

UNWRAPPING HIGHLY-DETAILED 3D MESHES OF ROTATIONALLY SYMMETRIC MAN-MADE OBJECTS

Bastian Rieck, Hubert Mara, and Susanne Krömker

Interdisciplinary Center for Scientific Computing, Heidelberg University, Germany
{bastian.rieck, hubert.mara, kroemker}@iwr.uni-heidelberg.de

KEY WORDS: Archæology, Cultural Heritage, Geometry, High resolution, Rendering, Visualization, Colour

ABSTRACT:

Rotationally symmetric objects commonly occur at archæological finds. Instead of creating 2D images for documentation purposes by manual drawing or photographic methods, we propose a method based on digitally colored surface models that are acquired by 3D scanners, thereby including color information. We then transform these highly-detailed meshes using simple geometrical objects such as cones and spheres and unwrap the objects onto a plane. Our method can handle curved vessel profiles by dividing the surface into multiple segments and approximating each segment with a cone frustum that serves as an auxiliary surface. In order to minimize distortions, we introduce a simple quality measure based on distances of points to a fitted cone. We then extend our method to approximately spherical objects by fitting a sphere on the surface of the object and applying a map projection, namely the equirectangular projection known from cartography. Our implementation generates true-to-scale images from triangular meshes. Exemplary results demonstrate our methods on real objects, ranging from small and medium-sized objects such as clay cones from the Ancient Orient and figural friezes of Greek vessels to extremely large objects such as the remains of a cylindrical tower of Heidelberg Castle.

1 INTRODUCTION

Due to the demand by archæologists for a fast and accurate documentation as well as analysis of small ancient objects, close-range 3D scanners are increasingly used within museums and on excavation sites (Sablatnig and Menard, 1992). These man-made objects are manufactured from various materials, and they are often rotationally symmetric. Furthermore, many of them depict lots of details, such as iconographic imagery, ornamental paintings or inscriptions. While parallel projections and cross-sections are used to show the general shape of an object (Orton et al., 1993), the details along the surface have to be unwrapped. Traditionally, unwrapping an object is a time-consuming task requiring either a highly skilled draftsman or special equipment for *roll-out photography* (Lynch-Johnt and Perkins, 2008). Fig. 1 shows an example of unwrapped conical friezes of an ancient vessel.

1.1 Related work

Previous research in photogrammetry used cylinder fitting for unwrapping 3D surfaces with known analytical expressions (Karras et al., 1996). Likewise, cartographic projections have already been proved useful for obtaining raster images of paintings on arches and spherical surfaces (Karras et al., 1997). More recent works use *triangle strips* that are adjusted to the surface of arbitrary objects in order to unfold them (Massarwi et al., 2007). A detailed body of methods for unwrapping spherical objects can be found in cartography (Grafarend and Krumm, 2006). Our method enables experts of cultural heritage to rapidly compute planar representations of cultural heritage objects for important tasks such as iconographic analysis. These planar representations cannot be substituted by 3D models e.g. via the Internet, because tasks such as historic interpretations require all parts of an object to be visible.

The simplest geometric primitive for an unwrapping transformation is the cylinder. It has already been used for 3D models of ancient cylinder seals (Pitzalis et al., 2008). Since the cylinder is a degenerate cone, we focus on unwrapping approximately conical objects in the next section. However, as there is a vast number of objects with a spherical shape, we also show how to unwrap a sphere.

1.2 Data format

In the following, we assume that the user is working with 3D representations of cultural heritage objects, such as ceramic vessels or cuneiform tablets. We require an object to be specified as a *triangular mesh*, e.g. in *Stanford Triangle Format* (PLY). This mesh is composed of *vertices*, i.e. points in \mathbb{R}^3 , and *faces*, i.e. triangles consisting of three distinct vertices. The *edges* of the faces are described implicitly by this representation. Triangular meshes are commonly used and generated by most of commercial 3D scanners: Three non-collinear vertices always define a plane and allow for the simple calculation of normals and other properties of the faces.



Figure 1: Unwrapping both figural friezes of a vessel (KHM Inv.-No. 3618) by fitting two different cones (Bechtold et al., 2010).

2 DEVELOPABLE SURFACES

In this part of the paper, we will focus on *developable surfaces* that can be unwrapped without any distortions. The unwrapping transformation is bijective, i.e. it can be reversed without loss of accuracy. For simplicity, we only talk about *finite cones* and *cone frusta*, i.e. truncated cones. We will later generalize these techniques to unwrap cylinders, as well.

