat{\mathbf{v}}$ and $\mathbf{u}$ represent the direction and offset of the symmetric axis respectively, $\mathbf{p}_i$ is the location of a surface point i and $\hat{\mathbf{n}}_i$ is the surface normal at $\mathbf{p}_i$. $\mathbf{u}$ is the axis offset perpendicular to $\hat{\mathbf{v}}$ and $\mathbf{c}_i$ is the point along the axial line toward $\mathbf{p}_i$. Each symbol with a hat sign is normalized.

RANSAC-initiated algorithm that utilizes surface normal data, which can provide strong geometric constraints as will be reviewed in Sec. 2.

Our development of PotSAC has led to the following specific contributions:

+ Extension of Cao and Mumford [3]'s geometric constraint (see Sec. 2.2) to consider both inner and outer surfaces while maintaining differentiability. This extended residual is utilized for MLESAC [16] and nonlinear refinement.
+ Development of an efficient 6-point minimal axis estimator based on Pottmann et al.'s coplanarity constraint (see Sec. 2.1) for use with a RANSAC-based framework.
+ Trust region-based robust nonlinear axis refinement incorporating the manifold structure of the axial line.
+ Experimental comparisons against other axis estimation algorithms on synthetic and real datasets.

Conversely, there are limitations to this work. We do not utilize information associated with the axis profile curve (defined as the curve along the revolutional rz-plane), which can provide additional useful geometric constraints but requires further assumptions on the curve shape. We leave this task of integrating constraints from the axis profile curve into the RANSAC framework for future work.

The structure of this paper is as follows: we review relevant literature in Sec. 2, and propose a modification to a previously used cost function in Sec. 3, which allows consideration of both inner and outer fragment surfaces. We illustrate our proposed method in Sec. 4. Experimental results and discussions are provided in Sec. 5 followed by conclusions in Sec. 6.

2. Review of known approaches

We start by reviewing known methods for axis estimation which form the basis of this work (refer to Fig. 2 for symbols).

2.1. Pottmann et al.'s method

Two decades ago, Pottmann et al. [14, 13] presented a pioneering work on axis estimation of symmetric-pot fragments. They proposed an algebraic constraint (termed *moment*) for a more general type of surfaces defined as the *linear complex*, which includes surfaces generated through rotational, helical and translational motions. For the rotational (symmetric-pot) case, Cao and Mumford [3] showed this can be viewed as a coplanarity constraint of three vectors, namely the position vector of a surface point with respect to a predefined point along the axis (e.g. $\mathbf{p}_i - \mathbf{u}$), its corresponding surface normal $\hat{\mathbf{n}}_i$ and the axis direction $\hat{\mathbf{v}}$. i.e.

$$(\hat{\mathbf{v}} \times \hat{\mathbf{n}}_i) \cdot (\mathbf{p}_i - \mathbf{u}) = 0 \quad \forall \quad i \in \Omega. \tag{1}$$

Using $(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{c} = (\mathbf{b} \times \mathbf{c}) \cdot \mathbf{a} = (\mathbf{c} \times \mathbf{a}) \cdot \mathbf{b}$ yields

$$(\hat{\mathbf{n}}_i \times \mathbf{p}_i) \cdot \hat{\mathbf{v}} - (\mathbf{u} \times \hat{\mathbf{v}}) \cdot \hat{\mathbf{n}}_i = 0 \quad \forall \quad i \in \Omega. \tag{2}$$

(2) is nonlinear due to the inherent bilinearity between $\hat{\mathbf{v}}$ and $\hat{\mathbf{u}}$ and the underlying spherical manifold of $\hat{\mathbf{v}}$.

Pottmann et al. [14] presented some tricks for reformulating (2) as simpler quadratic constraints. First, since $\mathbf{u} \perp \hat{\mathbf{v}}$ from Fig. 2, one can introduce a new variable $\mathbf{w} := \hat{\mathbf{v}} \times \mathbf{u}$ to make (2) linear in $\mathbf{w}$. (note $\mathbf{u} = \hat{\mathbf{v}} \times \mathbf{w}$). Second, $\hat{\mathbf{v}}$ can be replaced with its unnormalized counterpart $\mathbf{v}$ by adding a unit norm constraint on $\mathbf{v}$. Combining these transforms (2) to

$$(\hat{\mathbf{n}}_i \times \mathbf{p}_i)^\top \mathbf{v} + \hat{\mathbf{n}}_i^\top \mathbf{w} = 0 \quad \forall \quad i \in \Omega \quad s.t. \quad \|\mathbf{v}\|_2^2 = 1. \tag{3}$$

