Assembling Virtual Pots from 3D Measurements of their Fragments

The SHAPE Lab - STITCH project*

Brown University, Division of Engineering, Providence, RI 02912, USA
mailto:cooper@lems.brown.edu

VAST 2001 - Workshop Paper # *55*

Abstract

A heretofore unsolved problem of great archaeological importance is the automatic assembly of pots made on a wheel from the hundreds (or thousands) of sherds found at an excavation site. An approach is presented to the automatic estimation of mathematical models of such pots from 3D measurements of sherds. The overall approach is formulated and described and some detail is provided on the elements of the procedure. The end result is a representation suitable for comparisons, geometric feature extraction, visualization and digital archiving. Matching of fragments and aligning them geometrically is based on matching break-curves (curves on a pot surface separating fragments), estimated axes and profile curves for individual fragments and groups of matched fragments, and a number of features of groups of break-curves. Pot assembly is a bottom-up maximum likelihood performance-based search. In our case, associated with subassemblies of fragments is a loglikelihood which is a sum of energy functions. Experiments are illustrated on pots which were broken for the purpose, and on sherds from an archaeological dig located in Petra, Jordan.

Keywords: Object modeling and restoration. Laser scan data analysis. Virtual pots from sherds. Digital Archiving.

1 Introduction

Many archaeological excavation sites are rich in fragments of pots, called *sherds* hereafter, which are either axially symmetric,[1] or look as though they might have such rotational structure but really do not, *e.g.*, the handles of a jar or flat sections of the surface of a plate. There is great scientific and cultural interest in the archaeological community in reconstructing these axially symmetric pots from the sherds found. At present, few pots are reconstructed since the assembly is done manually and is time consuming often taking a few days for one pot. Instead, most pottery is approximately classified using two-dimensional drawing and measurement techniques [6, 5, 12]. This paper presents an approach to the largely automatic estimation of mathematical models of axially symmetric pots from 3D laser scanned data of the sherds. This data is a dense set of 3D points over the outer surface and perhaps other surfaces as well of each sherd, such as along breaks and the inside of the pottery.

1.1 Problem Formulation

The virtual-pot assembly consists of three parts: *(i)* a *description* of our model for sherd generation and sherd-data generation;[2] *(ii)* a *probability measure* for a hypothesized arrangement of sherd data to represent a pot (*i.e.*, how well does an alignment of sherd surface data and break-curve data represent an admissible axially symmetric pot shape); *(iii) approaches to aligning* the sherd data such that the probability of this alignment or of this alignment and prior pot shape information is a maximum, *i.e.*, a Bayesian-based approach to assembling 3D sherd data to produce a mathematical estimate of a complete pot.

The following is our probabilistic model for how the data is generated. There are N axially symmetric pots and A artifacts which are not such pots but look as though they could be sherds of such pots. N and A are either a priori unknown

* Andrew Willis, Stuart Andrews, Jill Baker, Yan Cao, Dongjin Han, Kongbin Kang, Weixin Kong, Frederic F. Leymarie, Xavier Orriols, Senem Velipasalar, Eileen L. Vote, David B. Cooper, Martha S. Joukowsky, Benjamin B. Kimia, David H. Laidlaw, David Mumford.

[1] *I.e.*, the intersection of the pot outer surface with a plane perpendicular to the pot axis is a circle or nearly so.

[2] By sherd generation, we mean the piece of broken pot. By sherd-data generation, we mean the erosion and laser measurements of the eroded piece.

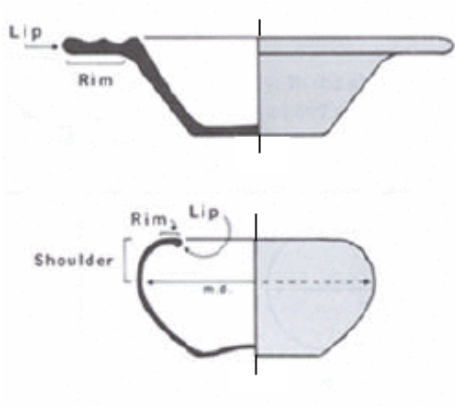


Figure 1: Examples of pottery profiles.

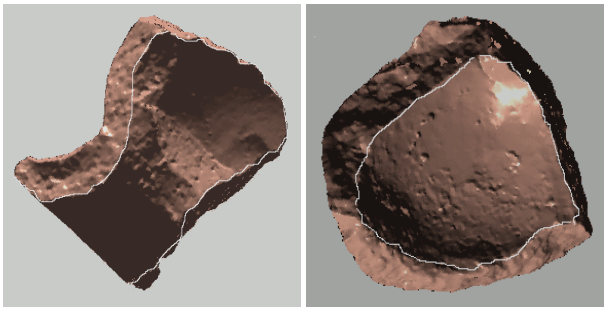


Figure 2: Sherds which have been laser scanned and for which we extracted break-curve measurement data using a dynamic programming approach (see [9]).

numbers constrained to some ranges, or are independent random variables. A pot is in some standard position. It can have a range of shapes and sizes. Its shape is completely specified by an *axis* and an associated *profile curve* (*i.e.*, the silhouette from the side). The profile curve can be *a priori* unknown, or can be a stochastic curve. The pot also has a *thickness-curve*, which is the thickness of the pot wall as a function of height along the pot axis. Again, thickness may be an *a priori* unknown function or may be stochastic. Note that, pots are usually thicker in vicinity of the lip and the wall near the base (Figure 1). In general, thickness at a certain height of the pot is not exactly constant around the surface; there are small variations which are a nuisance when one tries to estimate the axis and profile curve for a single sherd. Hence, we can introduce a *thickness perturbation* function which is a deterministic or a slowly-varying stochastic function.

The pot is further partitioned into sherds by a set of intersecting *break-curves* on the surface (Figures 2 and 3). Break-curves can be *a priori* unknown but satisfying constraints, or can be stochastic. In this paper, we use the former. A constraint that we impose is that break curves intersect *largely* in vertices of *three* break curves. These intersections are *T* junctions or *Y* junctions. Occasionally, four or more break curves will intersect, but that is rare. Some sherds in a pot can be missing. For the i^{th} pot, this number n_i can be *a priori* unknown or can be a random variable. Next, the geometry of each sherd may or may not be deformed. In the latter case, the deformation is due to erosion. This usually takes the form of a smoothing and perhaps a significant amount of material removal from the break surface and perhaps from the outer/inner surface. This erosion can be treated as *a priori* unknown or as stochastic. At present, we use the former assumption. Finally, each sherd undergoes a Euclidean transformation which puts it in its found position. Again, this transformation can be unknown or stochastic. We use the former. With all of these parameters, whether the parameters are assumed to be *a priori* unknown or stochastic with known distributions determines whether the pot assembly estimation is a *Maximum Likelihood Estimation* (MLE), or a *Maximum A posteriori Probability estimation* (MAP), or a combination of the two. The measurement data is a dense collection of 3D points of the inner, outer, and break surfaces of the sherds. We assume these measurements contain errors which are independent identically distributed random variables, which are either 3D around each point to be sensed, or are 1D in the direction perpendicular to the sherd surface. In theory, this should be the complete measurement-error model. In practice, we have found it to be useful to assume independent break-curve and surface measurement-noises.

Note that other features can be useful in helping reducing the search space when attempting matching sherds. For example, paintings and texture patterns on the outer and inner surface, color variations, 3D carving on the surfaces, horizontal circles on the inner/outer surface about the pot axis due to finger smoothing while the pot is spinning on the wheel, etc. We will not consider this additional information in this paper, but we note that our probabilistic framework can accommodate such data. Furthermore, higher level semantic description, as found in the descriptive approaches of traditional archaeology, shall be included in our system, at a later developing phase.