2.1 Fitting a cone frustum

Many ceramic vessels can be described by conical shapes. Typically, the objects are at least *approximately* rotationally symmet-

ric due to their manufacturing process, which uses a rotational plate. In most cases, one can think of vessels as being comprised of several cone frusta. Usually, two or three frusta are sufficient for unwrapping these vessels—see Fig. 1 for a typical example.

For fitting a cone frustum, we let the user interactively specify an *axis* of the cone as well as *two points* on the surface of an object. We then calculate the *height* of the truncated cone by projecting the selected points to the cone axis. The upper and lower radius of the cone are then determined by the position of the points selected by the user. Furthermore, by employing cone frusta, the user can exclude regions of the vessel that are removed upon unwrapping the object. This is useful when the user is interested in a depiction around the middle of the vessel, but does not want to include handles or the base of the vessel. From an algorithmic point of view, this approach of cone fitting offers additional possibilities: (i) The process of determining a rotational axis may be automated. In recent publications (Mumford and Cao, 2002, Mara and Sablatnig, 2006), methods for automatically finding the best rotational axis have been introduced. (ii) If an axis has been determined, numerical fitting methods can be applied in order to find the optimum radii of the cone frustum with respect to some quality measure, such as minimized squared distances (Strutz, 2010).

2.2 Cone distance calculation

When unwrapping an arbitrary object that is not a perfect cone, we want to minimize any distortions occurring due to the transformation. To this end, we need to define a *quality measure*. In the case of a complete cone, we use the distance of each point of an object to the fitted cone as a simple quality measure. When fitting a cone frustum, this method has the advantage of determining whether a point is located above or below the planes of the cone frustum such that distance calculations are not necessary. In these cases, the corresponding vertices are skipped and will not be processed any further. This greatly improves the speed of the unwrapping transformation.

For calculating the cone distance, we assume that the cone is situated in the origin of the coordinate system and opens towards the positive y -axis. The main idea is to use the angle $\gamma = \beta - \alpha$ between the position vector of the vertex and the vector along the surface of the cone. With Fig. 2 depicting the required angles, we obtain

$$\text{dist}(\mathbf{p}) = \sin(\gamma) \cdot \|\mathbf{p}\| = d, \quad (1)$$

where $\|\mathbf{p}\|$ is the Euclidean norm of the vector connecting the apex of the cone and \mathbf{p} . If the angle β becomes larger than $\pi/2$, we set $\text{dist}(\mathbf{p}) := \|\mathbf{p}\|$.

When fitting a cone frustum, we use the same distance calculation as defined above. In addition, we check whether the *orthogonal projection* of a query point onto the y -axis is outside the frustum. If so, we assign the point a semi-transparent colour. Points whose distances are defined, on the other hand, will be drawn using a heat map colour ramp. This enables the user to see the quality of the fitted cone frustum in one glance—see Fig. 3.

2.3 Unwrapping a cone

Before unwrapping a cone, we first transform the object and the cone into a *local* coordinate system with the axis of the fitted cone being the y -axis of the coordinate system. This transformation simplifies later calculations. See Fig. 4 for the current setting. We describe the required formulæ and refer to (Bechtold et al., 2010) for a more detailed derivation. For simplicity, we express the coordinates of the vertices in a conical coordinate system, i.e. one that employs an angle φ_i and a height h_i to uniquely describe the position of a point \mathbf{p}_i on the surface of the cone.

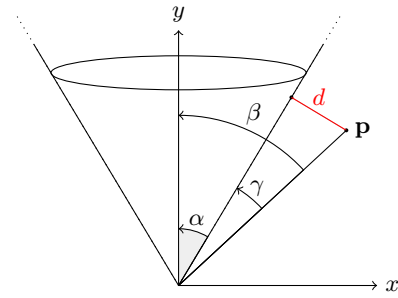


Figure 2: Angles used for calculating the distance to a cone.



Figure 3: Heat map visualization of the distance to a fitted cone. Due to the high curvature of the vessel (KHM Inv.-No. 3601) larger distances at the middle part of the frustum are visible.