Last, by defining $\mathbf{z} := [\mathbf{v}\ ;\ \mathbf{w}]$, (3) can be rewritten as

$$\mathbf{z}^\top\ \mathtt{J}^\top \mathtt{J}\ \mathbf{z} =: \mathbf{z}^\top\ \mathtt{M}\ \mathbf{z} = 0 \quad s.t. \quad \mathbf{z}^\top \mathtt{D} \mathbf{z} = 1, \tag{4}$$

where $\mathtt{J} \in \mathbb{R}^{|\Omega| \times 6}$ is defined as a row stack of $\mathtt{J}_i := [(\hat{\mathbf{n}}_i \times \mathbf{p}_i)^\top \ \ \hat{\mathbf{n}}_i^\top]\ \forall\ i \in \Omega$, $\mathtt{M} := \mathtt{J}^\top \mathtt{J}$, and $\mathtt{D} \in \mathbb{R}^{3\times3}$ is a diagonal matrix comprising entries $[1, 1, 1, 0, 0, 0]$.

In practice, measurement noise invalidates the equality in (4), and we therefore minimize $\mathbf{z}^\top\ \mathtt{M}\ \mathbf{z}$ over $\mathbf{z}$ subject to $\mathbf{z}^\top \mathtt{D} \mathbf{z} = 1$, which is essentially a generalized eigenvalue problem $\det(\mathtt{M} - \lambda \mathtt{D}) = 0$ that can be solved in closed form [14].

The main advantage of this algorithm is its insensitiveness to initialization. However, it can output a biased model due to the algebraic nature of the minimized objective, which lacks geometric consideration of the measurement noise.

2.2. Cao and Mumford's method

A few years later, Cao and Mumford [3] argued (1) on its own is not strong enough to prevent degenerate solutions, and one should incorporate the length constraint on

the radius of curvature for each surface point. From Fig. 2, we can see that the radius of the horizontal circle crossing point i at its boundary, r_i is $\|(\mathbf{p}_i - \mathbf{u}) \times \hat{\mathbf{v}}\|_2$. This means the length of the line joining $\mathbf{p}_i$ and its center of (spherical) curvature is $r_i / \|\hat{\mathbf{v}} \times \hat{\mathbf{n}}_i\|_2$, and therefore the center of the curvature, $\mathbf{c}_i$, should lie at

$$\mathbf{c}_i(\mathbf{u}, \hat{\mathbf{v}}) := \mathbf{p}_i - \frac{\|(\mathbf{p}_i - \mathbf{u}) \times \hat{\mathbf{v}}\|_2}{\|\hat{\mathbf{v}} \times \hat{\mathbf{n}}_i\|_2} \hat{\mathbf{n}}_i \quad \forall \quad i \in \Omega. \tag{5}$$

Ideally, $\mathbf{c}_i - \mathbf{u}$ should be parallel to the symmetric axis, i.e., $(\mathbf{c}_i - \mathbf{u}) \times \hat{\mathbf{v}} = 0, \ \forall \ i \in \Omega$. We define Cao and Mumford's geometric residual, $\boldsymbol{\varepsilon}_i \in \mathbb{R}^3$, as

$$\boldsymbol{\varepsilon}_i(\mathbf{u}, \hat{\mathbf{v}}) := (\mathbf{c}_i(\mathbf{u}, \hat{\mathbf{v}}) - \mathbf{u}) \times \hat{\mathbf{v}}, \tag{6}$$

which can be viewed as the deviation of the center of curvature from the symmetric axis on the (horizontal) xy-plane (i.e., the left nullspace of $\hat{\mathbf{v}}$). In practice, one may minimize

$$\sum_{i \in \Omega} \rho \left(\|\boldsymbol{\varepsilon}_i(\mathbf{u}, \hat{\mathbf{v}})\|_2^2 \right), \tag{7}$$

where $\rho : \mathbb{R} \to \mathbb{R}$ denotes a kernel for discarding outliers. Cao and Mumford assigned a binary weight on each residual to discard these outliers [3]. Their method can be viewed as a primitive variant of iteratively reweighted least squares (IRLS, see [20] for details) using a non-continuous kernel ($\rho(s) = s$ if within the inlier threshold, otherwise 0), adaptive kernel width and inner iterations [1] on the model variables.

Since (7) is nonlinear, it requires good initialization. Cao and Mumford obtained this through Pottmann et al.'s algorithm, but we will show empirically in Sec. 5 that this initialization can fail frequently on real dataset.

2.3. Other methods

Aside from the above methods which lay foundation to PotSAC, there are two other types of known approaches for estimating the symmetric axis which we will briefly review here.

