

writing also face the challenge of depicting the text inscribed on the cuneiform tablets when creating illustrations for use in books or research papers.

The simplest solution to the illustration problem is to photograph a tablet illuminated with a raking light source, as shown in the examples of Figures 1(a), 2(a), and 5(a). Although most of the writing on a cuneiform tablet is on the front (obverse) or back (reverse) faces, these faces are often curved. The curvature causes distortions from foreshortening on the periphery when looking perpendicularly at the center of the tablet, as seen in Figure 1(a). This effect is exacerbated by the fact that, on many cuneiform tablets, the text wraps around one or more edges. Finally, the raking illumination poses a problem when parts of the tablet are in shadow, thus giving the observer no visual information about critical portions of an inscription. An additional form of obfuscation from illumination occurs when the orientation of an inscription is nearly aligned with the incoming light direction.

An alternative to photography is to manually transcribe the tablet symbols, producing a result simi-

lar to Figure 2(b). Unfortunately, this task is tedious, and it fails to portray the nuances and ambiguity of the original. For most applications it is preferable to reproduce the text in a form that conveys the original in its entirety.

If we are willing to extend the publication medium beyond the printed book, we can consider using a computer to visualize the tablets. Digital photographs of the tablet may be taken from a variety of camera angles, processed, and viewed interactively, as in QuickTime™ Virtual Reality (QTVR). Another option is to scan the tablet, thereby creating a 3D model, and then display this model using an interactive 3D rendering program. Such programs enable a user to manipulate a virtual camera and light source relative to the model in order to view all of the markings on the tablet. Yet another approach is the lightfield, which combines photographic imagery with 3D-like interactivity [Lev96]. Lightfield viewers display images from new camera positions and orientations by interpolating from a dense array of photographic images taken from known camera positions and orientations. Here again, the user can manipulate the virtual camera