Unwrapping a cone is of the points of the object. Since the maximum height of a point in the conical coordinate system is given as h , whereas the unwrapped cone has a maximum polar radius of s , we require some sort of *scaling*. The same applies to the vertex angles: In the conical coordinate system, they span the half-open interval of $[0, 2\pi)$, whereas they are limited by an angle θ in the plane of the unwrapped cone. From Fig. 4, we obtain the ratio

$$\frac{2\pi r}{\theta} = \frac{2\pi s}{2\pi} \Leftrightarrow \theta = 2\pi \frac{r}{s}. \quad (2)$$

Consequently, the *angle scale factor* λ is given as

$$\lambda = \frac{r}{s} = \sin(\alpha). \quad (3)$$

For the *height scale factor* μ , we observe that a point at height h will be transformed to height s when being unwrapped. Hence, μ is given as

$$\mu = \frac{s}{h} = \frac{1}{\cos(\alpha)}. \quad (4)$$

We thus arrive at the following transformation for vertices:

$$T_c(\varphi_i, h_i) = (\lambda\varphi_i, \mu h_i) \quad (5)$$

This handles the unwrapping of a cone that is centered along the y -axis of the coordinate system.

2.4 Unwrapping a cone frustum

In the case of a cone frustum, care must be taken only to apply this transformation to vertices that are contained between the upper and lower plane of the cone frustum. To describe a cone frustum, we require an *axis*, which is again supposed to be the y -axis of the coordinate system, and two *limiting planes* perpendicular to the y -axis. We determine these limiting planes by specifying an upper position y_u and a lower position y_l , with $y_l < y_u$. The two

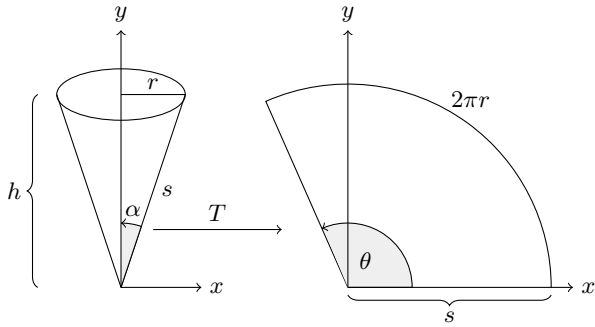


Figure 4: Depiction of the unwrapping process of a cone. Note how angles and heights of the points on the cone are changed.

positions then yield a simple and criterion for the transformation function T_f of a cone frustum:

$$T_f(\varphi_i, h_i) = \begin{cases} (\lambda\varphi_i, \mu h_i) & h_i \in [y_l, y_u] \\ \text{undefined} & \text{else} \end{cases} \quad (6)$$

In practice, we let the user interactively select two points on the surface of a digital object. From these two points, we obtain an upper radius r_u and a lower radius r_l , with $r_l < r_u$ because we require the apex of the complete cone to be the origin of the coordinate system. We then calculate the respective y -positions required for Eq. (6):

$$y_u = \frac{r_u}{\tan(\alpha)}, \quad y_l = \frac{r_l}{\tan(\alpha)} \quad (7)$$

In this section, we exploited the fact that the cone axis is the y -axis of the coordinate system.

In many cases, the user wants to center the projection of the cone on a certain angle, be it to highlight a certain figurative element on a vessel or to push handles to the side where the cone is split. Following usage in cartography, we call this process *selecting a prime meridian*. In mathematical terms, this is accomplished by applying a *fixed* offset $\sigma \in [-\pi, \pi]$ to the projected vertex coordinates, i.e. $\theta_i := \theta_i - \sigma$, and then normalizing the vertex angle θ_i to a range of $[-\pi, \pi]$. Likewise, we have a *splitting meridian*, i.e. the angle opposite to the prime meridian. This is where the cone will be ‘split’ when being unwrapped to the plane.

2.5 Unwrapping a 3D mesh

So far, we have only discussed the unwrapping of a perfectly conical object. In practice, objects are *approximately* conical at best and may consist of multiple layers. In order to apply the transformation described in Eq. (6), we perform the following steps for each vertex $\mathbf{p}_i = (x_i, y_i, z_i)^T$: (i) Determine coordinates $\varphi_i = \arctan(z_i/x_i)$, $h_i = y_i$ of vertex within the conical coordinate system. (ii) Obtain *radial distance* $r_i = \sqrt{x_i^2 + z_i^2}$ of vertex with respect to cone axis. (iii) Set new coordinates of vertex to be $(x'_i, y'_i, z'_i) := (T_f(\varphi_i, h_i), r_i)$.