Method based on the axis profile curve: Willis et al. [18] noted that the surface of an ideal symmetric pottery can be described by an axis profile curve (APC) on the rz-plane revolved around the axis of symmetry by 2π. They defined an objective function, which combines the point-to-surface distance the normal-to-normal distance on the rz-plane for each point, and minimized this over the axis parameters and the APC coefficients in an alternating fashion. Similar to Cao and Mumford's error, this approach relies on an accurate-enough initialization (e.g., Pottmann's method from Sec. 2.1) for it to converge to a good local minimum. However, its objective does not consider tangential deviations of surface normals.

Methods based on circle fitting: Mara and Sablatnig [9] proposed a method whereby, for a given direction of the symmetric axis, a circle template is fit onto each disc, and the axis is estimated by regressing through the estimated circle centers. This procedure is repeated for each axis direction drawn uniformly from a unit hemisphere. Son et al. [15] robustified this scheme by incorporating RANSAC for both the circle fitting and the final axis regression steps. The main drawback of these methods is that it discards potentially useful surface normal data. We will show in Sec. 5 that this leads to inferior precision in axis estimation. Additionally, regressing through the circle centers will generate an axis different from the sampled (tested) axis, requiring an iterative approach alternating between circle fitting and regression of circle centers.

3. Extension of Cao and Mumford's error

We now propose a modification to the Cao and Mumford's geometric error that allows utilization of both the inner and outer surfaces of symmetric-pot fragments.

The original error in (6) is sensitive to the sign of the surface normal direction; as shown in Fig. 2, it only holds for the outer surface in its current form. Since we have data containing both the outer and inner surfaces, we need an objective that can consider both modes of the normal.

One way of achieving the above is by computing (6) twice for each surface point, first with the original normal direction and second with its reverse, followed by taking the minimum of the two. If we call the original (forward) residual $\boldsymbol{\varepsilon}_i^+$ and define the residual with the reversed normal $\boldsymbol{\varepsilon}_i^-$, then we obtain

$$\min \left(\|\boldsymbol{\varepsilon}_i^+\|_2^2, \|\boldsymbol{\varepsilon}_i^-\|_2^2 \right). \tag{8}$$

Since (8) is indifferentiable, we apply the well-known *log-sumexp* trick [4, 11] to yield a smooth approximation of (8)

$$f_i := -\frac{1}{t} \ln \left(\sum_s \exp \left(-t \ \|\boldsymbol{\varepsilon}_i^s\|_2^2 \right) \right), \tag{9}$$

where $t \in \mathbb{R}^+$ is a smoothness parameter and $s \in \{+, -\}$ is a binary sign indicator for the normal direction. One can immediately see that $t \to \infty$ yields (8). Through empirical studies, we find $t = 100$ is a suitable choice for our application, resembling (8) while being numerically stable with differentiation when measurements are in the order of millimeters (mm).

4. Proposed method

PotSAC essentially minimizes a robustified version of our extended Cao and Mumford cost from Sec. 3. i.e., we solve

$$\arg\min_{\mathbf{u}, \hat{\mathbf{v}}} \sum_{i \in \Omega} \rho \left(f_i^2(\mathbf{u}, \hat{\mathbf{v}}) \right), \tag{10}$$

Algorithm 1 MLESAC-based axis estimation

input: a 3D point cloud with surface normals
for $k = 1 \cdots \texttt{max\_iter}$ **do**
    Sample and run 6-point algorithm to obtain the axis.
    Compute total MLESAC(truncated quadratic) cost
    using our extended Cao and Mumford error metric.
end for
Select the axis with the lowest cost.
output: a 3D line representing the axis of symmetry

where f_i is the extended Cao and Mumford cost, $\mathbf{u}$ and $\hat{\mathbf{v}}$ represent the axis translation and direction respectively and $\rho : \mathbb{R} \rightarrow \mathbb{R}$ is a robust kernel for removing the effect of gross outliers. Since (10) is highly nonlinear, it requires good initialization to converge to a high quality local optimum. PotSAC runs as a two-stage algorithm:

1. Run parallel MLESAC with our 6-point solver to obtain an axis based on our extension extended Cao and Mumford cost criterion.

2. Refine the axis using a trust-region based nonlinear least squares optimizer incorporating robustness.

4.1. MLESAC with parallel sampling

Standard RANSAC [6] works by repeatedly performing fitting a model to a set of random data samples and testing the model to other unseen data until a model with good consensus is found. A threshold parameter $\tau \in \mathbb{R}$ determines which data points are inliers, and a model with the largest number of inliers is selected. Since our objective is the modified Cao and Mumford's geometric error from Sec. 3, we set τ to 1.0 mm^2.

To implement RANSAC efficiently, one requires an efficient minimal solver that can estimate a model over thousands of iterations. Also, using fewer data points in each trial improves the probability of sampling only inlier data, reducing the number of trials required. For this purpose, we have devised an algorithm (see below) that can estimate an axis from 6 surface point-normal pairs.