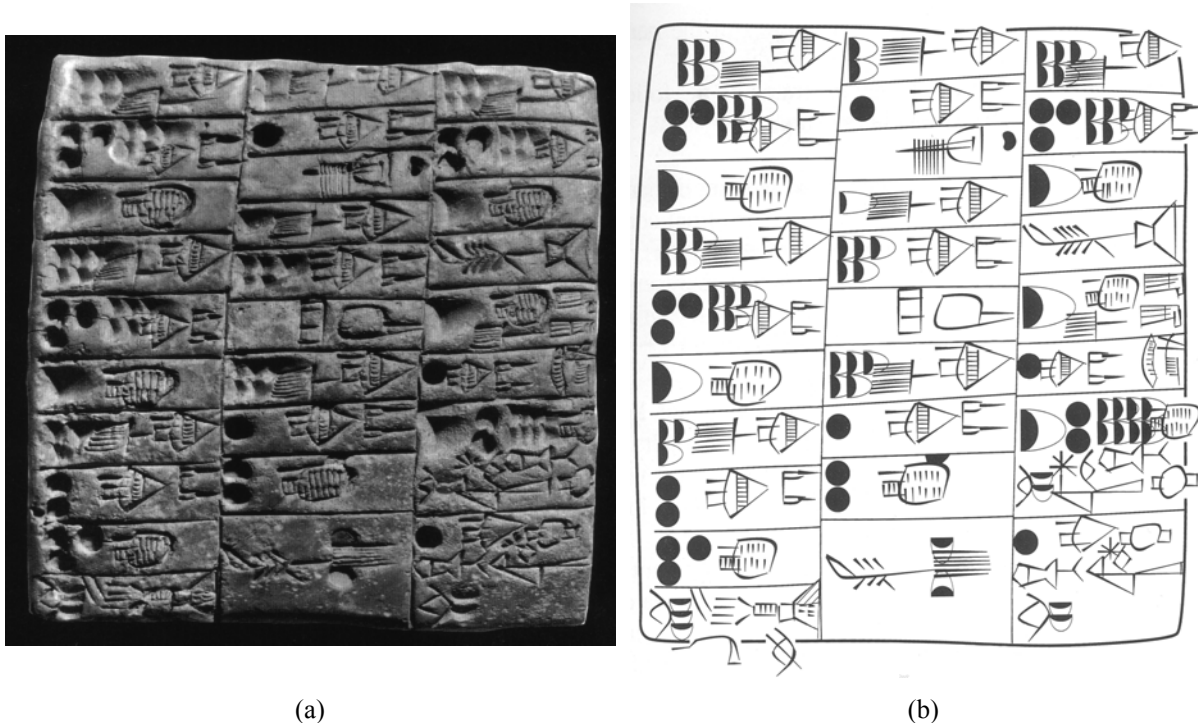


Figure 2. On the left is an example of a raking-light photograph of a proto cuneiform tablet (circa third millennium BC); it records an account of malt and barley groats used in beer production for four administration officials [Nis93]. Although the front surface of this tablet is relatively flat, the text wraps around the edges in places, making it illegible from a single viewpoint. The right picture is a pen-and-ink style transcription of this tablet. (Photo courtesy Christie's London; illustration from *Archaic Bookkeeping* [Nis93].)

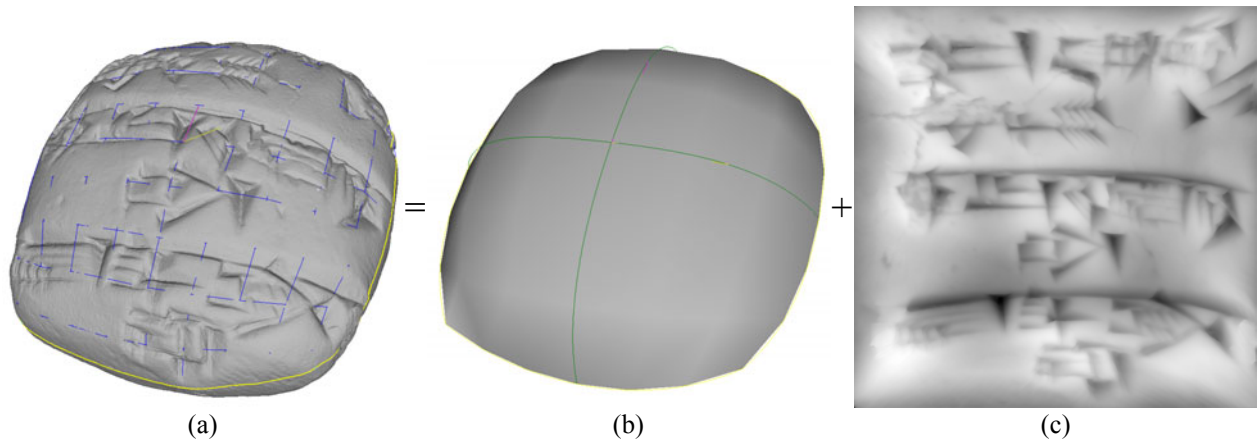


Figure 3. The 3D computer model ((a), also shown in Figure 1(b)) is decomposed into a coarse B-spline, (b), which captures the basic tablet shape, and a displacement map (c), which encodes the inscribed detail. We discard the B-spline surface, retaining only the displacement map.

to see all the tablet’s inscribed characters. However, there are benefits to the printed page, such as longevity and availability, which are not afforded by computers.

We have developed a semi-automatic method for concisely displaying the tablets’ inscribed writing, thereby providing a clear visualization that can be printed on paper. The tablets are first scanned with 3D range scanners, and the scan data are employed to construct a high-resolution 3D model (at a resolution of 50 microns). Next, we unwrap and warp the tablet surface to form a set of flat rectangles, one per side or edge of the tablet. This process permits all of the writing to be seen at once, although necessarily slightly distorted. Finally, we apply curvature coloring and accessibility coloring to the unwrapped text, thereby replacing raking illumination with a non-photorealistic rendering technique.