The second step solves the *depth-sorting problem* (Foley et al., 1997, pp. 673–686): Digital objects acquired with a structured light scanner do not have an infinitely thin surface. Instead, each object is composed of an inner and outer shell. The transformation described in Eq. (6) is 2D, which means that points of all layers would be projected on top of each other. By using a radial distance, we obtain a 3D object with the correct depth order—see Fig. 5 for an example.

The mesh consists of *discrete* data points in \mathbb{R}^3 . Therefore, removing all points whose distances are undefined results in truncated triangles and rough edges along the upper and lower part

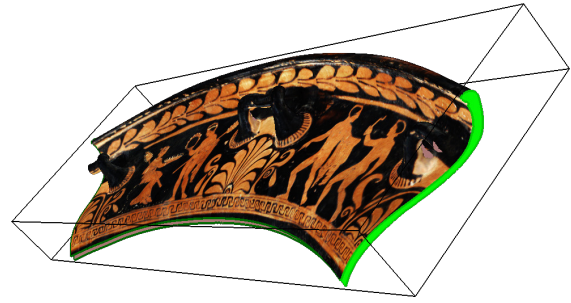


Figure 5: An unwrapped version of an approximately conical vessel, UMJ Inv.-No. 4611 (Karl et al., 2009), that has been rotated to convey the depth information. The highlighted contour is the cross-section at the splitting meridian. This line is identical to the *profile line* as drawn by archaeologists (Orton et al., 1993).

of the fitted cone. Truncated triangles, in turn, appear as jagged edges in the resulting object. As a remedy, we implemented an algorithm employing *splitting planes*. The main idea is to define a plane in *Hesse normal form*, and intersect it with the object. The intersection will create points that are situated on the plane, along the edge of a triangle. By re-triangulating the existing points, the object may be split seamlessly along the plane.

3 NON-DEVELOPABLE SURFACES

In the following, we will handle spheres and the *equiarectangular projection* only. However, the methods presented in this section can be generalized to use any cartographic projection or transformation operation.

3.1 Fitting a sphere through four points

If a vessel has spherical characteristics, unwrapping the object by using a *sphere* instead of a *cone* yields better results. We let the user choose four points $\{(x_i, y_i, z_i)^T \mid i = 1 \dots 4\}$ on the surface of the object. By solving a determinant equation,

$$\det \begin{pmatrix} x^2 + y^2 + z^2 & x & y & z & 1 \\ x_1^2 + y_1^2 + z_1^2 & x_1 & y_1 & z_1 & 1 \\ x_2^2 + y_2^2 + z_2^2 & x_2 & y_2 & z_2 & 1 \\ x_3^2 + y_3^2 + z_3^2 & x_3 & y_3 & z_3 & 1 \\ x_4^2 + y_4^2 + z_4^2 & x_4 & y_4 & z_4 & 1 \end{pmatrix} = 0, \quad (8)$$

we obtain a general equation for the sphere; see (Osgood and Graustein, 1921), pp. 524–526, for a detailed derivation. Expanding the determinant by minors M_{1i} , we get

$$(x^2 + y^2 + z^2)M_{11} - xM_{12} + yM_{13} - zM_{14} + M_{15} = 0.$$

The implicit equation of a sphere of radius r centered at a point $(x_0, y_0, z_0)^T$ is given by $(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = r^2$. Equating the terms of the last two equations yields both radius and center of the sphere:

$$x_0 = \frac{1}{2} \cdot \frac{M_{12}}{M_{11}}, \quad y_0 = -\frac{1}{2} \cdot \frac{M_{13}}{M_{11}}, \quad z_0 = \frac{1}{2} \cdot \frac{M_{14}}{M_{11}} \quad (9)$$

$$r^2 = x_0^2 + y_0^2 + z_0^2 - \frac{M_{15}}{M_{11}}$$

In Eq. (9), M_{11} is required to be nonzero. If this is not the case, we may assume that the points are either coplanar or three of the points are collinear. Consequently, fitting a sphere through these points is impossible.

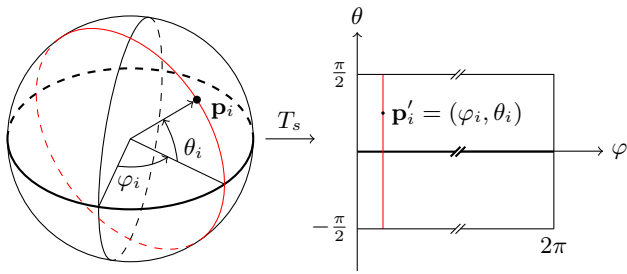


Figure 6: The equirectangular projection of a unit sphere.