We employ MLESAC [16], which is a variant of RANSAC that also considers the quality of inlier data points based on their deviations from the model. While RANSAC assigns zero costs to all inliers, MLESAC computes the L_2-norm cost for each inlier, saturating to 1 as the error distance reaches τ. Since both models assign unit cost to each outlier, MLESAC can be viewed as incorporating a truncated quadratic kernel in solving (10) instead of RANSAC's well-shaped kernel. Our pseudocode can be found in Algorithm 1. We perform RANSAC iterations in parallel to reduce runtime on multicore machines.

6-point algorithm: Our 6-point algorithm is essentially a minimal numerically-stable implementation of the

Algorithm 2 Nonlinear axis refinement

inputs: 3D point cloud with normals and symmetric axis
for $k = 1 \cdots \texttt{max\_iter}$ **do**
    Project the Jacobian to the tangent space of the axis.
    Augment the subproblem to penalize movement along the axis.
    Compute an update given current trust region radius
    Compute the quality of the second order model
    Accept the step if
    Raise/lower the trust region radius given the model quality
end for
Select the axis with the lowest cost.
output: a 3D line representing the axis of symmetry

Pottmann et al.'s method [14]. Continuing from Sec. 2.1, we first decompose the Jacobian into $\mathtt{J} =: [\mathtt{J}_\mathbf{v}\ \mathtt{J}_\mathbf{w}]$, where $\mathtt{J}_\mathbf{v} \in \mathbb{R}^{|\Omega|\times 3}$ and $\mathtt{J}_\mathbf{w} \in \mathbb{R}^{|\Omega|\times 3}$ represent the Jacobians with respect to $\mathbf{v}$ and $\mathbf{w}$ respectively. Since $\mathtt{M} = \mathtt{J}^\top\mathtt{J}$, we can write $\mathtt{M} - \lambda\mathtt{D}$ as

$$\begin{bmatrix} \mathtt{J}_\mathbf{v}^\top\mathtt{J}_\mathbf{v} - \lambda\mathtt{I} & \mathtt{J}_\mathbf{v}^\top\mathtt{J}_\mathbf{w} \\ \mathtt{J}_\mathbf{w}^\top\mathtt{J}_\mathbf{v} & \mathtt{J}_\mathbf{w}^\top\mathtt{J}_\mathbf{w} \end{bmatrix}. \tag{11}$$

Hence, by applying the Schur complement trick [17], we note $\det(\mathtt{M} - \lambda\mathtt{D}) = 0$ leads to $\det(\mathtt{J}_\mathbf{w}^\top\mathtt{J}_\mathbf{w}) = 0$ and/or

$$\det(\mathtt{J}_\mathbf{v}^\top\mathtt{J}_\mathbf{v} - \mathtt{J}_\mathbf{v}^\top\mathtt{J}_\mathbf{w}\mathtt{J}_\mathbf{w}^\dagger\mathtt{J}_\mathbf{v} - \lambda\mathtt{I}) = 0, \tag{12}$$

where $\mathtt{J}_\mathbf{w}^\dagger := (\mathtt{J}_\mathbf{w}^\top\mathtt{J}_\mathbf{w})^{-1}\mathtt{J}_\mathbf{w}^\top$ is the left pseudoinverse of $\mathtt{J}_\mathbf{w}$. Since the first equality only holds for a degenerate case, the smallest eigenvector of $\mathtt{J}_\mathbf{v}^\top\mathtt{J}_\mathbf{v} - \mathtt{J}_\mathbf{v}^\top\mathtt{J}_\mathbf{w}\mathtt{J}_\mathbf{w}^\dagger\mathtt{J}_\mathbf{v}$ satisfying (12) is the solution for minimizing the Pottmann's coplanarity error.

