

PRIP-TR-83

July 30, 2003

Automated Profile Extraction of Archaeological Fragments ¹

Hubert Mara

Abstract

Thousands of fragments of ceramics are found at archaeological excavation sites. Till today archaeologists have drawn and classified them manually. This method is very time consuming and classification depends on the experiences of the archaeologists.

Therefore we developed a system that speeds up this process by using a 3D-scanner for the acquisition and a software that generates a registered 3D-model of the sherd. The features for classification used by archaeologists and a 3D reconstruction of the unbroken vessel are estimated automatically. The registration of different views of the sherd are based on the estimation of the rotational axis by a Hough inspired method. The classification and reconstruction is done by extraction of the longest profile line, which is an intersection of the sherd along the rotational axis of the unbroken vessel. The extracted features for classification are diameters, heights and their relation to each other. These features are estimated by the use of extremal points of the profile line.

Results of the system developed are presented for both synthetic and real input data.

¹This work was partly supported by the Austrian Science Foundation (FWF) under grant P13385-INF, the European Union under grant IST-1999-20273 and the Austrian Federal Ministry of Education, Science and Culture.

1 Introduction

Motivated by the requirements of today's archaeology, we are developing an automated system for storage, reconstruction and classification of ceramics. Ceramics are among of the most widespread archaeological finds, have been used for a short period of time for classification purposes. Since the 19th century, the physical characteristics of archaeological pottery have been used to assess cultural groups, population movements, inter-regional contacts, production contexts, and technical or functional constraints (archaeometry [12]). Because archaeometry of pottery still suffers from a lack of methodology, it is important to develop analytical classification tools of artifacts [16]. A large number of ceramic fragments, called sherds, are found at every excavation site (Figure 1).



Figure 1: Boxes filled with ceramics stored in archives.

These fragments are documented by being photographed, measured, and drawn; then they are classified. The purpose of *classification* is to get a systematic view on the excavation finds.

Traditional archaeological classification is based on the so-called profile of the object, which is the cross-section of the fragment in the direction of the rotational axis of symmetry. This two-dimensional plot holds all the information needed to perform archaeological research. The correct profile and the correct axis of rotation are thus essential to reconstruct and classify archaeological ceramics.

The sample of classification rules [21] shown in Table 1 were applied at the late Roman burnished ware of Carnuntum [9].

Classification is applied on unbroken vessels - further called objects - or on fragments of objects called sherds. Not all sherds can be used for classification, because some of them are too small or do not contain robust features. So we start with classification of fragments that contain a part of the rim, which is the top level part of the object and a part of the orifice.

The classification of basic vessel form by archaeologists consists of several measurements [6]:

basic vessel form	characteristic ratio	rim-diameter (cm)	wall-diameter (cm)
plate	1:8	16-34	-
bowl	1:2 - 1:4	10-16 12-30	- -
beaker	1:1	8-11	7-14
jug	4:1 - 2:1	6-14	(no references)
pot	1:1 - 3:1	8-12 12-16	15-25 18-21

Table 1: Classification of basic vessel types.

- Rim-diameter: The diameter of the orifice plane.
- Wall-diameter: The maximum diameter of the object orthogonal to its rotational axis.
- Height: The overall height of the object.
- The characteristic ratio: The ratio between height and rim-diameter.

Archaeologists use the number of extremal (e.g. minimum, maximum, inflection) points of the curvature of the profile line and their distance ratios to determine which type of vessel a sherd belongs to [16]. A detailed set of rules for the extraction of extremal points used by archaeologists is shown in [13].

There are already two computer systems for analysis of sherds. These systems are called ARCOS (ARchaeological COmputer System) [10] and SAMOS (Statistical Analysis of Mathematical Object Structures) [22]. These two systems are half-automated and so these systems require user interaction as described in [20]. These systems have a resolution of approximately 1-2 *mm*. As today's computers can process large amounts of data generated by 3D-scanners that have a resolution of 0.1 *mm*, we were motivated to develop a fully automated and portable system that requires a minimum of user interaction using today's 3D-scanning hardware.

There is also an interactive system for 3D-reconstruction of vessels [14] available, which could be used to pre-process the sherds before extracting the profile line for archaeological classification.

Another approach for extraction of the profile line is to draw it on a piece of paper by hand or by using a mechanical tool called profilograph. This drawing of the profile line is scanned using a 2D-scanner and classified using scale space methods as shown in [7].

Section 2 describes the theory used to develop the system. The algorithm and the matlab-scripts used for the data processing are described in Section 3. The results of the experiments with real input data is discussed in Section 4 and a short outlook for future work is given in Section 5. A table of mathematical symbols used in our algorithm is shown in Appendix A. A list of input-values and output-values of the matlab-scripts is shown in Appendix B and C. An example of a VRML-file is shown in Appendix D.

2 Theory

This section describes the mathematical fundamentals (e.g. orientation, registration and estimation of the profile line) of our system. The algorithms used are inspired by the methods used by archaeologists. In this section these methods and the archaeological terms are explained.

First the data acquisition (Section 2.1) and the data format (Section 2.2) for storage are described. Then the profile line (Section 2.3), which is extracted by our system for classification is defined. Afterwards the orientation (Section 2.1) and registration of the acquired data of the sherds based on the rotational axis (Section 2.4) is explained. In the last sections the representations of the coordinate-system and the transformations of the coordinate system for data processing (Section 2.5), reconstruction and classification (Section 2.6) of a larger part (in respect to the size of the fragment) of the original vessel is shown.

2.1 Data Acquisition

In earlier work [1] we used different types of scanning technologies, which directly acquired the profile line (see Section 2.3), but this system had different drawbacks for use with a portable, easy-to-use and automated system. So we chose the *Eyetrronics ShapeSnatcher Technology* [3] (see Figure 2), because it is portable and can be operated without expert knowledge. It is based on a CCD-camera and a flashlight to acquire the shape from a structured (well defined) light pattern. The image, together with the knowledge about the pattern and its relative position to the camera, are used to calculate the coordinates of points belonging to the surface of the object [11]. Since the 3D-scanner can only capture one side of the sherd per scan, so the inner side and the outer side has to be scanned separately to gather a complete 3D-model of the sherd.



Figure 2: Eyetronic's ShapeCam consisting of a CCD-camera (left) and a slide-projector acting as flashlight mounted on the top-right part of the handling frame.

2.2 3D Data - VRML

The 3D-data acquired by the 3D-scanner is stored as 3D-surface, which consists of 3D-points (vertices) that are connected in form of triangles (called faces). The 3D-model contains the color information (called texture) for each face for visualization. These vertices and faces are stored in an indexed list (\mathcal{L}).

There are different types of file formats for storing 3D-data (e.g. AutoCAD, Wavefront, OpenInventor). We have chosen VRML which is software independent and can be viewed with a web-browser with a VRML-plugin, which is free of charge.

A sample of the indexed list of vertices \mathcal{L}_v (Figure 3 upper-left) and of an indexed list of faces \mathcal{L}_f (Figure 3 lower-left) from a VRML-file defining a cube with edge length 1 is shown in Figure 3 on the right-hand side. As faces f can consist of three or more points, the end of the list of points belonging to one face is tagged with -1. The VRML-source including all required tags for display with a web-browser and additional color-information is shown in Appendix D.