Previous Work

There are several major institutes of cuneiform study around the world, including the Faculty of Oriental Studies at Oxford, the Oriental Institute at the University of Chicago, the Department of Near East Languages and Cultures at UCLA, the Max Plank Institute for the History of Science in Berlin, and the Free University of Berlin. Researchers from the last three institutions in this list are collaborating on the Cuneiform Digital Library Initiative, in which tablets are scanned on a flatbed scanner and abstracted into pictographic symbols (as in Figure 2(b)). Their system [Nis93] employs a vector-based graphics program, such as Adobe Illustrator, and human interaction. In their sys-

tem, a user selects a type of mark from a graphical palette. Next, a scanned tablet image is used as a reference to aid the user in accurate placement of symbolic marks, such as wedge shapes. The computer also performs a statistical analysis of the inscriptions, thereby deducing some meaning about them. Another capability of their system is the portrayal of the marks on the tablet as clear black-and-white illustrations, free of soil or the effects of weathering. However, cuneiform inscriptions are open to interpretation, so valuable information may be filtered out during this process. Also, although their system is effective on flat tablets, it doesn’t work as well for rounded tablets or those whose writing extends around the sides of the tablets. By contrast, our method only visualizes the marks – it doesn’t transcribe them.

Malzbender [Mal01] et al. have recently visualized cuneiform tablets by taking pictures of a tablet with different light directions to compute what they call a “polynomial texture map.” These maps have a set of coefficients stored at each texel that are used for evaluating a biquadratic polynomial of the light direction. When viewing the clay artifacts, the effect of their method is similar to using photometric stereo to estimate a normal map and then interactively rendering it with different materials and lighting.

In other work, Rushmeier [Rus01] et al. published a method for computing “horizon maps” from images captured under controlled lighting. Horizon maps store self-shadowing information for a set of quantized azimuthal light directions at each point of a terrain

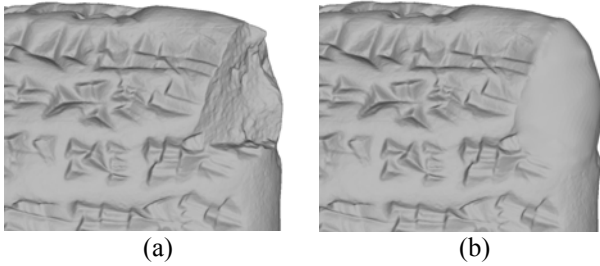


Figure 4. Tablet models may be broken with missing fragments, as seen on the upper right corner of (a). The user may need to repair the missing portions so the displacement maps can be extracted, as seen in (b).

surface. These may be rendered in real-time with modern hardware. When used in conjunction with bump maps, horizon maps produce realistic renderings. In contrast to the 3D scanning required for our method, the input images may be acquired relatively quickly in both MalzBender’s and Rushmeier’s methods. However their techniques do not address the unwrapping problem and are not suitable for publishing tablets.

Scanning

Our software pipeline is depicted in Figure 1. The first step in this pipeline is scanning a tablet with a 3D laser range scanner. Unfortunately, capturing the finest details on a cuneiform tablet requires a very high-resolution 3D scanner. Robert Englund, professor of Near Eastern Languages and Cultures at the University of California, Los Angeles, has found that the smallest details on cuneiform tablets (particularly from the small script of the second millennium B.C. and later) requires a resolution of 300 dots per inch (dpi), with 600 dpi preferred [personal communication]. After some experimentation, we confirmed his finding. This implies that a 3D scanner having a resolution of 50 microns is required. For the examples shown in this paper, we used a laser triangulation range scanner with exactly this resolution. It was custom-built at the National Research Council of Canada [NRC].

Since each scan of an object using a swept laser scanner sees only one side of the object, it is necessary to scan the object multiple times from different viewpoints and to merge the resulting range images. There are a variety of ways to merge range scans together into a single mesh. We used a volumetric intermediate representation to represent a cumulative weighted signed distance function [Cur01]. Each range image is

scan-converted to a distance function, and then combined with data already acquired using a simple additive scheme. Extracting the zero-distance isosurface from the accumulated volume generates the final dense triangle mesh manifold. Applying this method to 22 range images of the tablet in Figure 1(a), we created a 3.2-million-triangle model with a resolution of 50 microns, as shown in the example of Figure 1(b).