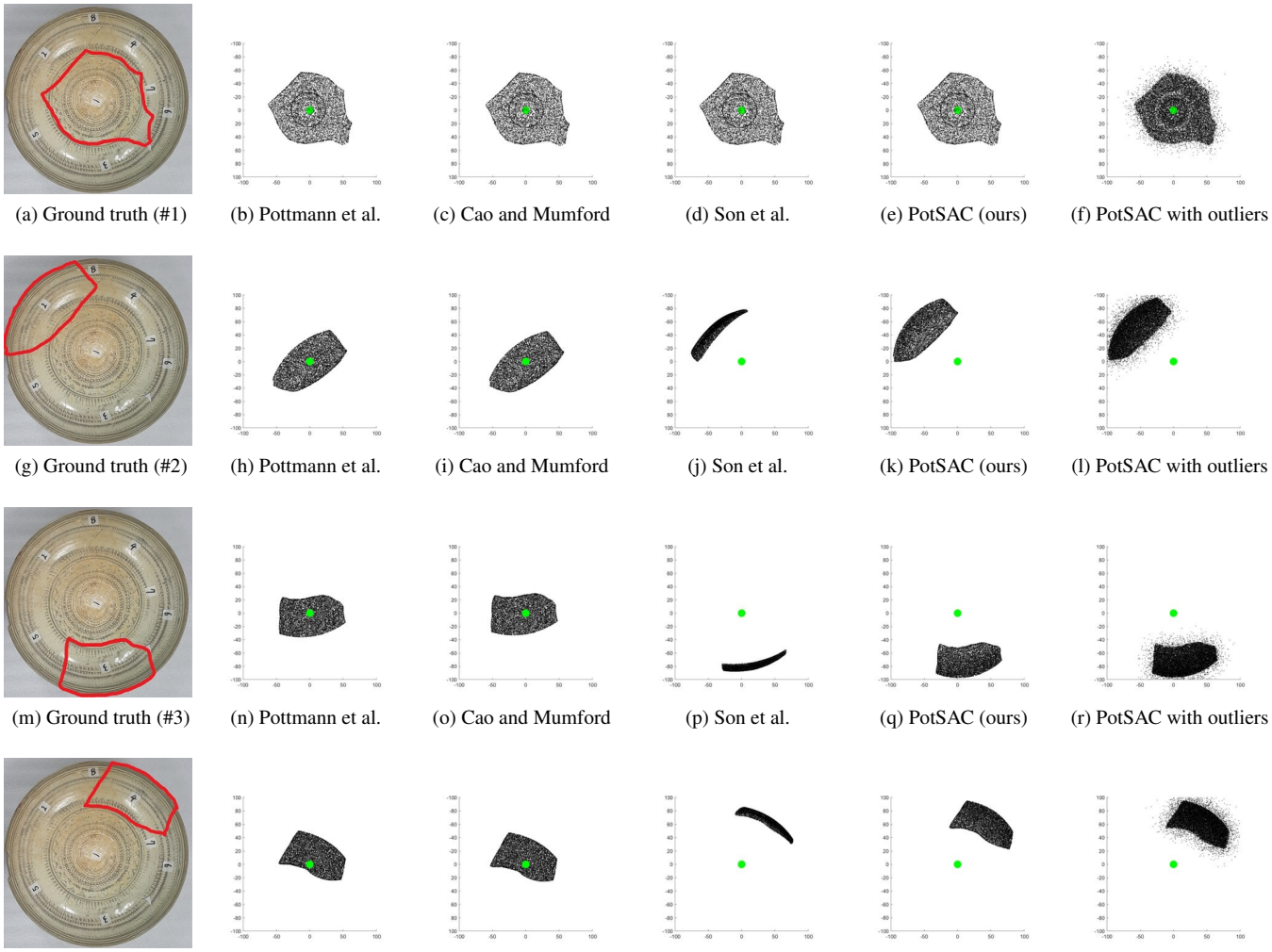
In terms of implementation, $\mathtt{J}_\mathbf{w}\mathtt{J}_\mathbf{w}^\dagger$ can be efficiently computed by (economic) QR-decomposing $\mathtt{J}_\mathbf{w}$ into $\mathtt{Q}_\mathbf{w}\mathtt{R}_\mathbf{w}$, and then computing $\mathtt{Q}_\mathbf{w}\mathtt{Q}_\mathbf{w}^\top$. Since the reduced system matrix is symmetric and positive semidefinite, we can apply the numerically-stable singular value decomposition (SVD) and compute the basis with the smallest singular value.

Since a 3D line can be represented by 4 parameters [14, 3], this algorithm is not the most compact minimal solver for symmetric axis estimation. We leave the development of an efficient 4-point algorithm for future work.

4.2. Robust nonlinear refinement

Once MLESAC yields a solution, it is refined to consider all data to alleviate the model bias towards the original 6 points.

Solving (10) requires a robust nonlinear least squares algorithm that can consider the manifold of the variables. First, we employ the Levenberg-Marquardt (LM) [8, 10] algorithm as the base method, which can be viewed as a 2nd



(a) Ground truth (#1) (b) Pottmann et al. (c) Cao and Mumford (d) Son et al. (e) PotSAC (ours) (f) PotSAC with outliers

(g) Ground truth (#2) (h) Pottmann et al. (i) Cao and Mumford (j) Son et al. (k) PotSAC (ours) (l) PotSAC with outliers

(m) Ground truth (#3) (n) Pottmann et al. (o) Cao and Mumford (p) Son et al. (q) PotSAC (ours) (r) PotSAC with outliers

(s) Ground truth (#4) (t) Pottmann et al. (u) Cao and Mumford (v) Son et al. (w) PotSAC (ours) (x) PotSAC with outliers

Figure 3. Visualization of the axis returned by each algorithm when using all data points. For the random sampling-based methods (Son et al. [15] and PotSAC), we output the mode axis across multiple runs of 100.

order trust-region algorithm. LM can be viewed as a 2nd-order trust-region approach, which in each iteration makes a local quadratic approximation of the objective and computes an update based on this and the size of the trust region (See [1]).