1.2 Recent Attempts at Solving the Sherd Assembly Problem

Classification based on qualitative (*e.g.*, global shape) and quantitative (*e.g.*, color) features of profiles with human-driven pre-processing is being developed at the Technical University of Vienna [8, 18]. The problem of matching 3D curves - taken

to be break-curves of sherds - has been addressed by the Middle East Technical University of Ankara. A simple matching technique relying on an accurate - but potentially non-robust - extraction of the trace of a curve and the computation of curvature and torsion is described, but only tested on artificial data [19]. Within the Digital Michelangelo project, a team is tackling the problem of assembling a Roman marble map from more than a thousand sherds. This jigsaw puzzle is mostly a 2D problem, based on matching break curves and possibly texture patterns [10]. At the University of Athens, a system called the Virtual Archaeologist has been developed to match 3D sherds modeled via surface patches [13]. A region growing method starting from a polygonal mesh of the scanned data is combined with a surface texture measure to locate break surfaces. Simulated Annealing is used to compute pairwise surface matching measures and reduce the error in matching due to varying poses. Some limiting assumptions are made. In particular, two sherds may only share one face (surface). While their approach focuses on surfaces (segmentation and matching), we turn our attention to other global shape features, such as the main axis of rotational symmetry, profile and thickness curves, break curves, and other geometric features such as medial axis symmetries, and ridges and valleys of the inner/outer surfaces [11]. Recently we have achieved high accuracy extraction and matching of 2D and 3D break curves by combining a coarse-scale representation of curves, refined iteratively via a fine-scale elastic matching [9]. Another relevant problem sharing many similarities with the sherd assembly challenge, is to assemble 3D surface fragments obtained by multiple scanning measurements in order to retrieve a complete surface description of an object. Each 3D fragment needs to be optimally matched and blended with others; this involve not only aligning the fragments but possibly deforming these to allow overlapping sub-regions to tightly *blend* together. An interesting recent project along these lines is the Pietà Project at IBM [2].

2 New Methods for Describing Sherds and their Re-Assembly

In this section we present the geometric parameters that uniquely and completely specify the geometry of a sherd or pot, indicate their role in computing the probability of the measurement data of a group of sherds aligned as a hypothesized portion of a possible pot, and comment on the search algorithm for doing MLE or MAP estimation of a complete pot. All of these are *major new contributions* to a concept of virtual pot estimation and specific algorithms for implementing the various pieces of the approach. For the purpose of this paper, we have focused on a subset of the geometric information that can be used. This subset is sufficient for a prototype system. It consists of the outer-surface break-curve common to the pairs of touching sherd data-sets in an aligned group, break-curve vertices, axis/profile curve for each aligned group, and Euclidean transformations that take each sherd from its data-measurement position to its position in an aligned group. We have ignored sherd-thickness geometry, painted or base-relief patterns on the sherd outer surface, circular rings made by potters fingers or an instrument on the inside-surface of a sherd, symmetry set representations [11], etc.

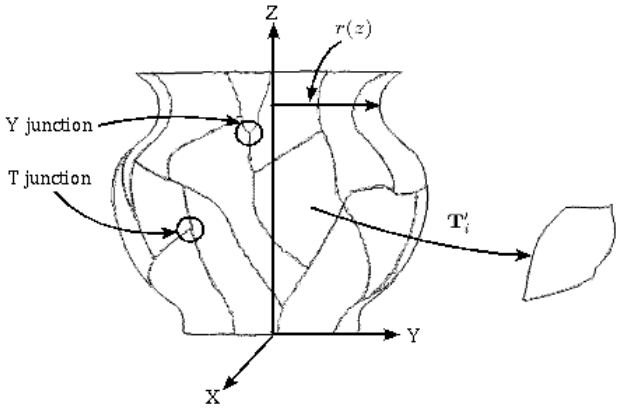


Figure 3: Geometry used in representing a fragmented vessel.

2.1 Parameters to Describe Sherd Geometry and Measurement Data

2.1.1 The Set of Geometric Parameters

Assume the pot is in standard position, *i.e.*, its axis is the z-axis and it sits on the xy-coordinate plane. Then the pot and its fragments are specified by vectors of parameters listed in Table 1. These determine the parameters of the geometry of the fragments in their original positions. We assume that the i^{th} fragment of a vessel in standard position undergoes a Euclidean transformation to move it to a measurement position, which simply consists of a rotation and translation. We call the transformed fragment a "sherd." This is an *a priori* **arbitrary** transformation. The specific models are described later in § 3.

The geometric parameters for transformed fragments, *i.e.*, for sherds, are listed in Tables 2 and 3. The basic transformation equations are then:

symbol	signification
$\mathbf{l}$	z-axis (axis of vessel in standard position)
α	Radius curve for entire vessel (*i.e.*, profile-curve, $r(z)$, viz. z-axis)
α_i	Portion of radius curve for fragment i
α_{ij}	Portion for the union of fragments i and j
β	Break-curves for entire vessel
β_i	Portion of break-curves for fragment i
β_{ij}	Portion shared by fragments i and j

Table 1: Basic geometric parameters. Note that, α's and β's are all *vectors* of model parameters.

symbol	signification
$\mathbf{T}'_i$	Euclidean transformation of i^{th} sherd
$\mathbf{R}'_i$	3×3 rotation matrix - 3 parameters
$\mathbf{t}'_i$	3×1 translation vector - 3 parameters
$\mathbf{l}_i^{'}$	z-axis transformed (via $\mathbf{R}'_i$ and $\mathbf{t}'_i$)
$\mathbf{w}_i$	Unit direction vector of $\mathbf{l}_i^{'}$ (2 parameters)
$\mathbf{p}_{0i} = (x_{0i}, y_{0i})$	Line location for $\mathbf{l}_i^{'}$ (*e.g.*, intersection with xy-plane)
α'_i	Profile curve for i^{th} sherd
β'_i	Transformed break-curve β_i
$\mathbf{T}'_{ij}$	Transformation to align the datasets of sherd i with those of sherd j
$\mathbf{l}'_{ij}$	Axis for the aligned datasets i and j

Table 2: Geometric parameter vectors for sherds under Euclidean transformations. Boldface indicates a vector or matrix.

$$\begin{aligned} \mathbf{l}_i^{'} &= \mathbf{T}'_i(\mathbf{l}) \\ &= \{\mathbf{w}_i, x_{0i}, y_{0i}\} \,, \\ \alpha'_i &= \mathbf{T}'_i(\alpha_i) \,, \\ \beta'_i &= \mathbf{T}'_i(\beta_i) \,. \end{aligned}$$

The *Measurement Data* parameters are listed in Table 3.

Sherd measurement-data is provided by a Shapegrabber laser/camera scanner [1]. It produces 15,000 3D points/sec. at a resolution and accuracy of the order of 0.25mm. All of these points are surface measurements, *i.e.*, measurements of outer, inner, and break surfaces including the 3D ridges that separate these surfaces. For the algorithms used in this paper, it has been convenient to extract and use two subsets of the measurement data, *i.e.*, those points which are measurements of a sherd outer-surface and those which are measurements of a sherd outer-surface break-curve. Our approach to doing this

symbol	signification
$\mathbf{U}'_i$	Outer surface data for sherd i
$\mathbf{V}'_i$	Break-curve data for sherd i
$\mathbf{V}'_{i,j}$	Break-curve data for i^{th} sherd over i, j^{th} sherd break
$\mathbf{U}'$	Vector composed of all the $\mathbf{U}_i$
$\mathbf{V}'$	Vector composed of all the $\mathbf{V}_i$

Table 3: Parameter vectors for measurement data.

for the data sets used in this paper is fast, based on clustering, and is described in [21]. An alternative approach is to use some sort of model to estimate break-curves as ridges [9], see Figure 2, and then extract those data points that are close to these estimated ridges.

2.1.2 Assumptions

A. 1 (Surface measurement points are *iid* $N(0, \sigma_s^2)$) *These are independent, identically distributed, Gaussian perturbations perpendicular to the surface and having mean* 0 *and variance* σ_s^2.

See [4] for a justification of this model.

A. 2 (Break-curve measurement points are *iid* $N(0, \sigma_b^2)$) *Independent, identically distributed, Gaussian perturbations in 3-space about each point on the true break-curve, with mean* 0 *and variance* σ_b^2.