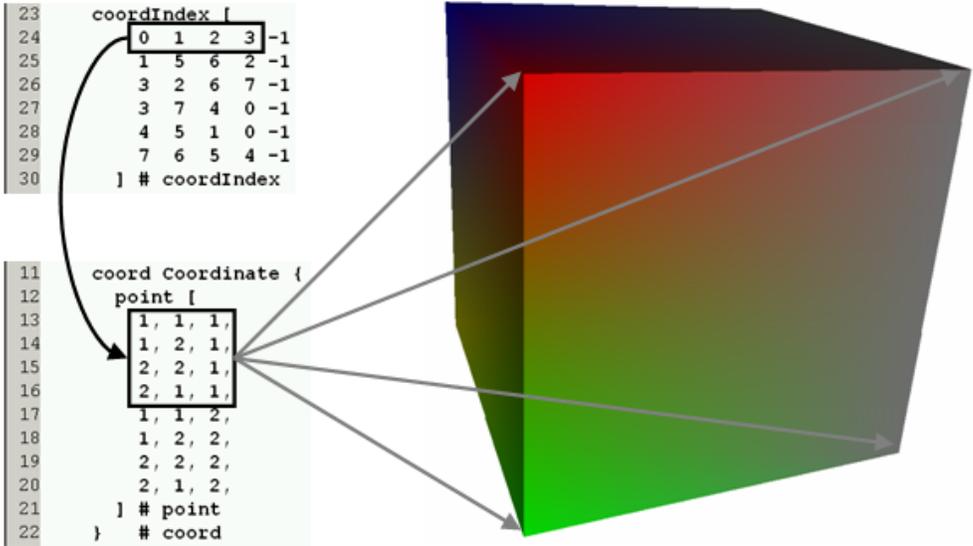


Figure 3: Upper left: Vertices \mathcal{L}_v , Lower left: Faces \mathcal{L}_f , Right: Visualization

In this example (Figure 3) every face consists of four indices to the corresponding vertices and for every face the color of its texture is stored as RGB-value. This kind of representation has been selected to export and import the data as a VRML-file [15].

2.3 Profile line

The profile line, which is used by archaeologists for classification and reconstruction, is defined below and shown in Figure 4. It is automatically extracted by our system from the 3D-model of the sherd consisting of a 3D-scan of the inner half and the outer half of the sherd.

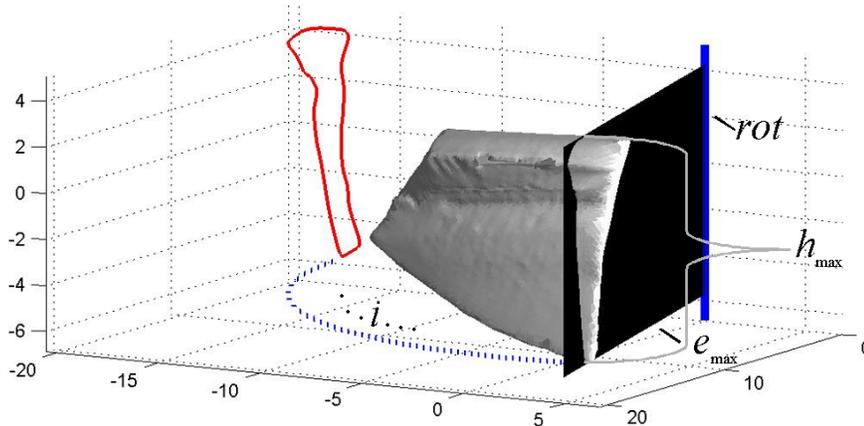


Figure 4: Oriented sherd, rotational axis rot , intersecting plane e_i , longest profile line $profile_{max}$, h_{max}

- A profile line (*profile*) is the crosscutting of the 3D-model of the sherd (*sherd*) and an intersecting plane e_i . This intersecting plane e_i , is defined by rot (see Section 2.4) and the direction i , so that e_i intersects the *sherd*.
- The intersection at the direction i_{max} , where the sherd has the maximum height $h_{max} = \max(h_i)$ is the profile line with the longest arc length and is called longest profile line ($profile_{max}$).
- The height h_i is defined as distance between two points of the surface of the sherd parallel to the rotational axis rot .
- The maximum height h_{max} is also used for classification by archaeologists.

2.4 Orientation and Registration by using the Rotational Axis

Ancient pottery was manufactured on rotational plates. So every object and thus every fragment of an object has a rotational axis rot . This axis rot is used by archaeologists to orientate the sherds. Based on this orientation the profile line is extracted (see Section 2.3). The rotational axis rot also leads to the exact position of a fragment on the original vessel.

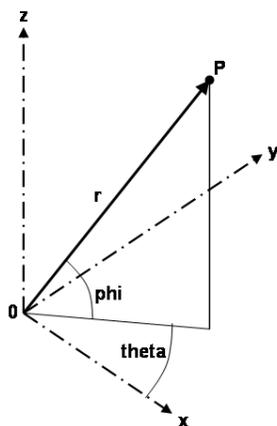
The two 3D-views of the sherd have to be registered [19] to estimate a complete 3D-model of the sherd. Therefore the rotational axis $rot_I N$ of the inner side and $rot_O UT$ of the outer side is estimated. The rotational axes $rot_I N$ and $rot_O UT$ are estimated by a hough-inspired method [23]. For registration the coordinate system of both sides have to be rotated and shifted about the rotational axis, so that the distances between the balance points of both views is minimized. The result is registered and oriented 3D -model of the sherd.

For orientation, registration and further data processing, two different types of representation of the coordinate system are used. For translation and rotation we use the

cartesian representation of the coordinate system, because in this representation the translation is done by an addition, which requires a small fraction of computing time compared to translation using a transformation matrix. To estimate the two angles (azimuth and elevation) required for rotation we use the spherical representation of coordinate system. Points \mathbf{p} and vectors \mathbf{v} in \mathbb{R}^3 using the spheric representation are described by the azimuth θ , elevation ϕ and the distance r to the origin point in the cartesian coordinate system (see Figure 5). This representation has been chosen for rotation, because rotation can be done addition.

Cartesian representation: $\mathbf{p}=(x, y, z)^T$

Spheric representation: $\mathbf{p}=(\theta, \phi, r)^T$



(a)

$$\theta = \text{atan}(x, y)$$

$$\phi = \text{atan}(z, \sqrt{x^2 + y^2})$$

$$r = \sqrt{x^2 + y^2 + z^2}$$

(b)

Figure 5: (a) Spherical coordinate system (b) conversion from cartesian to spheric coordinate system

2.5 Data processing

For the estimation of the profile line the inner and outer view has to be registered and the sherd has to be orientated. To register, orientate and reconstruct the inner and outer view of the sherd, the 3D-model has to be transformed by rotation (see Equations 1, 2 and 3) and translation (see Equation 4).

- Rotation of angle α about the x -axis:

$$\mathbf{R}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & \sin(\alpha) \\ 0 & -\sin(\alpha) & \cos(\alpha) \end{pmatrix} \quad (1)$$

- Rotation of angle β about the y -axis:

$$\mathbf{R}_y(\beta) = \begin{pmatrix} \cos(\beta) & 0 & -\sin(\beta) \\ 0 & 1 & 0 \\ \sin(\beta) & 0 & \cos(\beta) \end{pmatrix} \quad (2)$$

- Rotation of angle γ about the z -axis:

$$\mathbf{R}_z(\gamma) = \begin{pmatrix} \cos(\gamma) & \sin(\gamma) & 0 \\ -\sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3)$$

- Translation in x , y and z direction:

$$\mathcal{L}' = \mathcal{L} + (x, y, z)^T \quad (4)$$

For the extraction of the profile *profile* all vertices that are connected by an edge of the face intersecting the plane e_i are selected. Therefore the Hessian normal form $ax^2 + by^2 + cz^2 + d = 0$ is used to determine the distance d to the plane. The sign of the distance and the relative distance in respect to the maximum distance is used to reduce the number of vertices for estimation of the profile line. Vertices with a $d < 0$ are located on the left side of the plane. Vertices with $d > 0$ are located on the right side. Every face consists of three or more vertices, where each pair of vertices describe an edge of the face. The pairs of vertices, with vertices on different sides of the intersecting plane e_i are used to estimate the points of crosssections of the edges and the intersecting plane e_i , because the vertices of the 3D-model need not be located on the e_i . The result of connection of the points of crosssection is the profile line.