Second, we apply the Trigg's correction [17, 1] to LM, which integrates robustness by dynamically weighting the residual and Jacobian depending on the choice of the robust kernel and the error distance. We use the Cauchy kernel [1]. Implementing a more advanced robust optimization technique such as lifting [20] is for future work.

Last, we employ a manifold optimization framework [12, 7] considering the 3D line structure. We project the Jacobian to the tangent space of the current axis direction such that no update is performed along the axis. Since this results in a rank-deficient quasi-Hessian matrix, we add penalty terms to the local quadratic model to discourage the axis translation and direction from moving along the current axis direction (since both update directions leave the cost unchanged).

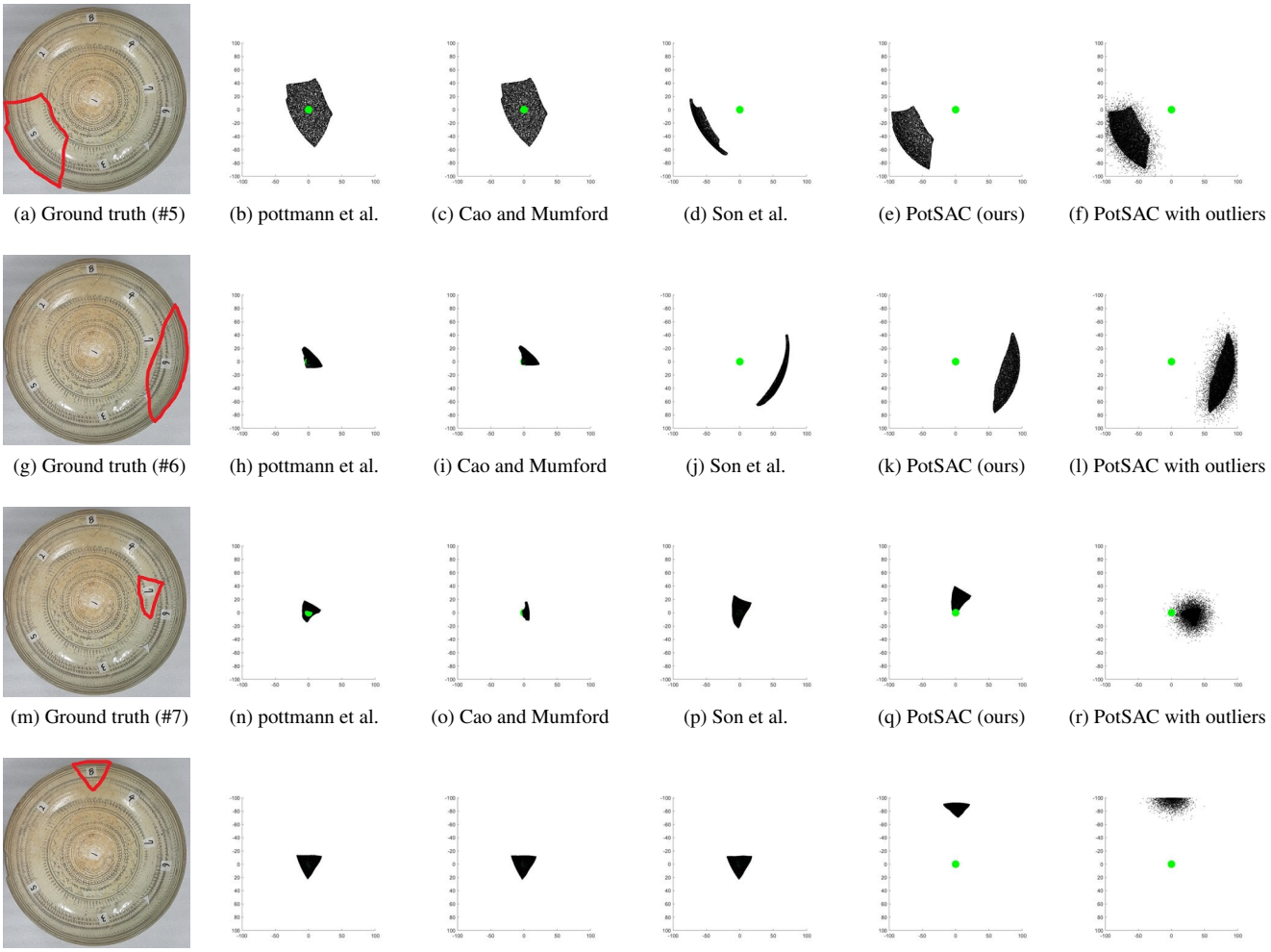
5. Experimental results

We carried out experiments on an 8-core AMD Ryzen 2700X machine with 32GB DDR4 RAM using MATLAB R2018b with Parallel Computing Toolbox. We have attached a short supplementary video showing a demo run of our code.