Note that, better, but more complicated models can be used, such as:

1. Noise dependent on the angle between the incident laser measurement ray and the surface normal.

2. Incorporating a sherd break-curve erosion model.

2.1.3 Example of one of these probability densities

Assume a plane specified by point $\mathbf{p}_0$ and unit vector $\mathbf{w}$, then the probability density for the surface measurement point $\mathbf{p}$ is given as:

$$P(\mathbf{p}|\mathbf{w}, \mathbf{p}_0) = \frac{1}{\sqrt{2\pi\sigma_s^2}} \exp\left(-\frac{1}{2\sigma_s^2}\left[\mathbf{w}^t(\mathbf{p} - \mathbf{p}_0)\right]^2\right). \quad (1)$$

2.2 The Probability of All the Data given All the Global Geometry Parameters

The joint probability of all surface and break-curve data given a profile curve, a break-curve, and all transformations to sherds, can be written as:

$$P(\mathbf{U}', \mathbf{V}'|\mathbf{T}', \alpha, \beta) = \prod_i P(\mathbf{U}'_i, \mathbf{V}'_i|\mathbf{T}'_i, \alpha, \beta) \quad (2)$$
$$= \prod_i P(\mathbf{U}'_i, \mathbf{V}'_i|\mathbf{T}'_i\mathbf{l}', \alpha'_i, \beta'_i)\,.$$

MLE is the vector of values for $\mathbf{T}', \alpha, \beta$ that maximizes (2). A combined **MAP** and **MLE** is the vector of values that maximizes $P(\mathbf{U}', \mathbf{V}'|\mathbf{T}', \alpha, \beta)P(\alpha)$, where $P(\alpha)$ is the *a priori* probability distribution for the vessel profile curve. Note that, eqn.(2) illustrates the role of $\mathbf{T}'_i$, *i.e.*, it translates $\mathbf{l}$, α_i and β_i to the i^{th} sherd position. Also note that $\beta_i = \mathbf{T}_i(\beta'_i)$ where $\mathbf{T}_i = (\mathbf{T}'_i)^{-1}$.

2.3 Combining Sherd Data using MLE

The following is our approach to sherd assembly. The fundamental features to be used for each sherd are outer surface break-curve, axis/profile curve, and vertices of the break-curve. For axis/profile curve, it is not just the shape of the profile curve that is important, but also its orientation and distance from the axis. Distance from the axis at each point along the profile curve is the radius of the pot at that height along the axis. The loglikelihood of a group of two or more sherds,[3] henceforth called a **configuration**, is the negative of the sum of *(i)* the loglikelihoods of the break-curve data fit to the estimated break-curve between pairs of adjacent sherds and *(ii)* the loglikelihood of the surface-data fit to the estimated axis/profile curve for the configuration (*i.e.*, the assembled group of sherds). In § 3.4 we show that this is the sum of a number of energy functions, which result in a cost (error) function for the assembly representing what should be minimized. The parameters to be estimated for this purpose are the Euclidean transformation for each sherd, and the resulting outer-surface break-curve and axis/profile curve.

The data-set to think about in the following is a set of 100 to 200 sherds, which, as indicated in § 1.1, could come from a number of pots as well as from other objects. The search aims at grouping and aligning sherds along contours of constant cost. Hence, we proceed as follows:

1. Estimate the axis/profile-curve for each sherd: associate the resulting error measure, *i.e.*, the sum of squared distances from the surface data measurements to the MLE for the sherd 3D surface; see § 3.2 thru 3.3. This is computationally costly.

2. For each pair of sherds, estimate all reasonable alignments. This is computationally fast; see § 3.1 thru 3.1.2. Each aligned pair forms a *configuration*. Then, a consistency check is run. This is to see whether the two sherds *overlap* significantly. If they do, that sherd-pair alignment is not considered further.

3. For each configuration from step 2, improve the alignment and estimate the axis/profile-curve for the improved-aligned pair. The improved alignment is obtained by minimizing the sum of the break-curve data-alignment errors and the surface-data model-fit errors. Since this can start from the results of steps 1 and 2, the computation cost is significant but not as large as that in step 1. The resulting group of configurations, together with the cost (*i.e.*, error measure associated with the pair) for each configuration, is stored in a table in order of increasing cost. Also included in this table are the 1-sherd configurations from step 1. The associated cost for each 1-sherd configuration is based only on the outer-surface measurement data; see § 3.4.

4. Explore merging pairs of configurations, where each merge is of two 2-sherd configurations or of one 2-sherd configuration with one 1-sherd configuration. Configurations to be aligned are those which result in new configurations having minimal associated costs. In general, only a small fraction of all possible alignments need to be considered for two reasons:

   (a) Those alignments involving pairs of configurations where each configuration in the pair has a high cost, will not result in a new configuration having small cost and, thus, only configurations near the top of the table need to be considered.

   (b) Since an alignment of two configurations involves aligning one or more pairs of sherds, each sherd in such a pair belonging to a different configuration, a first rough alignment of the two configurations can be obtained from table lookup, and alignment geometries that result in significant configuration overlaps are quickly detected and discarded.

5. Lastly, improve all remaining minimal-cost alignments through re-estimation of break-curves and profile-curves.

[3] *I.e.*, the log of the probability of the measurement data given the geometric arrangement of the sherds.

The resulting configurations along with their costs are added to the table, and the alignment of pairs of configurations in the updated table continues. This alignment proceeds along contours of constant cost.

Note that, sherds containing portions of both pot tops and pot walls, and sherds containing portions of both pot bottoms and pot walls are especially useful. Since they contain high-curvature ridge structure, they provide more accurate pot-axis estimates than do low-curvature sherds containing portions of pot walls only. Also, these sherds are pot-geometry delimiters: this information can be used effectively in determining which configurations to align next and thus reduce computational cost in the sherd assembly process

Various other issues have been glossed over in this discussion. For example, the costs that measure how well the break-curve data fits the estimated break-curve and how well the surface-data fits the 3D surface model determined by the estimated axis/profile-curve involve numbers of data points which differ by two or three orders of magnitude. Hence, some normalization must be introduced in order that one cost does not completely overwhelm the contribution of the other. Other normalizations may be necessary as well because the simple probabilistic distributions we are using may not be representative enough for the full range of geometric deformations to be encountered. A second observation is that even though only a few of the many sherds associated with a pot may be present in the data-set being processed and even if none or only a few of them share a common break-curve, it may still be possible to estimate the virtual pot based on the set of axis/profile-curve estimates for these sherds. In theory, this can be done if the vertical extents of the sherds present covers the vertical extent of the entire pot. A third comment is that in the preceding discussion, all the sherd surface and break-curve measurement data gets processed each time a sherd is involved in a configuration appearing in a pair of configurations that are being aligned. As mentioned in § 3.4, the possibility exists of using the estimated geometric parameters and their covariance matrices to do the alignment rather than using the raw data. This would speed up the required computation by one to two orders of magnitude. See [4] for basic ideas.

3 Sherd-Data Alignments and Pot Geometry Estimation

3.1 Modeling and Alignment of Break-Curve Data for Pairs and Triples of Sherds

In this section we first describe our approach to aligning a pair of sherds based on their break-curve data, and then show that this approach is the MLE of the portion of the break-curve common to the pair of fragments. The representation used for the portion of break-curve common to a pair of fragments is a finite sequence of points on this break-curve segment. Hence, the alignment of the two sherds involves *(i)* keeping the measurement-data for one sherd fixed, and, *(ii)* determining the Euclidean transformation of the measurement-data for the other sherd. This is performed such that *(i)* the break-curve data matches well at a finite number of points, and, *(ii)* the tangents for the outer-surface data of the two sherds at a pair of matching break-curve points are roughly the same. This ensures that the two break-curve data-sets match well and that the estimated outer-surfaces fit together smoothly with continuity of the normals to the surfaces.

We begin by matching the break-curve measurements for each sherd starting where it has vertices. These vertices occur where three, and in some cases four or more, sherd vertices meet, as a result of the breaking of the pot at that point. In the prevalent case of a three vertex points, the vertices can meet in Y-junctions or T-junctions. For T-junctions there are really only 2 vertices to consider,[4] while for Y-junctions, there are three vertices (Figure 3).