2.6 Reconstruction and Classification

For reconstruction the vertices $\mathbf{p}_{profile_{max}}$ of the longest profile line are copied n times. The smoothness of the display of the reconstructed 3D-model depends on n . Larger n means smoother, but slower to display. Experiments have shown that $180 < n < 360$ is the best trade off between smoothness and time to display the object. Each copy \mathbf{p}_n of the vertices of the longest profile is rotated, using $\mathbf{R}_z(n*\pi/180)$. The faces of the reconstructed object are estimated by connecting each point \mathbf{p}_m with its neighbours to rectangular mesh. Let k be the number of vertices of the profile line, than the indices of the neighbours of \mathbf{p}_m are: $m + 1$, $m + k$ and $m + k + 1$.

The second purpose for extraction of the profile line is classification, which is done by the rules defined by archaeologists. The features for this rules are estimated using the profile line (e.g. minimum and maximum diameter). A sample set of rules and how they are estimated is shown in [13];

3 Data processing & Matlab Functions

This Section describes the details of the algorithm for an automated registration and profile extraction of archaeological fragments. The subsections of the matlab-scripts are explained step by step. The algorithm begins with reading the 3D data from an VRML-file, estimation of the rotational axes of the inner and outer half, registration of both halves, extraction of the longest profile line, validation of the estimation of the rotational axis

rot and finish with the extraction of the extremal points of the profile line and the partial reconstruction of the original object.

The pottery data set we use for experiments consists out of the following recorded objects:

	BOX 1	BOX 2	BOX 3	BOX 4
Number of pieces	21	12	28	9
VRML Version	2		1	2
Mesh type	quadrangles			triangles
Nr. of views	2			Multiple
Vertices per view	4.000-9.000		3.000-8.000	15.000-80.000
Preregistered	No			Yes

Table 2: MURALE Pottery dataset.

3.1 Reading the 3D Data - `read_vrml`, `load_data`

Matlab does not provide support for reading VRML-files, so we had to develop our own script based on the work from [2]. As we do not need all of the functionality described in the VRML-standard, the implementation of `read_vrml` can only import single objects stored as polygonal geometrical objects [15].

Because VRML is a markup language the first step is to search for the starting tag of the vertices-section. The next step is to sequentially read the lines afterwards, which contain the vertices $\mathbf{p}_i = (x_i, y_i, z_i)^T$ until the closing tag is detected. x_i, y_i and z_i are the coordinates of the vertices of the surface of the object in centimeter. The result of this procedure is an indexed list of n vertices:

$$\mathcal{L}_v = \{\mathbf{p}_1, \dots, \mathbf{p}_i, \dots, \mathbf{p}_j, \dots, \mathbf{p}_k, \dots, \mathbf{p}_n\} \quad (5)$$

After reading the vertices the same procedure is applied to read the list of faces:

$$\mathcal{L}_f = \{f_1, \dots, f_m\}, f_k = (i, j, k) \quad (6)$$

Optionally the normal vectors $\mathcal{L}_n = \{\mathbf{n}_1, \dots, \mathbf{n}_n\}$ and the color information $\mathcal{L}_c = \{c_1, \dots, c_n\}$, $c_i = (red, green, blue)$ is read, which is not necessary for further calculation, but it is used for visualization purposes. The normal vectors \mathcal{L}_n are read if present, because they are used for estimation of the rotational axis *rot*. The execution time for reading previously calculated normale is shorter than calculating them again. These normal vectors n_i are mean values of normal vectors of faces to which the vertex \mathbf{p}_i belongs to. If they are not stored in the file they have to be calculated in `calcaxis` (Section 3.2) for the estimation of the rotational axis.

So the 3D-model of the view of sherd is described by an indexed list of vertices, faces, normal vectors and color information: $sherd_{view} = \{\mathcal{L}_v, \mathcal{L}_f, \mathcal{L}_n, \mathcal{L}_c\}$ The list of \mathcal{L}_n can be $\{\}$ and is estimated by the function described in the next section. \mathcal{L}_c can be $\{\}$.

`read_vrml` is used two times for the inner and outer view ($sherd_i$, $i = \{1, 2\}$) of the sherd, because every view is acquired individually. As importing VRML-files requires 1 to 20 minutes depending on the size of the sherd, the imported 3D-data $sherd_{view}$ is stored as matlab-file, which can be re-read for testing and experimenting by using `load_data` in less than 10 seconds.

3.2 Orientation and Registration using the Rotational Axis

This section describes the functions `calcaxis` for estimation of the rotational axis and `orientate_object` for orientation and registration of the inner and outer view.

The function `calcaxis` [17] requires the list of vertices \mathcal{L}_v and faces \mathcal{L}_f and optionally the list of previously estimated (by the 3D-scanner software) normal vectors \mathcal{L}_n as input, to reduce the execution time. If \mathcal{L}_n is $\{\}$, it is estimated. First the type of the sherd view $sherd_i$ is determined. A concave view is tagged as outer view $sherd_{out}$ and a convex view is tagged as inner view $sherd_{in}$. Afterwards the size of the data-set ($\mathcal{L}_v, \mathcal{L}_n$) is reduced to lower the execution time for the next step by selecting a sub-set of \mathcal{L}_v and \mathcal{L}_n (see [17] for details). Then the data-set is transformed into the Hough-space [8]. Using the Principal Component Analysis (PCA) [5] on the rastered Hough-Space returns the rotational axis rot , which is represented as point vector \mathbf{p}_{rot} and direction vector \mathbf{v}_{rot} .

`calcaxis` is applied on both views $sherd_i$ of the sherd. The result are two rotational axis $rot_{in} = \{\mathbf{p}_{in}, \mathbf{v}_{out}\}$ for the inner view and $rot_{out} = \{\mathbf{p}_{out}, \mathbf{v}_{out}\}$ for the outer view.

$$sherd_{in} = \{\mathcal{L}_v, \mathcal{L}_f, \mathcal{L}_n, rot_{in}\} \quad (7)$$

$$sherd_{out} = \{\mathcal{L}_v, \mathcal{L}_f, \mathcal{L}_n, rot_{out}\} \quad (8)$$

The proper orientation of a sherd and the registration of the inner and outer view is done in two steps per view using the rotational axis rot_{in} and rot_{out} (see Section 2.4).

For orientation 3D-model has to be transformed, so that its rotational axis is positioned at the point of origin (first step - Equation 9) and aligned along the z -axis (second step). Following these orientation steps there are only two degrees of freedom to move the sherd so that the orientation will not be affected. The views can only be shifted along the z -axis and rotated about the z -axis. So the registration (step four - Equation 12) of the orientated views is done by rotation and translation, so that the two points of balance (one per view) is aligned along the x -axis. The parameter for the translation is the negative z -coordinate z_b of the balance point \mathbf{b} . The angle (γ_b) for the rotation is estimated by transforming the cartesian coordinates into spheric coordinates (step three - Equation 10,11).

- In the first orientation step \mathcal{L}_v of $sherd_{in}$ is shifted, so that \mathbf{p}_{in} shifts into the origin:

$$rot_{in}' = \{(0, 0, 0)^T, \mathbf{v}_{in}\} \implies \mathcal{L}'_v = \mathcal{L}_v - \mathbf{p}_{in} \implies sherd_{in}' = \{\mathcal{L}'_v, \mathcal{L}_f, \mathcal{L}_n, rot_{in}'\} \quad (9)$$

- In the second orientation step \mathcal{L}'_v is rotated about the z -axis and the x -axis, so that the rotational axis $rot_{in}'' = \{(0, 0, 0)^T, (0, 0, 1)^T\}$ is identical to the z -axis. This is done to simplify the estimation of diameters and to improve the performance for rotations the rotational axis.