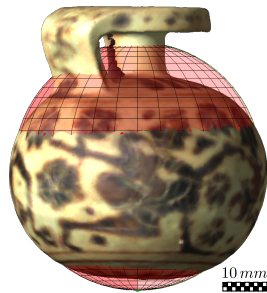


Figure 7: An *aryballos*, UMJ Inv.-No. 8738 (Karl et al., 2009), a vessel with a spherical shape that is well-suited for the equirectangular projection. Note that the sphere does not envelop the object. This signifies that the object is not perfectly spherical.

3.2 Equirectangular projection of spherical meshes

We assume that points of a perfectly spherical object are given and that the center of the sphere is the origin of the coordinate system. Having described the sphere on which they are situated, we are now able to perform an *equirectangular projection*, thereby unwrapping the sphere to the plane. To accomplish this, we convert the Euclidean coordinates of the points of the object into *spherical coordinates*: Let a point in \mathbb{R}^3 be described by $\mathbf{p}_i = (x_i, y_i, z_i)^T$. Then the corresponding spherical coordinates are given by

$$\begin{aligned} r_i &= \sqrt{x_i^2 + y_i^2 + z_i^2} \\ \varphi_i &= \arctan\left(\frac{z_i}{x_i}\right), \quad \theta_i = \arcsin\left(\frac{y_i}{r_i}\right) \end{aligned} \quad (10)$$

and we obtain the following *projection function*:

$$T_s(r_i, \varphi_i, \theta_i) = (\varphi_i, \theta_i) \quad (11)$$

Fig. 6 depicts the relation between spherical coordinates and the resulting equirectangular projection. Note that a variant of the equirectangular projections scales all points by the radius r of the sphere. This ensures that the projection is true to scale.

3.3 Equirectangular projection of arbitrary meshes

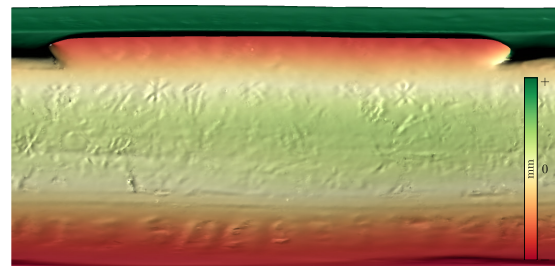
Eq. (11) is insufficient for arbitrary meshes: Not all vertices of the object may be situated on the outside of the same sphere. Since we lose all *depth information*, different vertices would overlap—this is essentially the same problem that occurs when unwrapping arbitrary 3D meshes using cone frusta. We hence again use the radius of the spherical coordinates in order to add depth information. This yields the following procedure for each point $\mathbf{p}_i = (x_i, y_i, z_i)^T$: (i) Fit a sphere to the object and make the center of the sphere the new origin of the coordinate system. (ii) Determine spherical coordinates r_i, φ_i, θ_i for \mathbf{p}_i by using Eq. (10). (iii) Set new coordinates of point to be $(x'_i, y'_i, z'_i)^T :=$



(a) Manual drawing



(b) Equirectangular projection



(c) Distances to the fitted sphere

Figure 8: A comparison of a manual drawing (courtesy of S. Karl) and the equirectangular projection for the *aryballos* in Fig. 7.

$(T_s(\varphi, \theta), r_i)$. This transformation ensures that the outside of the *original* object remains the outside of the *projected* object. Hence, we again obtain a 3D projection, which may be used for further post-processing, such as detecting curvature, inscriptions, and other surface details. Fig. 7 depicts a vessel that is well-suited for an equirectangular projection along with a fitted sphere. The result of the projection operation for this object is shown by Fig. 8.

4 RESULTS

As this work was originally motivated by the Austrian *Corpus Vasorum Antiquorum* (CVA) project, we provided 25 unwrapped vessels for CVA volume Austria 5 (Kratzmüller, 2014). All the final images are orthographic projections and hence true to scale. Our implementation can create high-resolution raster images of the transformed objects by using OpenGL-based tiled rendering techniques. Examples are shown in the previous sections. Fig. 1 depicts the usage of multiple cones for one vessel, while Fig. 10 shows the difference between unwrapping transformations using a cone and a sphere.