There are typically four to six such vertices for a break-curve of a sherd. Hence, there are roughly 65 such vertices for the 13 sherds resulting from one of the pots we have scanned.[5]

We try to match a pair of sherds starting at the vertices. Toward this end, we put down a finite number of points along the measured break-curve for each sherd starting at a vertex. Successive points are a Euclidean distance "d" from one another. For each pair of sherds and each pair of vertices, one from each sherd, we try to match the corresponding pairs of data points, one point from each sherd. The error criterion is a sum of squared errors, and the error function also contains the sum of squared differences in the measured unit normals to the surfaces at the data points used in the matches. Let us consider the break-curve data point $\mathbf{p}_{im}$, from sherd i, and denote $\mathbf{R}$ and $\mathbf{t}$ as the estimated rotation matrix and translation vector coming as solution to equation (3) below, where λ is a chosen positive constant, $\mathbf{n}_{im}$ is the unit normal to the surface data of

[4] One vertex remains "hidden" on the "top" of the T.

[5] Experiments were run on a pot broken for the purpose in order to have ground truth.

sherd i at the break-curve data-point $\mathbf{p}_{im}$ and $\mathbf{M}$ is the number of data points used, which is 5 at present. Then we have the optimization program:

$$\min_{\mathbf{R},\mathbf{t}} \sum_{m=1}^{\mathbf{M}} \left[\|\mathbf{p}_{im} - \mathbf{R}\mathbf{p}_{jm} - \mathbf{t}\|^2 + \lambda \|\mathbf{n}_{im} - \mathbf{R}\mathbf{n}_{jm}\|^2 \right] . \tag{3}$$

The resulting alignments are very good in practice. In general, the 5 data points occupy only a small portion of the break-curve segment common to the pair of sherds. In some instances, it may be useful to try a few starting locations for the first point at a vertex. Note that, in our experiments thus far, a human locates the break-curve vertices. We do have automatic vertex-detection running, but have not yet tested it extensively. Also, from a practical point of view, 3D scanning of sherds takes a certain amount of time, and for a human to mark the locations of ambiguous vertices takes comparatively negligible time.

Equation (3) has an *explicit* solution. This is a linear least-squares problem, and the solution is computed at little cost.[6] How much computation, then, is involved in examining data for all pairs of sherds in a archaeology site location in order to check for acceptable matches and alignments? Let us assume that a sherd break-curve has 4 vertices on average, and that 200 sherds must be considered at a loci. Then, there are roughly $4 \times 200 = 800$ break-curve segments that must be compared for possible matches. Hence, there are roughly $(800^2/2) = 320,000$ pairs that must be checked for matches. If each match computation takes about one millisecond, the total of all match checks and alignments takes about 6 minutes once the sherds have been scanned and their 3D break-curve and outer-surface data are available. Stored with each matched pair is the sum of squared errors between the five pairs of corresponding points used in the alignment computation. We denote this error $e_{\mathbf{T}}$:

$$e_{\mathbf{T}} \equiv \sum_{m=1}^{\mathbf{M}} \|\mathbf{p}_{im} - \mathbf{R}\mathbf{p}_{jm} - \mathbf{t}\|^2 , \tag{4}$$

where $\mathbf{T}$ is the transformation that produced the best alignment.

3.1.1 Break-Curve and Surface-Tangent Alignment Experiments

Experiments were run on a pot that we broke in order to have sherds that we knew could be matched, and in order to have ground truth. The pot was broken in a way to produce sherds consistent with those found at archaeology sites. There are 13 sherds in the pot used in this example. Figure 4 shows matched and aligned break-curve data for 8 pairs of sherds. The piece of break-curve used for each matching is indicated by a pair of numbers, one on each side of the common break-curve and located at the vertices used. A "1" and a "1′" are the pairs of numbers. The four examples in the first row , (a)-(d), represent correct matches with resulting alignments. The four examples in the second row, (e)-(h), are incorrect matches and alignments. For the four incorrect matches shown, the matching error is small, *i.e.*, the five pairs of points match well. It does occur in practice that an incorrect match may have a smaller match-error than will a correct match. Note how in Figure 4.(g) the break-curve data for the two sherds result in a very good fit, even though the match is incorrect. There are at least *two ways* to detect matches that should be *recognized as incorrect*. First, portions of the sherd surface regions bounded by the break-curve data *overlap*, as illustrated in cases (e) and (f), and this can be detected automatically and quickly. Second, the *profile curve* for the surface data for each matched pair of sherds will *not be accurate*, *i.e.*, there will be a sizable fitting error for the estimate of the profile curve for the data for the pair of sherds, which is also easily detected. This is illustrated by cases (g) and (h).

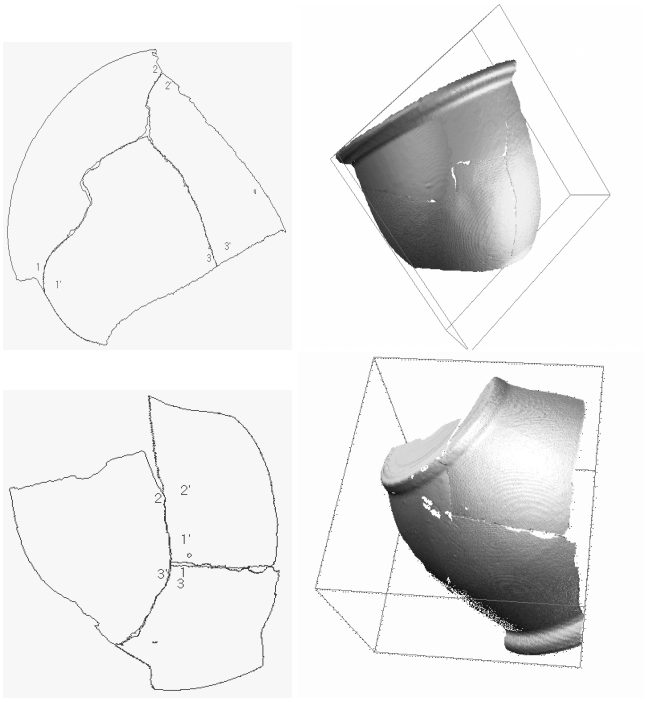


Figure 5: Break-curve triplet matches.

[6] It takes less than a millisecond of CPU time on our PC and UNIX workstations.

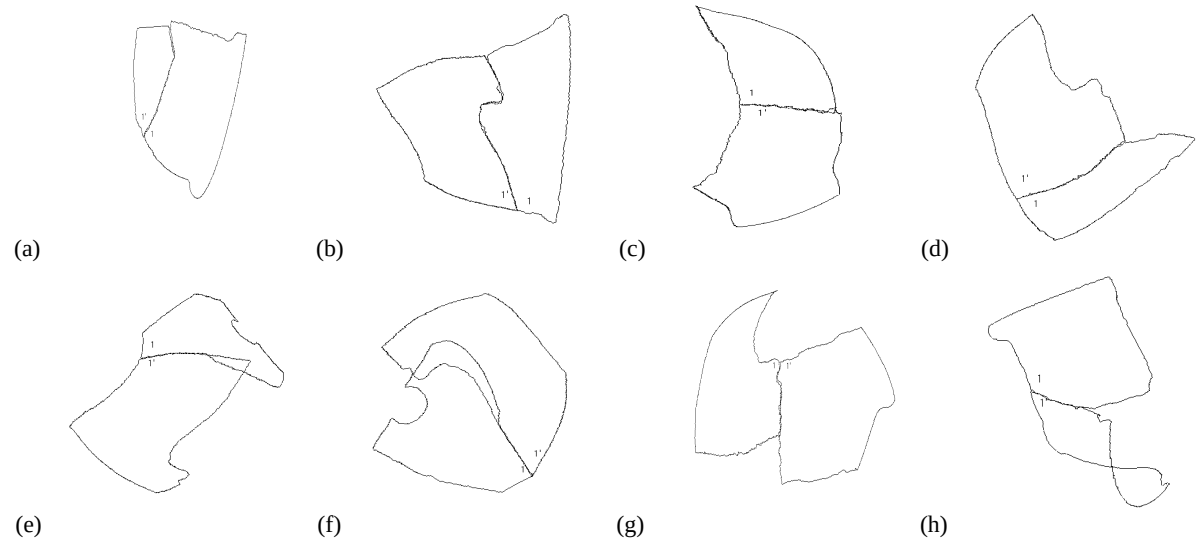


Figure 4: Break-curve pair matches.