- Spheric representation of the direction of the rotational axis:

$$\mathbf{v}_{in} = (\theta_{rot}, \phi_{rot}, r_{rot})^T \implies \gamma = -\theta_{rot}, \beta = \frac{\pi}{2} - \phi_{rot}, \alpha = 0 \quad (10)$$

$$\mathcal{L}''_v = \mathcal{L}'_v * \mathbf{R}_z(\gamma) * \mathbf{R}_y(\beta) * \mathbf{R}_x(\alpha) \implies \mathit{sherd}_{in_{orientated}} = \{\mathcal{L}''_v, \mathcal{L}_f, \mathcal{L}_n, \mathit{rot}_{in}''\} \quad (11)$$

- In the registration step the balance point $\mathbf{b} = \mathit{mean}(\mathcal{L}''_v) = (x_b, y_b, z_b)^T = (\gamma_b, \phi_b, r_b)^T$ is estimated. Then the $\mathit{sherd}_{in_{orientated}}$ is shifted, so that $\mathbf{b}' = (0, y'_b, 0)^T$.

$$\mathcal{L}'''_v = (\mathcal{L}''_v - (0, 0, z_b)^T) * \mathbf{R}_z(-\phi_b) \implies \mathit{sherd}_{in_{OR}} = \{\mathcal{L}'''_v, \mathcal{L}_f, \mathcal{L}_n, \mathit{rot}_{in}'''\} \quad (12)$$

- These two step are repeated for sherd_{out} to estimate the orientated outer view $\mathit{sherd}_{out_{OR}}$.

The rotational axis does not define which side of a view is upside or downside. It is possible that one view is registered upside down compared to the other view. This will be done automatically in a future version of the algorithm by comparing the outlines of the inner and outer view, currently it has to be checked manually. For the manual check the outlines of the registered inner and outer view is drawn. If these outlines match the orientation is correct. If they do not match one view has to be rotated around the x -axis by using $\mathbf{R}_x(\pi)$.

The result is an orientated and registered 3D-model sherd_{OR} of the sherd:

$$\mathit{sherd}_{OR} = \{\mathit{sherd}_{in_{OR}}, \mathit{sherd}_{out_{OR}}\} \quad (13)$$

3.3 Longest profile line - longest_profile

For classification and reconstruction only the longest profile line is used. In this procedure multiple profile lines are extracted and the longest is selected for further processing. All of the profile lines are used to evaluate the estimation of the rotational axis and the registration of the inner and outer view.

The number of profiles, which are extracted depends on the resolution of the 3D-scanner and the size of the sherd. Experiments have shown that 12 extracted profiles have the best ratio between performance and exactness. In Figure 6a four of the multiple intersecting planes e_i are shown as light gray rectangles crossing at the rotational axis (vertical black line throught the point of origin). The gray object is the 3D-model of the sherd and the black lines around this model are the intersection between the sherd and the planes e_i , which are the multiple profile lines $\mathit{profile}_i$.

Instead of estimating the distance d for each vertex to an intersecting plane e_i using the Hessian normal form, the sherd is rotated so that the intersecting plane is the xz -plane. Afterwards the y -coordinate is the distance d to the intersecting plane. Experiments have shown that the rotation is ten times faster than using the Hessian normal form, because a single matrix multiplication for the rotation is done faster than a loop of multiplications and additions with matlab.

$$\gamma_i = \frac{\max(\mathcal{L}_v(\theta)) - \min(\mathcal{L}_v(\theta))}{n}, \text{ i.e. } n = 12 \quad (14)$$

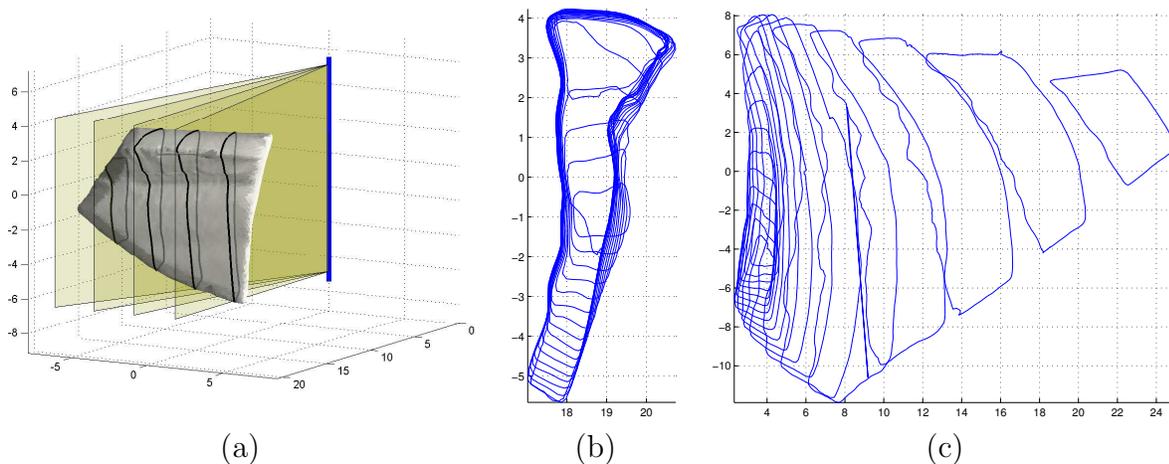


Figure 6: (a) Sample of intersecting planes e_i (b) Multiple profile lines using a correct estimated rotational axis (c) Multiple profile lines using a wrong rotational axis

So the sherd has to be rotated 12 times by γ_i . As the rotational axis is identical to the z -axis the rotation is done by using $\mathbf{R}_z(\gamma)$. This rotation positions the sherd so that the intersecting plane e_i is the xz -plane. So the distance for every vertex at every intersection is equal to the y -coordinate of each vertex. After every rotation a profile line is extracted by `extract_profile` (see Section 3.4) and applied to $sherd_{OR}$. Afterwards the arc length is estimated and the profile with the longest arc length is selected.

Evaluation: For evaluation of the estimation of the rotational axis the mean diameter for each profile is estimated. If the standard deviation of these mean diameters exceeds a certain threshold given by archaeologists (i.e. $0.5cm$) the estimation of the rotational axis was not correct. The extracted multiple profiles based on a correct estimated rotational axis are shown in Figure 6b. In Figure 6c the rotational axis was not estimated correct, because the fragment included a flat area from the bottom plane.

For evaluation of the registration the minimum and maximum diameter of each profile is estimated. The difference between the minimum and maximum diameter is the thickness of the sherd. As the range of thickness of a sherd is known a priori lower threshold (e.g. $0.5cm$) and an upper threshold (e.g. $2cm$) can be set. The thickness is defined by the manufacturing process by concerning the stability of a vessel. If the thickness is between these two thresholds this indicates, that the registration was done correctly. The diameters and the standard deviation of the mean diameter of the profiles from Figure 6b and Figure 6c are shown in Figure 7a and Figure 7b. The x -axis in Figure 7 shows the elevation θ in degree and the y -axis the z -normalized diameter in centimeter.

3.4 Extraction of a Profile Line - `extract_profile`

`extract_profile` extracts the profile of the sherd, where it is intersected by the xz -plane.

- First the relative distance (shown in Figure 8) in respect to the maximum distance between the vertices and the xz -plane is estimated. To speedup the estimation of the

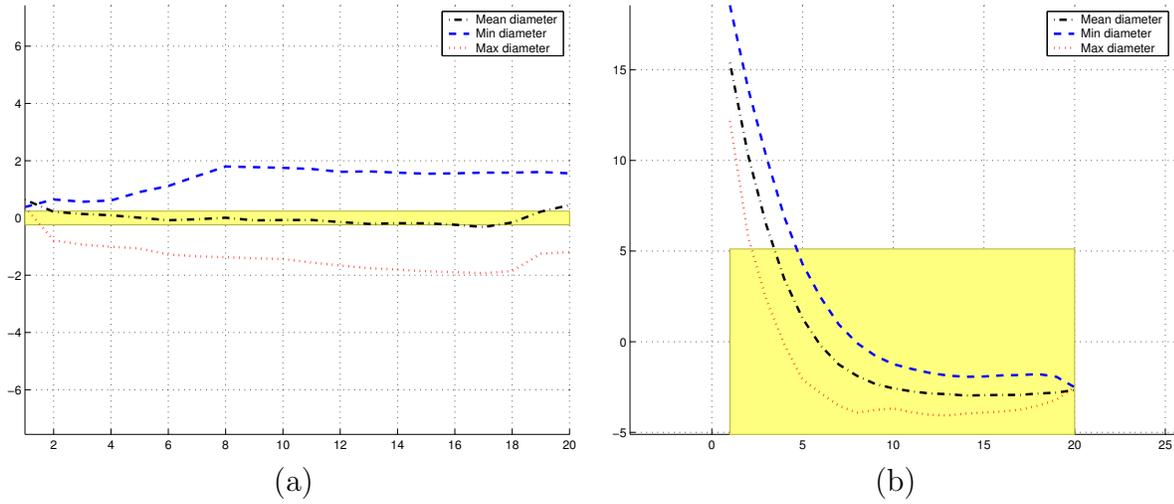


Figure 7: Maximum, mean and minimum diameter.