Cuneiform tablets from the Ancient Orient have their details represented within geometry. These ancient hand-writings become only visible when properly illuminated e.g. by bright sunlight. While virtual illumination is a global operation, we used local filters based on Multi-Scale Integral Invariants (MSII) (Mara et al., 2010) to compute a false colour texture map that highlights curvature-related features. An example of cuneiform inscription and such texture maps on a clay cone are shown in Fig. 9.

Finally, we demonstrate the independence of scale and acquisition device by applying the unwrapping transformation to one of the towers of *Heidelberg Castle*, namely the famous ruins known

as the *Pulverturm* (powder tower). Together with the *Institute of Geography*, Heidelberg University, we documented the present conditions of this part of the castle using *Terrestrial Laser Scanning* (Siart et al., 2012) with a *Riegl VZ-400*. Fig. 11 shows the tower and its 3D model.

For the model of the *Pulverturm*, we extended the techniques from Section 2.5 to unwrap cylinders, as well: A cylinder is a degenerate cone whose upper and lower radius are equal. This equality yields a cone angle of $\alpha = 0$, but, following Section 2.4, we can still use an angle φ_i and a height h_i to uniquely describe the position of a point \mathbf{p}_i on the surface of the cylinder. With the notation of Section 2.3, Eq. (3) evaluates to $\lambda = 0$, while Eq. (4) evaluates to $\mu = 1$. Since $\lambda = 0$, the transformation function from Eq. (5) maps all angles φ_i to 0, which is incorrect. We handle this by setting $T_{\text{cyl}}(\varphi_i, h_i) = (r\varphi_i, h_i)$, where r denotes the radius of the fitted cylinder. Scaling the angle φ_i with r ensures that the transformation is true to scale. Using the previous equation, we then proceed as described in Sec. 2.5. The result of the transformation is shown in Fig. 11.

5 CONCLUSIONS AND FUTURE WORK

The work shown in this article demonstrates the use of elementary geometry to unwrap high-resolution 3D meshes of highly detailed objects of cultural heritage. The presented algorithms were successfully tested on a large number of real-world 3D data from archaeological projects. Due to this interdisciplinary work, we can suggest several improvements: (i) For some types of objects, it may be advantageous to use standard cartographic projections. There are many projections available offering a variety of different results: Some preserve *angles*, while others preserve *areas*. For further discussion regarding several cartographic methods, we refer to (Grafarend and Krumm, 2006). (ii) Future additions to our framework will be the usage of primitives such as ellipsoids or parameterizable surfaces. (iii) To improve the color representation, flashlights can be added to cancel shadows during acquisition of the texture-map of the 3D model as described by (Dellepiane et al., 2010).

ACKNOWLEDGMENTS

We would like to thank Dir. Dr. Alfred Bernhard-Walcher and Dr. Bettina Kratzmüller of the collection of antiquities of the *Kunsthistorisches Museum*, Vienna, as well as Stephan Karl from *Graz University* for kindly authorizing us to present images of items from the respective collections. For the collaborative work on cuneiform tablets we thank the *Assyriology Group*, *Heidelberg University*, and the *Assur Forschungsstelle* of the *Heidelberg Academy of Sciences and Humanities*.

For data acquisition and post-processing of the terrestrial laser scan data, we thank Markus Forbriger and Larissa Müller of the *Institute of Geography*, *Heidelberg University*. Access to the *Pulverturm* was kindly granted by *Heidelberg Castle administration* and plans were provided by Dr. Claudia Mohn.

REFERENCES

Bechtold, S., Krömker, S., Mara, H. and Kratzmüller, B., 2010. Rollouts of Fine Ware Pottery using High Resolution 3D Meshes. In: Proc. VAST Int. Symp. on Virtual Reality, Archaeology and Cultural Heritage, Paris, France, pp. 79–86.

Dellepiane, M., Callieri, M., Corsini, M., Cignoni, P. and Scopigno, R., 2010. Improved Color Acquisition and Mapping on 3D Models via Flash-based Photography. *Journal on Computing and Cultural Heritage*.

Foley, J. D., van Dam, A., Feiner, S. K. and Hughes, J. F., 1997. *Computer Graphics: Principles and Practice in C*. 2nd edn, Addison-Wesley Professional.

Grafarend, E. W. and Krumm, F. W., 2006. *Map projections*. Springer.