Aligning Triplets of Sherds. Two examples of correctly-matched triplets are shown in Figure 5. With each there are two views of the assembled sherds in order that one be able to visualize the quality of the sherd assembly. In these cases one sherd is held fixed and each of the other two is transformed, which means that two transformations must be estimated or 12 parameters. These parameters are estimated simultaneously using all the matching data in a single cost function. Hence, the estimate of a pair of transformations is based on the matching of 15 pairs of points and associated pairs of normals. Unfortunately, in this case the estimation is *nonlinear*. Note that the matching is not perfect, especially for the bottom of the pot in the second triplet. The matching can be improved by using data at the other vertices as well, by automatically adjusting the locations of the first points at the vertices better, and, more generally, by using points along the entire common portions of the break-curves and using denser sets of points and dynamic local curve-scaling during the point matching. The approach extends immediately to four or more groups of sherds, where $k-1$ transformations must be estimated simultaneously for optimally matching k sherds.

3.1.2 Joint Estimation of β_{ij} and T for MLE Aligning

We show that the solution of equation (3) is the MLE for $\mathbf{T}$ and β_{12}. Where $\mathbf{T}$ is the transformation that aligns sherd 2 break-curve data with that of sherd 1.

For the approach in § 3.1, we have represented break-curve segment β_{12} by a sequence of M points p_m, $m = 1, \ldots, M$ on β_{12}. Hence, for this curve-representation, β_{12} is the column vector $(p_1^t, p_2^t, \ldots, p_M^t)^t$ which has $3M$ components.

The data generation probability model we use is the following. Denote by μ a point on the true break-curve segment β_{ij}. Denote by $\mathbf{q}$ a measurement of this point, and by Σ it's covariance matrix. Then we assume $\mathbf{q}$ has the *pdf* (probability density function):

$$P(\mathbf{q} \mid \mu, \Sigma) \quad = \quad (2\pi)^{\left(-\frac{3}{2}\right)} |\Sigma|^{-\frac{1}{2}} \quad \exp\left(-\frac{1}{2}(\mathbf{q}-\mu)^t \Sigma^{-1}(\mathbf{q}-\mu)\right) .$$

For § 3.1, we took $\Sigma = \sigma_b^2 \mathbf{I}$ where $\mathbf{I}$ is the 3×3 identity matrix. Then,

$$P(\mathbf{q} \mid \mu, \Sigma) = (2\pi\sigma_b^2)^{\left(-\frac{3}{2}\right)} \exp\left(-\frac{1}{2\sigma_b^2} \|\mathbf{q}-\mu\|^2\right) .$$

Now, fix sherd 1. It has break-curve measurement points $\mathbf{p}_{1m}$, $m = 1, \ldots, M$ (see § 3.1). Sherd 2 has break-curve measurement points $\mathbf{p}_{2m}$. These get transformed by Euclidean transformation $\mathbf{T}$ to produce points $\mathbf{p}'_{2m} = \mathbf{T}(\mathbf{p}_{2m}) = \mathbf{R}\mathbf{p}_{2m} + \mathbf{t}$, which are to align with the $\mathbf{p}_{1m}$. We assume $\mathbf{p}_{im}$, $(i = 1, 2; m = 1, \ldots, M)$, are statistically independent measurements, and that $\mathbf{p}_{1m}$ and $\mathbf{p}'_{2m}$ are measurements of point $\mathbf{p}_m$ which is on the true break-curve β_{12}. Denote by $\{\mathbf{p}_{im}\}$ the set $\{\mathbf{p}_{im} : i = 1, 2; m = 1, \ldots, M\}$. Then,

$$\ln P\left(\{\mathbf{p}_{im}\} \mid \{\mathbf{p}_m\} \cup \mathbf{T}\right) \quad = \quad \ln((2\pi\sigma_b^2)^{-3M}) -$$

$$\left(\frac{1}{2\sigma_b^2}\right)\sum_{m=1}^{M}\left[\,\|\mathbf{p}_{1m}-\mathbf{p}_m\|^2 \;+\; \|\mathbf{p}'_{2m}-\mathbf{p}_m\|^2\,\right]. \quad (5)$$

The MLE of $\mathbf{T}$ and $\mathbf{p}_m : m = 1,\ldots,M$, are the values for which (5) is maximum. These can be found by fixing $\mathbf{T}$ and solving for the optimum $\mathbf{p}_m$, and then solving for the optimum $\mathbf{T}$. The values for $\mathbf{p}_m : m = 1,\ldots M$, for which (5) is maximum are $\widehat{\mathbf{p}}_m = (\mathbf{p}_{1m} + \mathbf{p}'_{2m})/2$, and then (5) becomes:

$$\ln((2\pi\sigma_b^2)^{-3M}) - \left(\frac{1}{4\sigma_b^2}\right)\sum_{m=1}^{M}\|\mathbf{p}_{1m}-\mathbf{p}'_{2m}\|^2 \;=\; \ln((2\pi\sigma_b^2)^{-3M}) - \left(\frac{1}{4\sigma_b^2}\right)e_{\mathbf{T}} \quad . \quad (6)$$

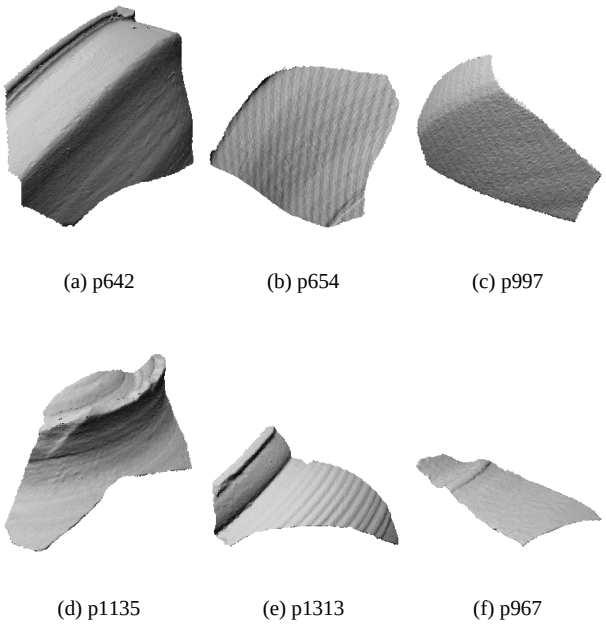
Finally, (6) must be maximized with respect to $\mathbf{T}$, equivalently, minimize $e_{\mathbf{T}}$ with respect to $\mathbf{T}$ to obtain $\widehat{\mathbf{T}}$. The MLEs $\widehat{\mathbf{T}}$ and $\widehat{\mathbf{p}}_m,\ m = 1,\ldots,5$ are exactly those used in matching and alignment in § 3.1.

3.2 Sherd Surface Characterization

Whereas effective algorithms have been developed for estimating the axis and associated profile curve for a sherd that comprises a large portion of the surface of an axially symmetric pot, there do not appear to be effective algorithms for estimating a pot axis when the sherd is a *very small portion of the pot* or when the *profile curve is complicated*. But these cases are important in practice and challenging in concept. In [21], we introduce three new approaches to estimating axis/profile-curves, quantitatively evaluate their performances, and discuss their relative strengths. They all are more accurate than algorithms presently in the published literature. We present two of them in § 3.2.2 and § 3.3, respectively. One estimates the axis and profile-curve pair jointly through maximization of a performance function which is close to MLE. The other estimates the axis based on local data curvature-estimates, and then obtains a profile curve based on the estimated axis. Both are tested on a representative spectrum of challenging outer-surface measurement data of sherds from Petra, Jordan [7]. The algorithms of § 3.2.2 and 3.3 are the most accurate. The third algorithm presented in [21] is less accurate, but is computationally almost two orders of magnitude faster.