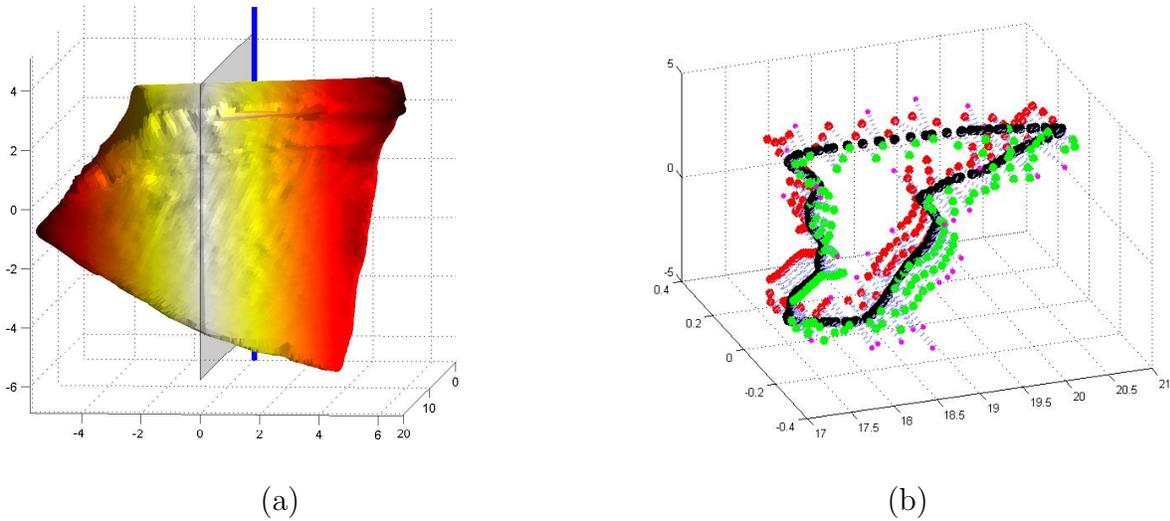


Figure 8: (a) Oriented sherd and intersecting plane e_i . The level of gray corresponds to the distances. Lighter means nearer to the intersecting plane. (b) Vertices $\mathbf{p}_{d<0.01}$ left (light gray) and right (dark gray) of the intersecting plane, edges $edge_{intersect}$, vertices of the profile line $\mathbf{p}^{intersect}$ (black).

profile line only the nearest 1% vertices $\mathbf{p}_{d<0.01}$ are selected for further processing. Experiments have shown that vertices with a distance larger than 1% can not be used for the estimation of the profile line. This threshold depends on the resolution of the 3D-scanner and can be adjusted, when another resolution is used.

- In the next step the faces $f_{d<0.01}$ which contain the indices of the nearest vertices $\mathbf{p}_{d<0.01}$ are selected. These faces $f_{d<0.01}$ are split into edges, for estimation of the vertices of the profile line: A face $f = \{i, j, k\}$ connects the vertices $\mathbf{p}_{[i]}$, $\mathbf{p}_{[j]}$ and

$\mathbf{p}[k]$, so the edges of the face f are described by $edge = \{(i, j), (j, k), (k, i)\}$. Each of these vertices can have a positive or a negative sign, which corresponds to the position in respect to intersecting plane. A vertex with a negative sign is located on the left side and a vertex with a positive sign is located on the right side of the plane. So the edges $edge_{intersect}$, which intersect the plane e_i must contain one vertex with a negative sign and one with a positive sign. The points from those edges $edge_{intersect}$ $\mathbf{p}_{intersect}$ are selected and the parameters of the line describing these edges $edge_{intersect}$ are estimated. With these parameters the point of intersection $\mathbf{p}_{intersect}$ between the line and the intersecting plane (equal to the xz -plane) is estimated. These points $\mathbf{p}_{intersect}$ are connected with their nearest neighbor. This connected points $\mathbf{p}_{intersect}$ are the profile line *profile*.

- All vertices $\mathbf{p}_{intersect}$ of the profile have the distance $y = 0$ to the intersecting xz -plane. The x -coordinate is the distance (radius r) to the rotational axis *rot* and the z -coordinate is the height h .

$$profile = \{\dots, (x_i, 0, z_i)^T, \dots\} \implies profile' = \{\dots, (r_i, h_i)^T, \dots\} \quad (15)$$

Reconstruction is done by rotating the *profile* 360° about the rotational axis using $\mathbf{R}_z(\gamma)$.

$$object = \{profile * \mathbf{R}_z(\gamma_i)\}, i = 1 \dots n, alpha_1 = 0, \gamma_{i+1} = \gamma_i + \Delta\gamma, \Delta\gamma = 2 * \pi / n \quad (16)$$

Figure 9a and Figure 9b shows the reconstruction based on the longest profile of Figure 6b and Figure 6c. The reconstructed object can be stored with `export_vrml` as VRML-file.

4 Experiments and Results

Experiment were done on the 27 sets of 3D images of archaeological fragments given by archaeologists for experiments. Each set contained one image of the inner half and one of the outer half of the sherd. In 9 cases the estimation of the rotational axis returned a correct result. 10 results had three different types of minor errors:

- Flipped: Inner or outer half was upside down - This was expected, because the position of the rotational axis does not determine the upside or underside of one half.
- Twisted: One or both sides were slightly twisted (less than 10°).
- Thickness: The distance between the inner and outer half was too big (2 to 3 cm).

The second and third error has been observed on small or flat sherds. For 7 sets the estimation of the rotational axis did not have a correct result, because the sherds were too small, too flat, contained a handle or were part of a bottom fragment. All of these 7 sherds have normal vectors, which do not point at the rotational axis. So the hough-inspired [23] estimation of the rotational axis was not done correctly.

The success rate for correct extraction of the profile line and consequently the percentage of sherds, which is used for further classification is 50% of the sherds found at the excavation site. This has to be seen with respect to manual archivation done by archaeologists [16]: for coarse ware 35% [4] and for fine ware 50% [18] of the findings are used for further classification. It depends on the ratio between bending of the shape and the fragment and its diameter [19] (e.g. handle, flat fragments like bottom pieces, small size, etc.).

The execution time using a prototype written in matlab running on a *Pentium III* 1 GHz is less than a minute per sherd. The experiments have shown that a correct estimation of the rotational axis is required for further data processing and that this estimation uses 70% to 80% of the execution time for estimation of the longest profile line. Considering the performance related results for the extraction and segmentation of profile lines comparing it to the time used by archaeologists drawing a profile line by hand shows that the number of classification per day can be increased dramatically.

The estimation of the rotational axis will also be used to reconstruct whole objects from several sherds. A 3D reconstruction of a rim-fragment is shown in Figure 9. The gray part is the artificially created 3D -object and the dark part is the original fragment.

Table 3 shows samples of the results of Box 1 and Box 2. The box number and the piece number identify the fragment in the storage (see Figure 1). The radius is the estimated mean radius of the whole object. The standard deviation ($Std(Radius)$) of the radius was estimated along the circumference of the fragment. The wall thickness is the difference between the mean radius of the inner side and the outer side of the sherd. The fragment size (Fragm. size) is the percentage of the circumference of the sherd compared to the circumference of the whole object. The overlapping area (Overl.) is the ratio of the intersecting area between the inner and outer side compared to the aggregation of the area of the inner and outer side. The maximum height (h_{max}), rim radius (Rim-R.) and bottom radius (Bottom-R.) are listed. The "Type (check)" is set to "OK", when the sherd contains a convex and concav side and "wrong", when both sides have the same type.