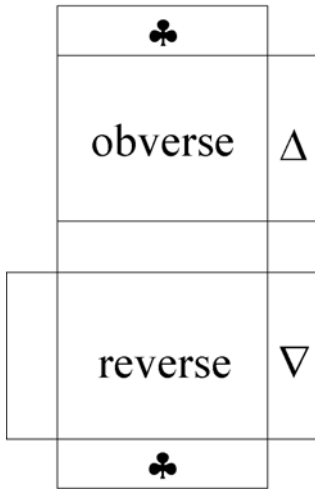
In the next step of the pipeline, we will unwrap approximately rectangular patches of text, but we must first prepare the mesh for the unwrapping. Many cuneiform tablets are broken or are missing significant pieces, as seen in Figure 4. Unwrapping a rectangular patch from a (broken) tablet that is L-shaped would leave the unwrapped patch extremely distorted. Therefore, we virtually repair the tablet model by using Paraform software (published by Paraform Inc.) to define new geometry where pieces have broken away from the original tablet. The resulting repaired areas do not have inscriptions, but approximate the shape of the tablet prior to writing. In particular, Paraform’s software provides a virtual spherical tool with which we can interactively sculpt the broken surfaces of the mesh, gradually pushing the vertices outwards to approximate the unbroken surface, as seen in Figure 4(b).

Unwrapping

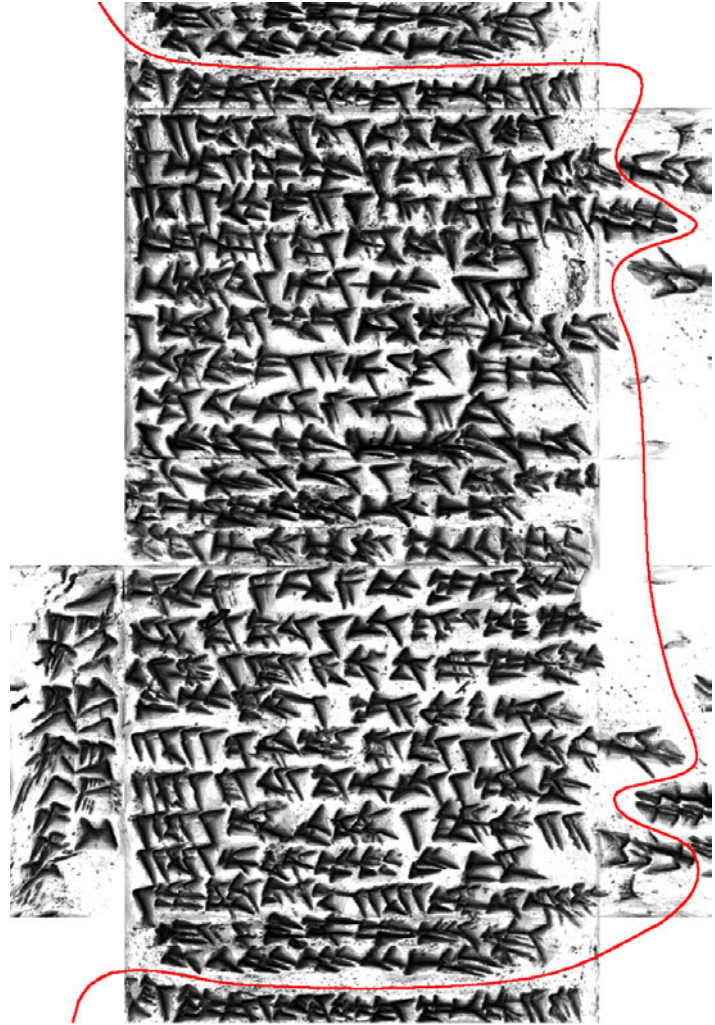
The second step in our pipeline is to unwrap the characters from the model. Once again using Paraform’s software, we partition the irregular mesh into rectangular patches [Kri96]. Specifically, for each patch, we interactively specify the patch boundary as four connected curves on the surface of the 3D tablet model. Next, a grid of springs is fit to each patch by iteratively relaxing and subdividing the spring grid. Relaxing makes the gridlines more evenly spaced across the model, while subdividing creates a better fit of the spring grid to the irregular mesh. The resulting spring grid defines a U-V parameterization over the mesh. A coarse tensor product B-spline surface is then fit to the spring grid. The error between the fitted surface and the spring grid is represented as a displacement map [Coo84]. The reason for using this hybrid B-spline and displacement map representation is that, given suitable parameters in the fitting process, the B-spline surface captures the overall tablet shape, and the displacement map captures the inscribed marks. It is the latter that interests us in this application.