PotSAC was compared against our re-implementations of several reference and state-of-the-art methods, including the work of Pottmann et al. [14], Cao and Mumford [3] and Son et al. [15]. Son et al. estimates the axis by slicing the point cloud data into many thin layers and fitting a circle template onto each slice. We modified Son et al.'s algorithm to efficiently draw normal directions [5] and fit a hollow disc rather than a circle since we deal with both inner and

Method	Frag. #1	Frag. #2	Frag. #3	Frag. #4	Frag. #5	Frag. #6	Frag. #7	Frag. #8	$\bar{t}$ (s)
Pottmann et al.	0.5°(3.9)	4.4° (49)	3.0°(11)	3.0°(29)	3.6°(41)	14.0°(78)	17.8°(69)	**1.7°(13)**	**0.01**
Cao & Mumford	0.4°(2.6)	5.8°(52)	4.3°(12)	4.9°(34)	4.9°(52)	15.9°(50)	30.2°(95)	**2.1°(13)**	1.18
Son et al.	2.2°(12)	12.1°(71)	8.6°(24)	7.2°(62)	5.1°(61)	6.0°(54)	**9.1°(63)**	9.6°(65)	4.46
MLESAC (ours)	0.4°(3.6)	4.4°(16)	3.2°(14)	2.5°(6.9)	**2.6°(6.3)**	**4.9°(9.4)**	34.3°(22)	30.6°(19)	3.63
PotSAC (ours)	**0.2°(2.4)**	**2.2°(11)**	**1.5°(8.9)**	**1.2°(5.3)**	**1.3°(10)**	**2.2°(10)**	21.7°(54)	21.1°(21)	4.36

Table 1. 2σ-precisions of symmetric axes' directions (°) and positions (mm, in parenthesis) obtained by different algorithms on real pottery fragments. $\bar{t}$ denotes mean runtime on original datasets of 10,000 points. Note *higher precision does not necessarily yield higher accuracy*— on #7 and #8, the pieces are relatively small and have shape ambiguities, making all algorithms fail (See Fig. 4).



(a) Ground truth (#5) (b) pottmann et al. (c) Cao and Mumford (d) Son et al. (e) PotSAC (ours) (f) PotSAC with outliers

(g) Ground truth (#6) (h) pottmann et al. (i) Cao and Mumford (j) Son et al. (k) PotSAC (ours) (l) PotSAC with outliers

(m) Ground truth (#7) (n) pottmann et al. (o) Cao and Mumford (p) Son et al. (q) PotSAC (ours) (r) PotSAC with outliers

(s) Ground truth (#8) (t) pottmann et al. (u) Cao and Mumford (v) Son et al. (w) PotSAC (ours) (x) PotSAC with outliers

Figure 4. Visualizations of the axes returned by each algorithm when using all data points for each of the sherds #5 to #8. For the random sampling-based methods (Son et al. [15] and PotSAC), we show the median axis across multiple runs of 100.

outer surfaces.

All methods were tested on real pot fragments from the 15th century. These contain inner, outer and break surfaces, which can have different surface characteristics such as normal directions. We resampled each piece to contain 10,000 points in 1–2s. To generate data with gross outliers (e.g. Fig. 3b), we replaced 30% of samples by randomly sampled points and normals around the surfaces.

Fig. 3 and Fig. 4 show visualizations of results achieved by each algorithm from top view when run on each full fragment dataset (i.e. using 10,000 points on fragments #1–#8). Green dots at the origin denote axes of symmetry (in cases green dots are missing, the axis center is at $(0, 0)$). Note for each fragment, (a) and (b) tend to find the axis center

at the center of piece, while (c) tends to lift the fragment to form a more cylindrical shape. PotSAC is robust to these issues. All algorithms fail on #7 and #8, motivating future research towards incorporating more constraints jointly within the RANSAC framework.

Since we do not have ground truth axes, we measured each algorithm's precision on each dataset by applying bootstrapping as in prior work [3, 18]. More specifically, each algorithm is run 100 times on each dataset, with each run using randomly drawn 1,000 samples from 10,000 points. The corresponding results are shown in Table 1 with visualizations of mode solutions shown in Fig. 3 and Fig. 4.