Table 4 show all estimated features for classification of the successful registered and orientated sherds. These features are:

- rdm : Rim-diameter (cm) - The diameter at the highest point of the sherd.
- wdm : Wall-diameter (cm) - The maximum diameter of the sherd orthogonal to the rotational axis.
- bdm : Bottom-diameter (cm) - The diameter at the lowest point of the sherd.
- h_{max} : The overall height (cm) of the sherd.
- $crat$: The characteristic ratio $h : rdm$

The table in Figure 5 shows the result of box number 3. This box contain sherds with handles and sherds with a large curvature with respect to box 1 and box 2. The experiment has shown that the extraction of the profile line could be done on 5 out of 26

Box	Nr	Radius (<i>cm</i>)	Thickn. (<i>cm</i>)	<i>Std(R)</i> (<i>cm</i>)	Fragm. size (%)	Overl. (%)	h_{max} (<i>cm</i>)	Rim-R. (<i>cm</i>)	Bottom R. (<i>cm</i>)	Type (check)
1	02	18,47	5,2	0,02	3,93	63	5,97	18,34	18,5	OK
1	04	12,29	0,51	0,235	8,63	70	7,12	12,58	12,47	OK
1	05	12,76	10,02	0,215	9,08	19	6,24	12,36	12,97	wrong
1	06	37,74	22	0,01	1,80	41	5,28	37,71	37,75	OK
1	07	24,99	8,22	0,09	3,68	62	7,32	25,05	25,17	OK
1	08	15,97	0,39	0,035	3,35	67	5,63	15,89	16,04	OK
1	09	13,63	4,42	0,145	7,24	58	4,85	13,78	13,57	OK
1	11	20,15	3,19	0,05	4,06	76	4,33	19,94	20,26	OK
1	12	24,56	3,63	0,08	3,75	71	8,51	24,73	24,35	OK
1	13	33,05	19,33	0,135	2,57	35	6,13	33,18	32,76	wrong
1	14	21	4,29	0,105	5,34	70	6,35	21,34	21,37	OK
1	15	32,63	5,57	0,08	2,47	66	8,13	31,63	33,28	OK
1	16	15,43	1,75	0,09	4,29	66	6,68	15,34	15,45	OK
1	17	16,9	1,51	0,145	6,53	67	8,81	16,32	17,16	OK
1	18	16,03	1,75	0,025	4,54	70	4,68	16,17	16,05	OK
1	19	14,96	0,84	0,145	6,54	89	9,15	14,45	14,69	OK
1	20	13,77	1,56	0,08	6,00	63	7,09	13,39	13,66	OK
1	21	22,25	4,98	0,035	3,45	57	6,03	22,36	22,21	OK
1	22	11,9	0,75	0,1	9,35	57	6,04	11,4	11,93	OK
1	23	65	0,65	0,08	6,41	69	7,4	12,29	12,66	OK
1	24	27,02	11,98	0,025	3,27	56	6,41	27,03	27,03	OK
2	01	10,38	0,82	0,09	8,43	86	7,05	10,34	10,4	OK
2	02	16,51	1,13	0,025	4,79	64	8,94	16,29	16,45	OK
2	03	17,48	3,03	0,055	6,53	53	4,4	17,39	17,28	OK
2	04	7,49	1,67	0,18	11,49	68	6,9	6,27	8,69	OK
2	05	10,19	0,49	0,08	8,06	82	6,15	9,96	10,28	OK
2	06	3,05	1,03	0,205	24,92	16	10,15	3,42	3,5	OK
2	08	17,74	2,37	0,07	6,19	55	4,88	17,52	17,8	OK
2	09	12,79	0,97	0,08	8,50	75	6,96	12,33	12,88	OK
2	10	21,11	4,63	0,05	3,33	27	7,29	21,03	20,67	OK
2	10	18,83	1	0,06	4,74	33	7,36	18,85	18,86	OK
2	11	11,32	2,24	0,05	7,81	82	5,19	11,43	11,27	wrong
2	12	7,97	4,25	0,125	10,80	27	6,83	7,84	7,64	wrong

Table 3: Extracted features for the evaluation of the estimation of the rotational axis and the evaluation of the registration.

sherds of box 3. The sherds with a valid profile line from box 3 are number 6, 11, 16, 17 and 34.

Figure 9a displays a reconstructed pot (gray object) out of one fragment (dark object) based on the profile line (light line) and its axis of rotation (dashed line). Figure 9b shows a detailed part of the same object as Figure 9a.

Box	Nr	rdm	wdm	bdm	h_{max}	$crat$
1	04	25,16	75,61	24,94	7,12	0,33
1	08	31,78	50,16	32,08	5,63	0,63
1	16	30,68	49,32	30,9	6,68	0,62
1	17	32,64	42,31	34,32	8,81	0,77
1	18	32,34	31,06	32,1	4,68	1,04
1	19	28,9	33,80	29,38	9,15	0,85
1	20	26,78	30,15	27,32	7,09	0,89
1	22	22,8	27,62	23,86	6,04	0,83
1	23	24,58	44,61	25,32	7,4	0,55
2	01	20,68	54,22	20,8	7,05	0,38
2	02	32,58	35,02	32,9	8,94	0,93
2	04	12,54	35,66	17,38	6,9	0,35
2	05	19,92	25,79	20,56	6,15	0,77
2	06	6,84	22,85	7	10,15	0,30
2	09	24,66	19,05	25,76	6,96	1,29

Table 4: Results of proper registered and orientated sherds.

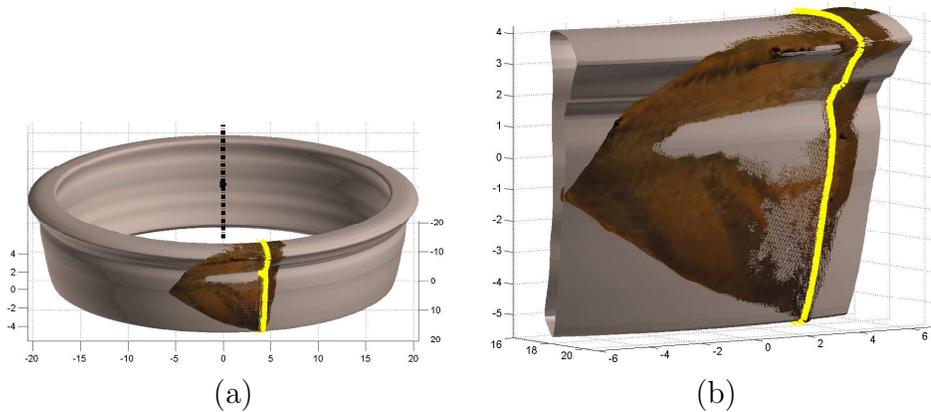


Figure 9: 3D reconstruction: (a) Reconstructed fragment and recorded fragment detected (b) Cross- sections of the original fragment along the perimeter of the virtual fragment.

5 Outlook

To improve the archaeometric statistics [12] based on the results of our system, the number of processed sherds will be increased in future work by using a robust method for estimation of the rotational axis, so that the registration can be applied on a larger percentage of sherds. The existing system can only identify fragments with handles and bottom fragments, when the estimation of the rotational axis fails. The estimation of the rotational axis has to be adapted, so that these fragments can be processed for profile extraction and registration.