3.2.1 Surfaces of Revolution

Surfaces of revolution are commonly encountered in the field of elementary differential geometry [15]. A surface of revolution S is obtained by revolving a planar curve C about a line $\mathbf{l}$. C corresponds to the previous notion of profile curve, α, and $\mathbf{l}$ is the *axis* of S. When the z-axis is taken as the axis of revolution for the profile curve the surface S may be defined parametrically simply as:



(a) p642 (b) p654 (c) p997

(d) p1135 (e) p1313 (f) p967

Figure 6: A spectrum of archaeological sherds from the Great Temple site of Petra, Jordan [7], used to illustrate our axis/profile-curve estimation.

$$S(r,\,\theta,\,z) \equiv (r(z)\cos\theta,\; r(z)\sin\theta,\; z)\,. \quad (7)$$

With this parameterization, the curves $r = constant$ are *parallels* of S and the curves $\theta = constant$ are *meridians* of S.

3.2.2 Mathematical Models

We have parameterized the axis of symmetry, $\mathbf{l}$, using the standard parametric equation of a 3D line:

$$\begin{aligned} x &= m_x z + b_x\,, \\ y &= m_y z + b_y\,. \end{aligned} \quad (8)$$

These equations contain four unknown parameters. Two of these, m_x and m_y, describe the slope of the line when it is projected onto the xz-plane and the yz-plane, respectively. The remaining two parameters, b_x and b_y, specify where the line intercepts the xy-plane at $z = 0$.

The profile curve model for the pot surface must be general enough to represent a wide variety of possible shapes.

Figures 1 and 6 illustrate the range of geometric complexity which pottery sherds from only a single site may exhibit. It is important to note that some of the pot sherds shown cannot be represented by explicit functions with respect to the axis. For example, $r(z)$ is multivalued for values of z near the top of the sherd in Fig. 6.(d). Consequently, we propose to use *implicit polynomial models*, rather than explicit profile models, to represent the profile curve in the approach described in this section. The approach in § 3.3 can also handle these multi-valued radius situations because it does not use a profile-curve model in the axis estimation.

Implicit polynomial surfaces and curves are very powerful shape models which are capable of representing the wide variety of geometries involved in this problem [17, 16, 3]. The general form of an implicit curve of degree d has $[(d+1)(d+2)/2]$ unknown coefficients and takes the following form:

$$f_d(r,z) = \sum_{0\le j+k\le d;\ j,k\ge 0} a_{jk} r^j z^k = 0\,. \qquad (9)$$

Here, d is a parameter which is related to the geometric complexity of the pottery sherd to be estimated. Typically one assigns a value to d which is large enough to represent all objects of interest. In this way, objects which may have little geometric complexity are described as degenerate cases of the more complex model. For the artifacts in this paper, all experiments are performed with $d = 6$.

To estimate the polynomial model, the (r,z) component of the spatial data and the normal data are computed. Note that this can also be viewed as an orthogonal projection of the data from $(r,\ \theta,\ z)$ to (r,z). This projection into rz-space preserves the distance relationship between the axis and each point $\mathbf{p}_i$ in xyz-space, but discards the component of the surface normal in the θ direction. Hence, for any 3D normal, $\mathbf{n}_i$, the corresponding projected normal, $\mathbf{n}_i^p$, may not be of unit length, *i.e.*, $\|\mathbf{n}_i^p\| \le 1$. These parameters define the objective function (10) below, which is a modified version of the energy function in [16], for estimating the profile curve coefficients:

$$e_{grad} = \sum_{i=1}^{I} (f_d^2(r_i,\ z_i) + \mu\ \|\mathbf{n}_i^p - \nabla f_d(r_i,\ z_i)\|^2)\,, \quad (10)$$

where $\nabla f_d(r_i,\ z_i) = \left[\ \frac{\partial f_d}{\partial r}\quad \frac{\partial f_d}{\partial z}\ \right]^t$ denotes the gradient. Note that, $f_d^2(r_i, z_i)$ is the data fitting error in (10). The remaining portion of the summation is relatively small and is for the purpose of regularization in the minimization.

Model Estimation Algorithm This method utilizes a two-step iterative algorithm to estimate the axis and associated profile curve which best describes the observed 3D data. The non-linear iterative minimization is:

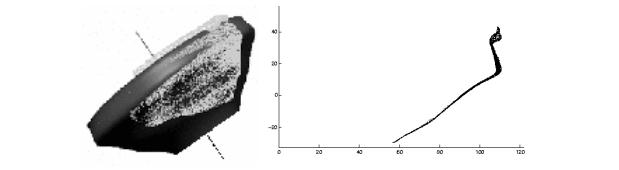
1. Based on the value of the objective function after the preceding iteration, choose a new value for the parameter vector $(m_x,\ m_y,\ b_x,\ b_y)$ specifying the pot axis.
2. Minimize e_{grad} in eqn. (10) by solving for the optimum profile-curve polynomial coefficients a_{jk} (see eqn. (9)). This is a linear least-squares problem and there is a fast-to-compute explicit solution.
3. Return to 1 or stop.

Since the surface model depends on the axis, the resulting objective function is highly non-linear. Consequently, convergence to a local minimum may occur if minimization is started far from the true parameter value. The estimation algorithm needs only a hypothesized axis of symmetry in order to begin. In practice, we begin with the axis estimate provided by an improved version [21] of the Plücker coordinates method as described in [14, 20] (some additional information is given in § 3.3). This initial estimate is very fast — probably less than a millisecond. The total computation time[7] is of the order of 1 to 3 minutes for roughly 3,000 data points.

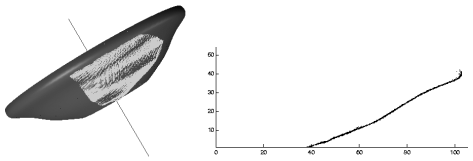
3.2.3 Results on Archaeological Data

Figure 6 shows a set of six sherds from the Brown University archaeological dig at Petra, Jordan [7], which were used to evaluate the algorithm above. For each of the data-sets the parameters, *i.e.*, axis and profile curve coefficients, of the surface of revolution were estimated. These parameter values were then used to generate the portion of axially-symmetric pot surface over the extent of the sherd-axis covered by the data, and the sherd surface-data is overlaid in white on this slab of estimated pot-surface. Also, upon rotating the estimated sherd data-axis to coincide with the z-axis, the distance of each 3D surface data-point is plotted as a function of z. These points would lie exactly on the sherd profile-curve if the sherd-axis were estimated without error and if the data-points were noiseless. However, since the data-points are not noiseless and the true axis estimate is not without error, the plotted points lie in a swath about the true profile curve. In general, increasing error in estimated axis position results in increasing data-swath spread about the true profile curve. The narrowness of the swathes of data in Figure 7 are indicative of the accuracy of our estimates for axes for these difficult examples. Covariances for the axis parameters appear in [21]. Note that, some of the sherds shown are very small, *e.g.*, Figures 6.(b),(f), and

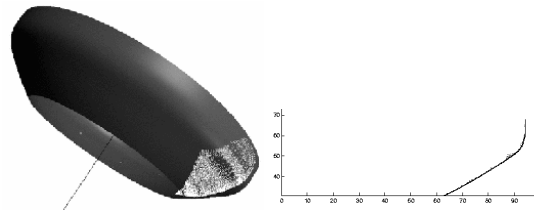
[7] Implemented using the C programming language.