Pottmann et al.'s method runs very fast (< 0.01 s) but usually yields visually less feasible solutions potentially due to its algebraic nature and lack of robustness. This provides bad initialization for Cao and Mumford's method, which consequently fails after running for 1–2s despite their algorithm incorporating robustness. This reconfirms good initialization is key to improving the accuracy of axis estimation. Son et al.'s algorithm runs in about 5s for 5000 normal direction samples. It does sometimes provide visually better estimates than the above two methods, but without utilizing normal information, it often biases towards cylinder-shaped models. On the other hand, PotSAC mostly yields close-to-ground-truth axis in 5s, even when 30% of data points are gross outliers. All algorithms either fail to produce visually convincing results or yield low precision on small fragments #7 and #8 with shape ambiguities, motivating future work to explore on RANSAC-compatiable constraints on the axis profile curve.

6. Conclusions

In this work, we have proposed PotSAC, a robust method for estimating the axis of symmetry for symmetric pot fragments. The method appropriately utilizes algebraic and geometric constraints and extends them to efficiently adopt the random sample consensus (RANSAC) framework. PotSAC also employs a robust nonlinear refinement step to minimize a geometrically meaningful error. Experimental results demonstrate improved accuracy of the estimated axes over other methods for different pot fragments.

Future work should focus on jointly incorporating a constraint on the axis profile curve for removing practically less feasible models. A more ambitious goal is to build a large pottery database and develop an end-to-end learning-based framework for fast axis estimation of fragments.

References

[1] S. Agarwal, K. Mierle, and Others. Ceres solver. http://ceres-solver.org. 1, 3, 5

[2] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, Feb 1992. 1

[3] Y. Cao and D. Mumford. Geometric structure estimation of axially symmetric pots from small fragments. In *Proceedings of Signal Processing, Pattern Recognition, and Applications*, 2002. 2, 3, 4, 5, 7

[4] J. D. Cook. How to compute the soft maximum. https://www.johndcook.com/blog/2010/01/20/how-to-compute-the-soft-maximum/. 3

[5] M. Deserno. How to generate equidistributed points on the surface of a sphere. Technical report, Max Planck Institute for Polymer Research, 2004. 5

[6] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981. 1, 4

[7] J. H. Hong, C. Zach, A. W. Fitzgibbon, and R. Cipolla. Projective bundle adjustment from arbitrary initialization using the variable projection method. In *14th European Conference on Computer Vision (ECCV)*, pages 477–493, 2016. 5

[8] K. Levenberg. A method for the solution of certain nonlinear problems in least squares. *Quarterly of Applied Mathmatics*, 2(2):164–168, 1944. 4

[9] H. Mara and R. Sablatnig. Orientation of fragments of rotationally symmetrical 3d-shapes for archaeological documentation. In *Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, 3DPVT '06, pages 1064–1071, Washington, DC, USA, 2006. IEEE Computer Society. 3

[10] D. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963. 4

[11] N. Matsuda, A. Fix, and D. Lanman. Focal surface displays. *ACM Trans. Graph.*, 36(4):86:1–86:14, July 2017. 3

[12] T. Okatani, T. Yoshida, and K. Deguchi. Efficient algorithm for low-rank matrix factorization with missing components and performance comparison of latest algorithms. In *2011 IEEE International Conference on Computer Vision (ICCV)*, pages 842–849, 2011. 5

[13] H. Pottmann, M. Hofer, B. Odehnal, and J. Wallner. Line geometry for 3d shape understanding and reconstruction. In *Computer Vision - ECCV 2004*, pages 297–309, 2004. 2

[14] H. Pottmann, M. Peternell, and B. Ravani. An introduction to line geometry with applications. *Computer-Aided Design*, 31(1):3 – 16, 1999. 2, 4, 5

[15] K. Son, E. B. Almeida, and D. B. Cooper. Axially symmetric 3d pots configuration system using axis of symmetry and break curve. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013. 1, 3, 5, 6

[16] P. Torr and A. Zisserman. Mlesac. *Comput. Vis. Image Underst.*, 78(1):138–156, Apr. 2000. 1, 2, 4

[17] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, ICCV '99, pages 298–372, London, UK, UK, 2000. Springer-Verlag. 1, 4, 5

[18] A. Willis, X. Orriols, and D. B. Cooper. Accurately estimating sherd 3d surface geometry with application to pot recon-

struction. In *Proceedings of the 2003 Conference on Computer Vision and Pattern Recognition Workshop*, volume 1, pages 5–5, June 2003. 3, 7

[19] A. R. Willis and D. B. Cooper. Bayesian assembly of 3d axially symmetric shapes from fragments. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–I, June 2004. 1

[20] C. Zach. Robust bundle adjustment revisited. In *13th European Conference on Computer Vision (ECCV)*, pages 772–787, 2014. 3, 5