Box	Nr	Radius (<i>cm</i>)	Thickn. (<i>cm</i>)	<i>Std(R)</i> (<i>cm</i>)	Fragm. size (%)	Overl. (%)	h_{max} (<i>cm</i>)	Rim-R. (<i>cm</i>)	Bottom R. (<i>cm</i>)	Type (check)
3	01	22,38	0,08	0,28	24,80	0,19	26,18	22,96	22,02	ok
3	02	14,33	3,78	0,09	14,74	0,78	5,51	16,22	16,34	ok
3	03	9,62	2,09	0,23	51,64	0,36	9,91	9,85	10,22	ok
3	04	13,35	7,77	0,03	12,75	0,48	6,62	17,11	17,21	ok
3	05	35,80	17,11	0,21	13,40	0,50	9,62	45,02	44,37	wrong
3	06	14,43	0,95	0,08	37,01	0,60	9,07	14,82	15,01	ok
3	08	0,26	0,60	1,98	188,28	0,04	8,18	1,69	2,24	wrong
3	09	11,25	1,46	0,31	27,23	0,52	10,23	11,64	12,28	wrong
3	10	15,46	2,01	0,52	32,96	0,75	4,35	16,69	16,73	ok
3	11	2,31	0,23	0,13	93,76	0,16	9,89	3,56	2,63	ok
3	12	7,12	0,56	0,23	74,46	0,39	5,39	8,39	7,42	ok
3	15	7,52	3,08	0,37	25,85	0,46	8,38	8,99	8,37	wrong
3	15	29,63	53,11	0,35	5,80	0,05	7,87	56,50	56,07	wrong
3	16	13,30	0,98	0,14	18,59	0,68	7,80	13,93	13,70	wrong
3	17	8,00	2,43	0,23	44,41	0,51	7,63	8,95	9,41	ok
3	18	13,80	10,22	0,10	23,02	0,44	8,48	18,07	19,09	ok
3	20	10,53	11,15	0,16	22,01	0,24	8,10	15,71	15,85	wrong
3	20	4,68	3,96	0,56	31,60	0,03	6,12	7,11	6,17	ok
3	21	15,57	0,00	0,04	21,95	1,00	6,60	15,31	15,61	wrong
3	22	23,97	6,39	0,22	20,72	0,67	7,27	27,21	27,18	ok
3	23	21,64	0,24	0,12	23,03	0,90	6,54	21,90	22,03	ok
3	24	13,20	0,63	0,33	38,94	0,81	12,28	12,46	13,17	ok
3	25	11,86	4,37	0,19	33,41	0,45	10,88	14,06	13,35	ok
3	26	23,66	5,93	0,06	10,52	0,76	5,90	26,53	26,61	ok
3	27	8,36	1,38	0,15	38,10	0,48	8,88	9,15	8,88	ok
3	28	5,13	3,43	0,11	39,25	0,32	6,54	6,21	6,64	ok

Table 5: Extracted features for the evaluation of the estimation of the rotational axis and the evaluation of the registration.

As the rotational axis leads to the position of the fragment in the unbroken vessel multiple fragments will be matched to reconstruct the whole object like archaeologists do manually (shown in Figure 10). Another approach is to match the profile lines and reconstruct the whole object by rotation of the matched profile line similar to Section 2.6.

The method has been tested on real data with reasonably good results. It is part of future work to increase the amount of processed sherds and to enhance the algorithm for registration of different sherds to acquire a whole object.

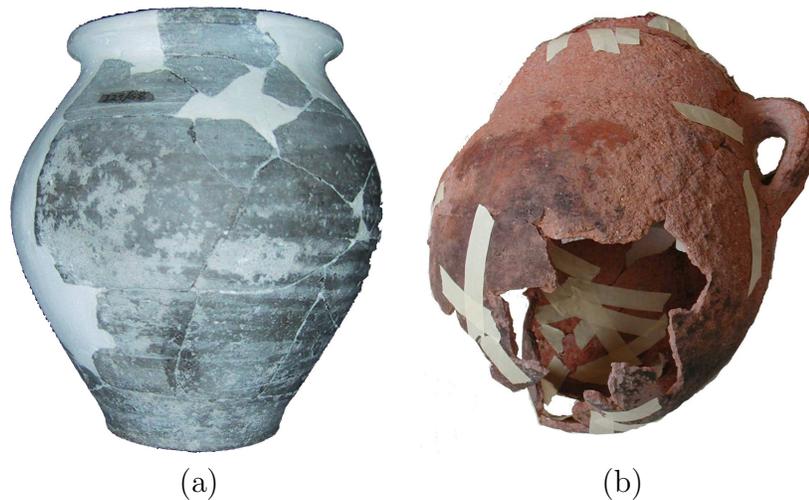


Figure 10: (a) complete manually reconstructed vessel (b) manually orientated sherds

References

- [1] K. Adler, M. Kampel, R. Kastler, M. Penz, R. Sablatnig, and K. Hlavackova-Schindler. Computer aided classification of ceramics - achievements and problems. In W. Börner and L. Dollhofer, editors, *Proc. of 6th Intl. Workshop on Archaeology and Computers*, pages 3–12, Vienna, Austria, 2001.
- [2] M.J.F. Cairns. *1st Year Report: An Investigation into the use of 3D Computer Graphics for Forensic Facial Reconstruction*. Glasgow University's Computing Science Department, 2001. <http://www.dcs.gla.ac.uk/mc/1stYearReport/Contents.htm>.
- [3] J. Cosmas, T. Itagaki, D. Green, E. Grabczewski, L. Van Gool, A. Zalesny, D. Vantintel, F. Leberl, M. Grabner, K. Schindler, K. Karner, M. Gervautz, S. Hynst, M. Waelkens, M. Pollefeys, R. DeGeest, R. Sablatnig, and M. Kampel. 3D MURALE: A multimedia system for archaeology. In *Proc. of Intl. EuroConference on Virtual Reality, Archaeology and Cultural Heritage*, page in press, Athens, Greece, Nov. 28–30 2001.
- [4] R. Degeest. The Common Wares of Sagalassos. In M. Walken, editor, *Studies in Eastern Mediterranean Archaeology*, III, 2000.
- [5] I. Konstantinos I. Diamantaras and S.Y. Kung. *Principal Component Neural Networks*. John Wiley & Sons, 1996.
- [6] N. Schliep-Andraschko F.M. Andraschko, A. Krekeler. Terminologie, Klassifikation und computergestützte Bearbeitung der Keramik des 1. Jahrtausends v. Chr. vom Westkom auf Elephantine, Oberägypten. In W.R. Teegen F.M. Andraschko, editor, *Gedenkschrift für Jürgen Driehaus*, pages 327–338. 1990.

- [7] A. Gilboa, A. Karasik, I. Sharon, and U. Silmansky. Towards computerized typology and classification of ceramics. In *Proceedings of: Enter the Past, The E-way into the four dimensions Cultural Heritage Conference*, page in press, 2003.
- [8] R.C. Gonzalez and P. Wintz. *Digital Image Processing, Sec. Edition*. Addison-Wesley Publishing Company, 1987.
- [9] M. Grünewald. Ausgrabungen im Legionslager von Carnuntum (Grabungen 1969-1977). *Der römische Limes in Österreich*, 34:10–11, 1986.
- [10] et. al. I. Gathmann. *ARCOS, ein Gerät zur automatischen bildhaften Erfassung der Form von Keramik*. 1984.
- [11] M. Kampel and R. Sablatnig. On 3d Modelling of Archaeological Sherds. In *Proceedings of International Workshop on Synthetic-Natural Hybrid Coding and Three Dimensional Imaging, Santorini, Greece*, pages 95–98, 1999.
- [12] U. Leute. *Archaeometry: An Introduction to Physical Methods in Archaeology and the History of Art*. John Wiley & Sons, December 1987.
- [13] H. Mara, M. Kampel, and R. Sablatnig. Preprocessing of 3D-Data for Classification of Archaeological Fragments in an Automated System. In F. Leberl and F. Fraundorfer, editors, *26th Workshop of the Austrian Association for Pattern Recognition (OeAGM/AAPR)*, pages 257–264, Graz, Austria, September 2002.
- [14] F. Melero, A. Leon, F. Contreras, and J. Torres. A new system for interactive vessel reconstruction and drawing. In *Proceedings of: Enter the Past, The E-way into the four dimensions Cultural Heritage Conference*, page in press, 2003.
- [15] D.R. Nadeau. Building virtual worlds with vrm. In *Computer Graphics and Applications, IEEE*, pages 18–29, 1999.
- [16] C. Orton, P. Tyers, and A. Vince. *Pottery in Archaeology*, 1993.
- [17] B. Peer. *PRIP-Report - Estimation of the Rotational Axis*. TU Vienna, Institut 183/2 - PRIP, Austria, 2003.
- [18] J. Poblome. Sagalassos Red Slip Ware. In M. Walkens, editor, *Studies in Eastern Mediterranean Archaeology*, II. Brepols, 1999.
- [19] R. Sablatnig and M. Kampel. Model-based registration of front- and backviews. *Computer Vision and Image Understanding*, 87(1–3):90–103, July 2002.
- [20] Robert Sablatnig, Christian Menard, and Petros Dintsis. A preliminary study on methods for a pictorial acquisition of archaeological finds. Technical Report PRIP-TR-010, Vienna University of Technology, Inst. of Computer Aided Automation, Pattern Recognition and Image Processing Group, 1991.