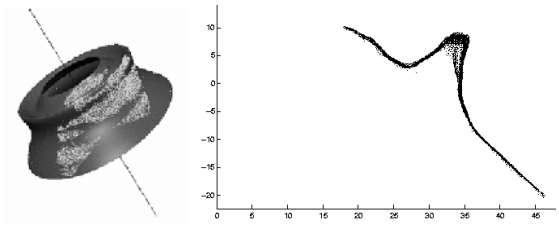
(a) p642



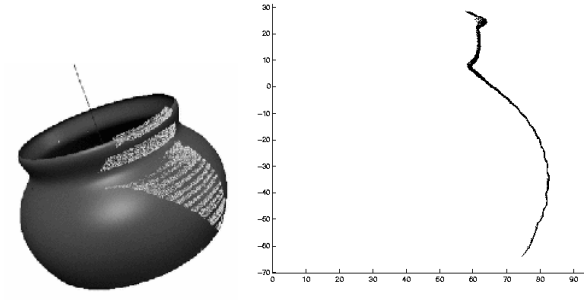
(b) p654



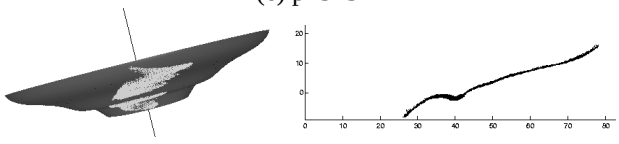
(c) p997



(d) p1135



(e) p1313



(f) p967

Figure 7: Axis/profile-curve estimates for the sherds of Figure 6, and visualization of these estimated 3D pot models over the associated z-axis intervals.

it is agreeably surprising that useful axis/profile-curve pairs can be found for these. Sherd 6.(b) appears to be from the same pot as is 6.(a), and though the two sherds do not share a common break-curve segment, 6.(b) appears to cover the same axis height-range as does the portion of 6.(a) from the surface-ridge down. Also, 6.(d) is an example of a small sherd having a somewhat complicated multi-valued radius function $r(z)$.

3.3 Extracting the Axis of Revolution and Profile Curve via Spheres of Curvature

This section describes an approach to estimating the pot axis directly. Given a perfect surface of revolution, one of the two families of lines of curvature are circles of revolution (parallels of the surface) with centers on the axis and the other are the generator curves (meridians of the surface), lying in planes containing the axis. But, unfortunately, pot sherds are far from perfect, *e.g.*, due to imperfections in the modeling process and subsequent erosion. They are noisy surfaces of revolution.

We define *spheres of curvature* to be the spheres centered at one of the principal centers of curvature and having radius equal to the corresponding radius of curvature. These spheres are tangent to the surface. It is easy to show that for each point on the surface, the center of the sphere of principal curvature corresponding to the parallel circles is on the main axis (or revolution). By finding the line which minimizes the least squares distance from the estimated centers to it, we can find the main axis and its profile curve.

One useful feature of a surface of revolution is that the principal curvature corresponding to the parallel circles does not depend on second derivatives; we denote this principal curvature κ_π. Given a set of m 3D data-points from a surface of revolution, let $\mathbf{p}_i$ and $\mathbf{n}_i$ be the m 3D data points and their corresponding normals. Suppose the axis of revolution is the line $\mathbf{l}$. This axis can be specified by a point $\mathbf{p}_0$ on $\mathbf{l}$ and a unit vector $\mathbf{w}$ corresponding to the direction of $\mathbf{l}$. We make the point $\mathbf{p}_0$ unique by requiring $\mathbf{p}_0 \cdot \mathbf{w} = 0$. For any point $\mathbf{p}$ on the surface, suppose the normal at that point is $\mathbf{n}$. It can be shown that

$$\kappa_\pi = \frac{\|\mathbf{n} \times \mathbf{w}\|}{\|(\mathbf{p} - \mathbf{p}_0) \times \mathbf{w}\|} .$$

All the centers of the sphere of curvature corresponding to κ_π should be on the axis of revolution, hence we can minimize following function:

$$f(\mathbf{p}_0, \mathbf{w}) = \sum_{i=1}^{m} \left\| (\mathbf{p}_i - \mathbf{p}_0) \times \mathbf{w} - \frac{\|(\mathbf{p}-\mathbf{p}_0)\times\mathbf{w}\|}{\|\mathbf{n}\times\mathbf{w}\|} (\mathbf{n}_i \times \mathbf{w}) \right\|^2 .$$

There are six parameters in the function f, which are not independent. They satisfy the following constraints:

$$\mathbf{p}_0 \cdot \mathbf{w} = 0 \text{ and } \|\mathbf{w}\| = 1 .$$

We want to reduce the six dependent parameters to four independent parameters. Define the matrix R as following:

$$R = \begin{bmatrix} \cos\phi\sin\psi & \sin\phi\cos\psi & \sin\psi \\ -\sin\phi & \cos\phi & 0 \end{bmatrix} ,$$

then the vector $\mathbf{w}$ can be represented as:

$$\mathbf{w} = \begin{bmatrix} -\cos\phi\sin\psi & -\sin\phi\sin\psi & \cos\psi \end{bmatrix}^T .$$

Define $\mathbf{p}'_0 = (x'_0, y'_0)^T = R \cdot \mathbf{p}_0$. Then the function f can be represented as a function of four parameters, ϕ, ψ, x'_0, y'_0:

$$f(\phi, \psi, x'_0, y'_0) = \sum_{i=1}^{m} \left\| (R \cdot \mathbf{p}_i - \mathbf{p}'_0) - \frac{\|(R\cdot\mathbf{p}-\mathbf{p}'_0)\|}{\|R\cdot\mathbf{n}_i\|}(R \cdot \mathbf{n}_i) \right\|^2$$

Potmann *et al.* [14] proposed a direct solution to reconstruct helical surfaces or surfaces of revolution using line geometric concepts. Their algorithm is based on the fact that the normals of these surfaces lie in linear complices. Using normalized Plücker coordinates, the least squares distance between a line and the set of normals can be represented by a positive semidefinite quadratic form. Minimizing that form is then reduced to a generalized eigenvalue problem. We use this solution as our initial starting point. After we get the axis of revolution, we can calculate the distance from each data point to the axis, and do a cubic spline fit to get the profile curve.

Experiments were run on the dataset used in § 3.2.3. The data for each sherd is a dense set of unorganized 3D points and their estimated normals. Figure 8 is an example of experimental results. The statistics shown and others were gathered for the sherds in § 3.2.3. The accuracy (covariance matrix) of this axis estimator is very good – comparable to that in § 3.2.2 – better in some cases and not as good in others. See [21] for a quantitative comparison of these and another algorithm.

We use the global features of the surface of revolution, and avoid computing high order derivatives, hence this method is robust to noisy data. Due to its least squares nature, it may be sensitive to outliers in the current format. This can be overcomed by weighted least squares. The goal function is easy to calculate, and the minimization can be carried out by general iterative methods. In its present implementation, the computation time is a few minutes for a dataset of a few thousand points.[8]

[8] Implemented in the MATLAB® software environment.

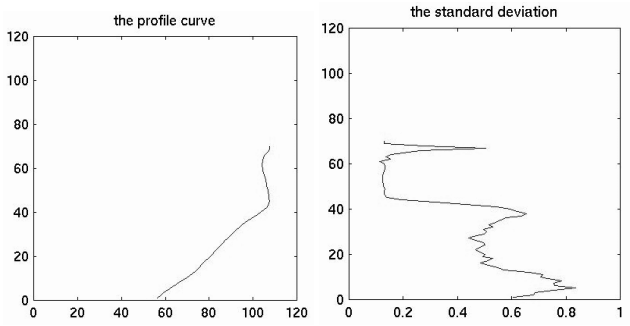


Figure 8: On the left is the profile curve for sherd 642 (see Figure 6.(a)); on the right is the standard deviation of the estimated profile curve as a function of height along the sherd axis, based on axis estimates using 500 independent bootstrap samples.

3.4 Aligning Sherd-Data for Pairs by MLE of Break-Curves and Axis/Profile-Curves

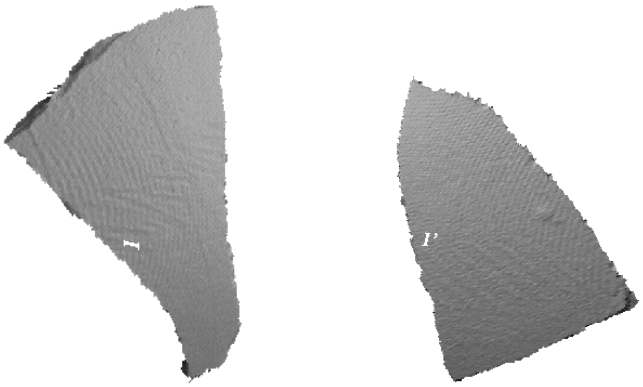
Our approach to estimating the optimum joint geometry of a pair of sherds based on their break-curve and surface datasets jointly is described in this section. In § 3.1 we described sherd-pair alignment based on break-curve data alone. In § 3.2 and 3.3 we described sherd-surface model estimation, *i.e.*, axis/profile-curve estimation based on surface data only. Hence, the approach in this section is to minimize a cost function which is the sum of the costs functions discussed in the aforementioned sections.