Karl, S., Modl, D. and Prod, B. (eds), 2009. *Katalog Archäologiemuseum. Schild von Steier*, Vol. 22, Phoibos Verlag, Graz, Austria.

Karras, G. E., Patias, P. and Petsa, E., 1996. Digital monoplotting and photo-unwrapping of developable surfaces in architectural photogrammetry. *Int. Arch. Photogramm. and Remote Sensing XXXI*, Part B5, pp. 290–294.

Karras, G. E., Patias, P., Petsa, E. and Ketipis, K., 1997. Raster projection and development of curved surfaces. *Int. Arch. Photogramm. Remote Sensing XXXII*, Part 5C1B, pp. 179–185.

Kratzmüller, B., 2014. *CVA Österreich 5: Attisch Geometrische, Protoattische und Attisch Schwarzfigurige Vasen Band 1*. Verlag der Österreichischen Akademie der Wissenschaften.

Lynch-Johnt, B. A. and Perkins, M., 2008. *Illustrated Dictionary of Photography: The Professional's Guide to Terms and Techniques*. Amherst Media Inc.

Mara, H. and Sablatnig, R., 2006. Orientation of Fragments of Rotationally Symmetrical 3D-Shapes for Archaeological Documentation. In: Proc. 3rd Int. Symp. on 3D Data Processing, Visualization, and Transmission, pp. 1064–1071.

Mara, H., Krömker, S., Jakob, S. and Breuckmann, B., 2010. GigaMesh and Gilgamesh - 3D Multiscale Integral Invariant Cuneiform Character Extraction. In: Proc. VAST Int. Symp. on Virtual Reality, Archaeology and Cultural Heritage, pp. 131–138.

Massarwi, F., Gotsman, C. and Elber, G., 2007. Papercraft Models using Generalized Cylinders. In: Proc. 15th Pacific Conf. on Computer Graphics and Applications, pp. 148–157.

Mumford, D. and Cao, Y., 2002. Geometric structure estimation of axially symmetric pots from small fragments. In: Proc. Intl. Conf. on Signal Processing, Pattern Recognition, and Applications, pp. 92–97.

Orton, C., Tyers, P. and Vince, A., 1993. *Pottery in Archaeology*. Cambridge University Press.

Osgood, W. F. and Graustein, W. C., 1921. *Plane and solid analytic geometry*. The Macmillan Company.

Pitzalis, D., Cignoni, P., Menu, M. and Aitken, G., 2008. 3D enhanced model from multiple data sources for the analysis of the Cylinder seal of Ibni-Sharrum. In: Proc. 9th Int. Symp. on Virtual Reality, Archaeology and Cultural Heritage, pp. 79–84.

Sablatnig, R. and Menard, C., 1992. Stereo and structured light as acquisition methods in the field of archaeology. In: *Mustererkennung 1992*, Springer, pp. 398–404.

Siart, C., Ghilardi, M., Forbriger, M. and Theodorakopoulou, K., 2012. Terrestrial laser scanning and electrical resistivity tomography as combined tools for the geoarchaeological study of the Kritsa-Latô dolines. *Géomorphologie (1)*, pp. 59–73.

Steible, H., 1991. *Die neusumerischen Bau- und Weihinschriften I*. *Freiburger altorientalische Studien*, Vol. 9, Steiner.

Strutz, T., 2010. *Data Fitting and Uncertainty: A Practical Introduction to Weighted Least Squares and Beyond*. Vieweg and Teubner.