(a)



(b)



(c)

Figure 5. The tablet shown in the photograph in (a) was unwrapped using the diagram shown in (b). The resulting computer rendering is shown in (c). The patch (labeled Δ in (b)) on the right of the obverse is repeated but rotated 180 (and labeled ∇) on the right of the reverse, because the writing spills onto it from both the obverse and reverse sides; likewise, the topmost patch (\clubsuit) is repeated as the bottommost, without rotation. These duplicated sides patches allow text running over from the obverse and reverse sides to be read at a glance. The text in (c) has been rendered using a combination of curvature coloring, accessibility coloring, and Phong shading, as described in the paper. The inscriptions to the right of the dotted line may be erased (in Photoshop, for example), because they are duplicated.

Since we cannot fit an entire tablet using a single patch, we treat the tablets as approximating a rectangular box, and we fit one patch to each side. This effectively unfolds the tablet. Sides where the text runs over from both the obverse and reverse can be repeated (see Figure 5).

By discarding the B-spline surface and rendering the displacement map, the text appears unwrapped, so that all of the script is visible at once (as seen in Figure 5(c)).

Visualizing

The third and final step of our pipeline involves visualizing the inscriptions. The visualizations involve converting the displacement maps into dense triangle meshes, estimating normals at the vertices, accessibility and curvature coloring the vertices, and finally, rendering with Phong shading

A displacement map is converted to a triangle mesh by first dicing a plane into a dense rectangular mesh, so that each pixel in the displacement map image lies on a vertex of the rectangle mesh. Each vertex is then

moved up or down from the plane by an amount proportional to the displacement value of the corresponding pixel. Each tiny rectangle is then bisected diagonally to yield two triangles.

The resulting geometry may be used to calculate approximate normals at each vertex, which are needed for renderings. To find the normal of a vertex, we scale the normal of each triangle that incorporates that vertex by the triangle's area. We sum the area-scaled normals for all such triangles and we divide by the total area of all triangles that incorporate that vertex.

At this point, we *could* now render the newfound triangle mesh and normals with Phong lighting, but ambiguities resulting from incisions aligned with the light direction would persist. To reduce these ambiguities and visually enhance the shape of the inscriptions, we instead artificially color the mesh's vertices, using accessibility coloring and curvature coloring, individually or in combination.

Accessibility coloring involves coloring each point on a mesh according to the maximum size of a probe sphere that can be placed at some fixed offset above the point without intersecting the mesh [Mil94]. Using this technique, narrow crevices are darkly colored, where at most a small sphere can fit in them, while plateaus and mountaintops are lighter in color, where a much larger probe sphere meets the criterion. The offset improves the appearance of the result, since with zero offset, all vertices and edges associated with (even slightly) concave parts of the mesh would be colored black.

When dealing with dense triangle meshes, we compute accessibility only at vertices, and we approximate the accessibility at a vertex by determining if other vertices fall inside a sphere associated with that vertex. We then map this accessibility to color, and we bilinearly interpolate these colors across triangles.

More formally,

```

For each vertex  $v \in \text{verts}(\text{surface})$ , do:
  Let real  $r = 0$  // Radius
  do:
    Let  $r = r + \text{epsilon}$ 
    Let point  $c = v + \text{NormalAt}(v) * (r + \text{offset})$ 
    // Next find the closest point to  $c$ , excluding  $v$ 
    Let point  $p = \text{Nearest}(\text{verts}(\text{surface}) - v, c)$ 
    While  $\text{Distance}(c, p) > r$ 
    Let  $r = r - \text{epsilon}$ 
  SetColor( $v$ ,  $\text{power}(r / \text{maxRadius}, \text{gamma})$ )

```

In the pseudocode above, *offset* is the distance that the probe sphere is placed above the vertex (see Figure 6). The last line sets the color of a vertex by raising the maximum radius found to the power *gamma*. This nonlinear mapping of radii to colors distributes the colors in a more even and visually appealing manner.