Denote (all) the available geometric data by $\Theta_{i,j}$, the break-curve data by $\Phi_{i,j}$ and the surface data by $\Omega_{i,j}$, this for sherds i and j. Then alignment and pot model estimation for sherd i and j data sets jointly, is done by the minimization over all geometric parameters, *i.e.*:

$$\begin{aligned} \min\left[-\log P\left\{\Theta_{i,j} \mid \beta_{ij}, \mathbf{l}_{ij}, \mathbf{T}_{ij}, \alpha_{ij}\right\}\right] &= \\ \min\left[-\log P\left\{\Phi_{i,j} \mid \mathbf{T}_{ij}\right\} - \log P\left\{\Omega_{i,j} \mid \mathbf{T}_{ij}, \mathbf{l}_{ij}, \alpha_{ij}\right\}\right] &= \\ e_{\mathbf{T}} + e_{grad} + \text{constant} , & \end{aligned}$$

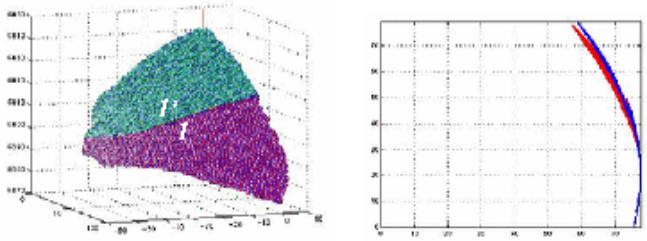
where the two energies, $e_{\mathbf{T}}$ and e_{grad}, are defined in equations (4) and (10), respectively.

The implemented algorithm uses approximations for doing the minimization, is computationally fast, and is reasonably accurate for the few examples tried. Figures 9 and 10 are results for one such experiment. This algorithm is presently undergoing refinement.



(a) p11 (b) p12

Figure 9: Two of the 13 sherds comprising a broken pot. Some of these sherds also appear in Figure 7. Breaking a pot provided sherds that shared common boundaries. The 1 and 1' denote break-curves that are common to the pair of sherds.



(a) (b)

Figure 10: Estimated joint geometry. The sherd data fits together well, (a), the projected-data scatter about the estimated profile curve is small, (b).

4 Conclusion

A Bayesian approach has been outlined for the estimation of mathematical representations for pots based on sherds found at archaeology sites. The key algorithms for implementing the approach have been developed, and experimental results from applying these algorithms to real sherd 3D data have been presented and discussed. At this time, experiments have been run on automatic matching and aligning pairs and triples of sherds.

Work remaining in this project is basically to speed-up some of the algorithms by carrying along and using extracted geometric shape information and its covariances, rather than using the sherd raw measurement-data each time two groups of sherds are aligned to form a possible larger group of assembled sherds toward constructing a virtual pot. The virtual-pot estimation is realized as a bottom-up merging of sherds through minimization of a cost function for aligned groups of sherds. This search algorithm is also in a state of refinement. If the pot has a spout and/or handles, the associated sherds must be included in the assembly.

The framework discussed in this paper is for estimating arbitrary a priori unknown axially-symmetric pot models. Hence, it is *unsupervised* pot geometry-learning from sherd data. If instead we know *a priori* that the pot sherds present are not arbitrary but rather that each belongs to one of a group of 10 known pot shapes, the problem is computationally much easier because the sherd alignment problem is then more of a pot shape-recognition problem and less of a shape-estimation problem.

The framework presented can accommodate additional geometric and pattern information which should result in doing the pot estimation faster, or with fewer sherds, or estimating models for more complex ceramic objects.

References

[1] Shapegrabber inc. http://www.shapegrabber.com.

[2] F. Bernardini et al. A digital model of Michelangelo's Florentine Pietà. In *4th Int.l Conf. on Cultural Heritage Networks Hypermedia*, Milan, Italy, September 1999. www.research.ibm.com/pieta/pieta_refs.htm.

[3] M. Blane et al. The 3L algorithm for fitting implicit polynomial curves and surfaces to data. *IEEE Trans. on Pattern Anal. Machine Intell.*, 22(3):298–313, March 2000.

[4] R. Bolle and D. B. Cooper. On optimally combining pieces of information, with application to estimating 3-D complex-object position from range data. *IEEE Trans.*

on Pattern Anal. Machine Intell., 8(5):619–638, September 1986.

[5] N. Griffiths, A. Jenner, and C. Wilson. *Drawing Archaeological Finds: A Handbook*. Occasional Paper 13. Institute of Archaeology, University College London, London, UK, 1990.

[6] M. S. Joukowsky. *A Complete Manual of Field Archaeology*. Prentice-Hall, Englewood Clifs, NJ, USA, 1980. 630 pages.

[7] M. S. Joukowsky. *Petra Great Temple - Volume 1: Brown University Excavations 1993-1997*. Petra Exploration Fund, Providence, RI, USA, 1999. 390 pages.

[8] M. Kampel and R. Sablatnig. Color classification of archaeological fragments. In *Proc. of the 15th International Conference on Pattern Recognition*, volume 4, pages 771–774, Barcelona, Spain, 2000.

[9] W. Kong and B. B. Kimia. On solving 2D and 3D puzzles using curve matching. In *Proc. of CVPR*, Hawaii, USA, December 2001. IEEE, Computer Society.

[10] M. Levoy et al. The digital Michelangelo project: 3D scanning of large statues. In *Proc. Siggraph 2000*, pages 131–144. ACM, July 2000.

[11] F. F. Leymarie et al. The SHAPE Lab: New technology and software for archaeologists. In *CAA 2000: Computing Archaeology for Understanding the Past*, BAR International Series no. 931, pages 79–90. Archaeopress, Oxford, UK, 2001.

[12] V. Nautiyal, S. Nautiyal, and M. Naithani. Optical plotting and AutoCAD for drawing pottery. *CSA Newsletter*, XII(3), Winter 2000. www.csanet.org/newsletter/winter00/nlw0003.html.

[13] G. Papaioannou, E.-A. Karabassi, and T. Theoharis. Virtual archaeologist: Assembling the past. *IEEE Computer Graphics and Applications*, 21(2):53–59, March/April 2001.

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

[15] D. J. Struik. *Lectures on Classical Differential Geometry*, chapter Elementary Theory of Surfaces. Dover, NY, 1950.

[16] T. Tasdizen, J. P. Tarel, and D. B. Cooper. Improving the stability of algebraic curves for applications. *IEEE Trans. on Image Proc.*, 9(3):405–416, March 2000.

[17] G. Taubin. Estimation of planar curves, surfaces and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Trans. on Pattern Anal. Machine Intell.*, 13(11):1115–1138, November 1991.

[18] S. Tosovic and R. Sablatnig. 3D modeling of archaeological vessels using shape from silhouette. In *Third International Conference on 3D Digital Imaging and Modeling*, Quebec, Canada, June 2001.

[19] G. Ucoluk and I. H. Toroslu. Automatic reconstruction of broken 3-D surface objects. *Computers and Graphics*, 23(4):573–582, August 1999.

[20] A. Willis et al. Extracting axially symmetric 3D geometry from limited 3D range data. Technical Report SHAPE-TR-2001-01, SHAPE Lab., Brown University, Providence, RI, 2001. http://www.lems.brown.edu/vision/publications/.

[21] A. Willis et al. The quantitative performances of three algorithms for estimating axis/profile curves of pottery sherds. Technical Report SHAPE-TR-2001-02, SHAPE Lab., Brown University, Providence, RI, 2001.