- [21] R. Schreg. *Keramik aus Südwestdeutschland. Eine Hilfe zur Beschreibung, Bestimmung und Datierung archäologischer Funde vom Neolithikum bis zur Neuzeit.* Tübingen, 1978.
- [22] C. Steckner. Samos: Dokumentation, vermessung, bestimmung und rekonstruktion von keramik. In *Akten des 13. internationalen Kongresses für klassische Archäologie*, pages 631–635, 1988.
- [23] S. Ben Yacoub and C. Menard. Robust Axis Determination for Rotational Symmetric Objects out of Range data. In W. Burger and M. Burge, editors, *21 th Workshop of the Oeagm*, pages 197–202, Hallstatt, Austria, May 1997.

A Symbols

Symbol	Description
\mathcal{L}	Indexd list
\mathcal{L}_v	Indexed list of vertices (see Equation 5)
\mathcal{L}_f	Indexed list of faces (see Equation 6)
\mathcal{L}_n	Indexed list of normals
\mathcal{L}_c	Indexed list of color (texture)
\mathbf{p}	Vertex
$\mathbf{p}_{d<0.01}$	Vertices with distance below a certain threshold (i.e. 1%) to e_i
$\mathbf{p}^{intersect}$	Vertices estimated by intersecting $edge_{intersect}$ and e_i
\mathbf{v}	Vector of direction
\mathbf{n}	Normal vector
$f = \{i, j, k\}$	Face (i.e. triangle described by the indices of the vertices \mathbf{p}_k , \mathbf{p}_j and \mathbf{p}_i)
$f_{d<0.01}$	Face containing indices of vertices $\mathbf{p}_{d<0.01}$
$edge = \{(i, j), \dots\}$	Edge(s) i.e. between the vertex $\mathbf{p}[i]$ and $\mathbf{p}[j]$
$edge_{intersect}$	Edges between vertices $\mathbf{p}_{d<0.01}$ width different sign of distance to e_i
$sherd$	3D-model of the sherd
$sherd_{view}, sherd[i]$	untagged view of one side of $sherd$
$sherd_{in}$	view of one side of $sherd$ tagged as inner side
$sherd_{out}$	view of one side of $sherd$ tagged as outer side
$sherd_{in\ orientated}, sherd_{out\ orientated}$	orientated inner/outer side of $sherd$
$sherd_{in\ OR}, sherd_{out\ OR}$	pre-registered inner/outer side of the orientated $sherd_{in\ orientated} / sherd_{out\ orientated}$
$sherd_{OR} = \{sherd_{in\ OR}, sherd_{out\ OR}\}$	registered inner and outer side
$rot = \{\mathbf{p}, \mathbf{v}\}$	Rotational axis
rot_{in}	Rotational axis of $sherd_{in}$
rot_{out}	Rotational axis of $sherd_{out}$
$e_i = \{rot, \mathbf{v}_i\}$	intersecting plane
$profile$	profile line
$profile_i$	profile line estimated by plane e_i
$profile_{max}$	longest profile line - profile line with longest arc length

Table 6: Symbols.

B Input and Output of the Matlab-Scripts

Function	Input	Output
<code>sherd_profile</code>		
<code>read_vrml</code>	Filename	$sherd = \{ \mathcal{L}_v, \mathcal{L}_f, \mathcal{L}_n, \mathcal{L}_c \}$
<code>load_data</code>	Index of sherd	$sherd_1, sherd_2$
<code>calcaxis</code>	$sherd$	rot , Type=(convex,concav)
<code>orientate_object</code>	$rot, sherd$	$sherd_{orientated}$
<code>longest_profile</code>	$sherd_{OR}$	Multiple $profile_i$, Index j of $profile_{max}$
<code>profile_check</code>	$profile_i$	Evaluation: Valid / Invalid
<code>estimate_features</code>	$profile_{max}$	Features=($wdm, rdm, bdm, h_{max}, crat$)
<code>show_results</code>	Features	Features for archaeological classification
<code>reconstruct_object</code>	$profile_{max}$	VRML of the reconstructed object

Table 7: Input and Output of the Matlab-Scripts.

`sherd_profile` is the main-script, which uses all other functions(see Table 7). The pseudo-code of `sherd_profile` describing the algorithm of the system is shown in appendix C.

C Pseudo-Code sherd_profile

```
for i = 1 to 2*n                                % two views per sherd
    read_vrml( vrml_files, matlab_files ) % convert and store data
end

for i = 1 to n
    [ sherd_1, sherd_2 ] = load_data( n ) % load both views per sherd

    % estimate the rotational axis per sherd
    [ rot_1, type_1 ] = calcaxis( sherd_1 )
    [ rot_2, type_2 ] = calcaxis( sherd_2 )

    % stop if both sides have the same type (convex, concav)
    if ( type_1 == type_2 )
        break;

    % orientate both views
    sherd_orientated_1 = orientate_object( rot_1, sherd_1 )
    sherd_orientated_2 = orientate_object( rot_2, sherd_2 )

    % stop if registration failed
    if not ( thickness_max > sherd_orientated_2 - sherd_orientated_1 > thickness_max )
        break

    % estimate multiple profiles and longest profile
    [ profiles_1, index_longest_1 ] = longest_profile( sherd_orientated_1 )
    [ profiles_2, index_longest_2 ] = longest_profile( sherd_orientated_2 )

    % estimate features for evaluation
    eval_1 = profile_check( profiles_1 )
    eval_2 = profile_check( profiles_2 )

    % stop if rotational axis failed
    if( not( eval_1 ) or not( eval_2 ) )
        break

    % estimate features for manual classification, reconstruct object
    longest_profile = profiles_1( index_longest_1 ) + profiles_2( index_longest_2 )
    results = estimate_features( longest_profile )
    show_results( results )
    reconstruct_object( longest_profile )
end
```

D VRML-Sample Cube

```
#VRML V2.0 utf8
Shape {
  appearance Appearance {
    material Material { }
  } #appearance
  geometry IndexedFaceSet {
    colorPerVertex TRUE
    coord Coordinate {
      point [
        1, 1, 1,
        1, 2, 1,
        2, 2, 1,
        2, 1, 1,
        1, 1, 2,
        1, 2, 2,
        2, 2, 2,
        2, 1, 2,
      ] # point
    } # coord
    coordIndex [
      0 1 2 3 -1
      1 5 6 2 -1
      3 2 6 7 -1
      3 7 4 0 -1
      4 5 1 0 -1
      7 6 5 4 -1
    ] # coordIndex
    color Color {
      color [
        1.0 1.0 0.0
        0.0 0.0 1.0
        0.5 0.5 0.5
        0.5 0.5 0.5
        0.0 1.0 0.0
        1.0 0.0 0.0
        0.5 0.5 0.5
        0.5 0.5 0.5
      ] # color
    } # Color
  } # IndexedFaceSet
} # Shape
```