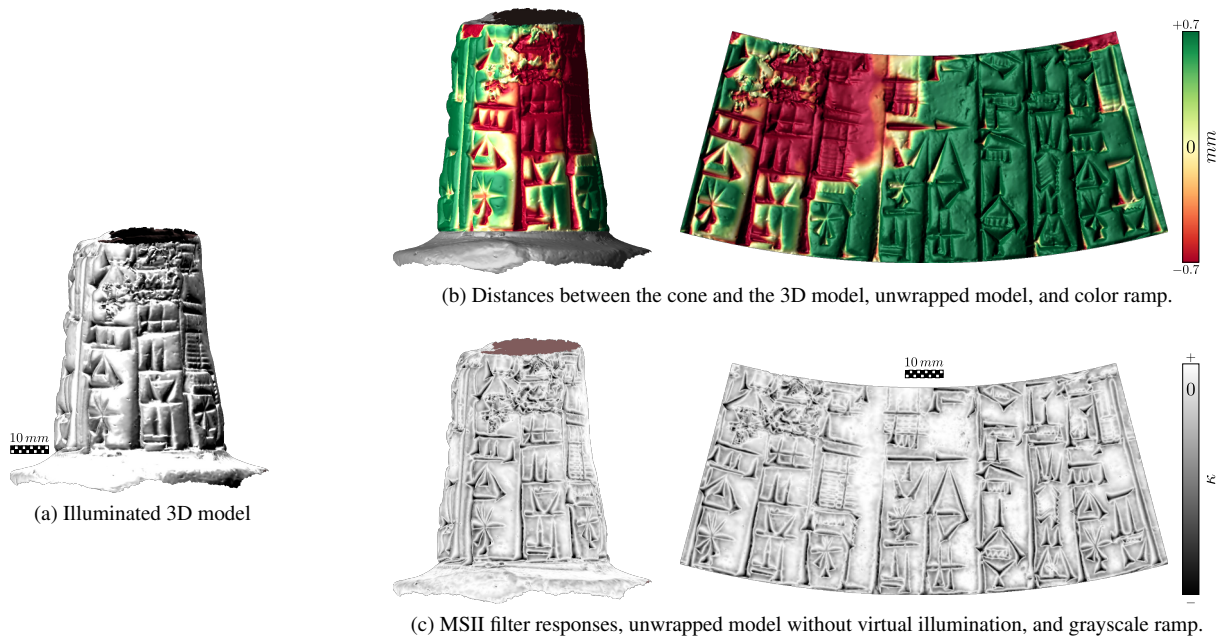


Figure 9: 3D model of a building inscription with cuneiform characters on a *Tonnage* (clay cone), ca. 2200–2000 B.C., *Sammlung des Seminars für Sprachen und Kulturen des Vorderen Orients, Assyriologie, Universität Heidelberg*. For information about the inscription and its known 451 ancient duplicates see (Steible, 1991, *Text Ur-Baba 7*).



Figure 10: *Oinochoe*, UMJ Inv.-No. 4181 (Karl et al., 2009). From left to right: A scanned version of the object, an unwrapped variant using a cone, and the result of the equirectangular projection. Unpainted parts of the vessel have been discarded.

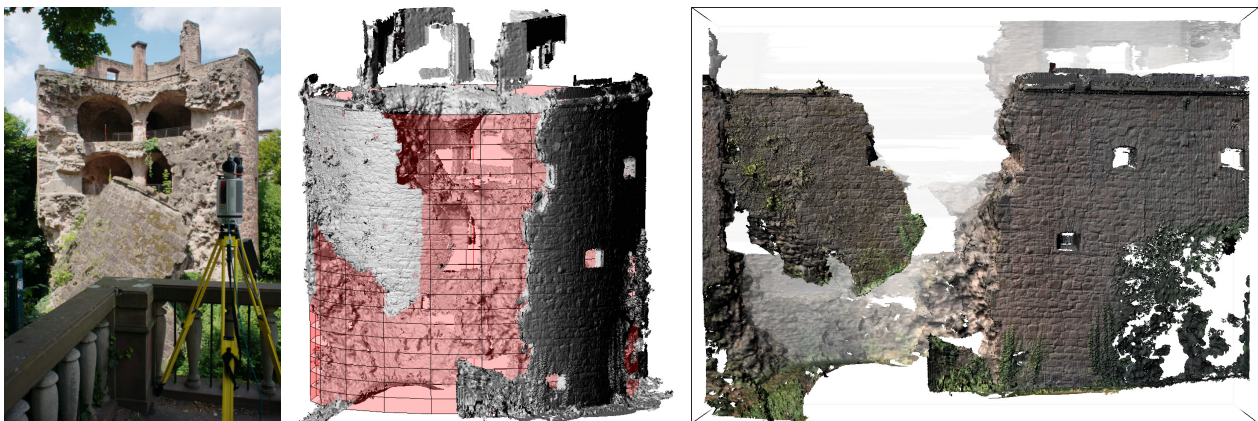


Figure 11: *Pulverturm* of Heidelberg Castle. Left: Photograph of the tower and the *Riegl VZ-400 Terrestrial Laser Scanner*. Middle: Orthographic, true-to-scale, isometric view of the 3D model with the realigned wall fragment. The cylinder for the unwrapping transformation has a radius of 12m and is shown in red color. Right: Perspective front view of the unwrapped 3D model.