Alternatively, we can color a surface according to curvature. To determine curvature coloring for a surface defined by a mesh, the mesh's curvature at each vertex is approximated using other vertices in its vicinity. Vertices with more absolute curvature (upwards or downwards), such as those in the bottoms of inscribed marks, receive darker coloring. To be precise,

```

For each vertex  $v$  on the surface, do:
  Let vector  $\text{sum} = (0, 0, 0)$ 
  For each vertex  $w \in \text{Neighborhood}(v)$ , do
    Let  $\text{sum} = \text{sum} + \text{NormalAt}(w)$ 
  Let  $\text{sum} = \text{sum} / \text{CountNeighbors}(v)$ 
  Let  $\text{Curv} = 1 - \text{dot}(\text{NormalAt}(v), \text{sum})$ 
  Let  $\text{RawColor} = 1 - \text{sqrt}((2 - \text{Curv}) * \text{Curv})$ 
  SetColor( $v$ ,  $\text{power}(\text{RawColor}, \text{gamma})$ )

```

To combine the output of the two methods, we take the product of their gamma-mapped intensities.

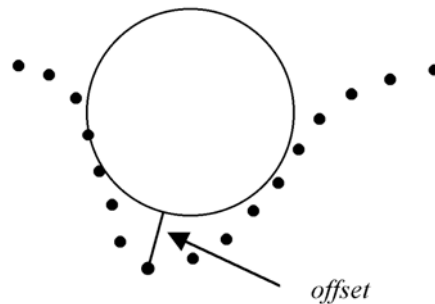
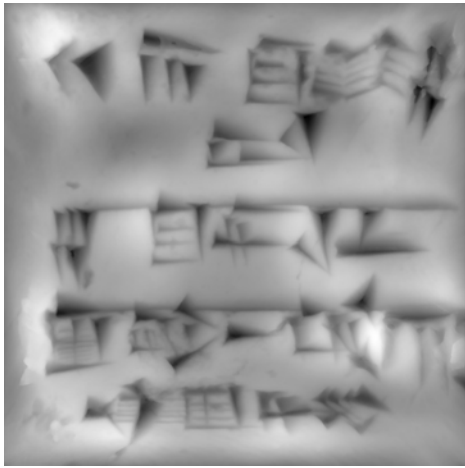
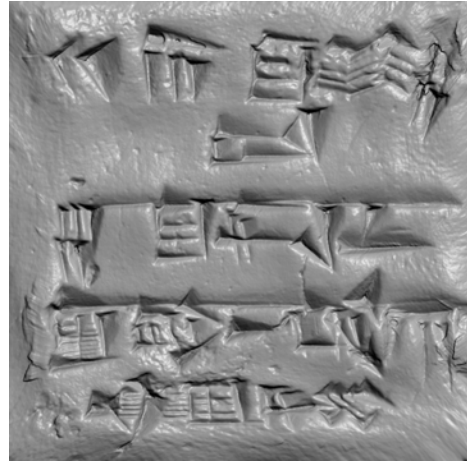


Figure 6. To accessibility-color a vertex (shown above as a dot), we must find the largest sphere that can fit at distance *offset* away along the normal assigned to that vertex.



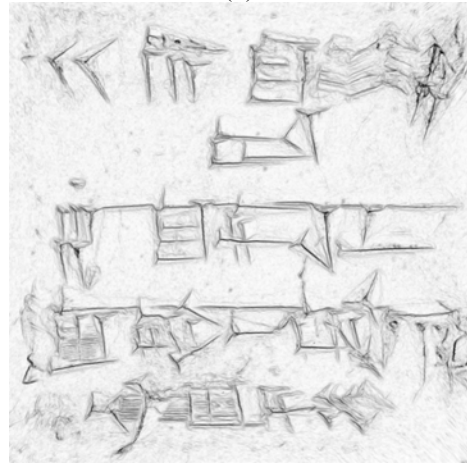
(a)



(b)



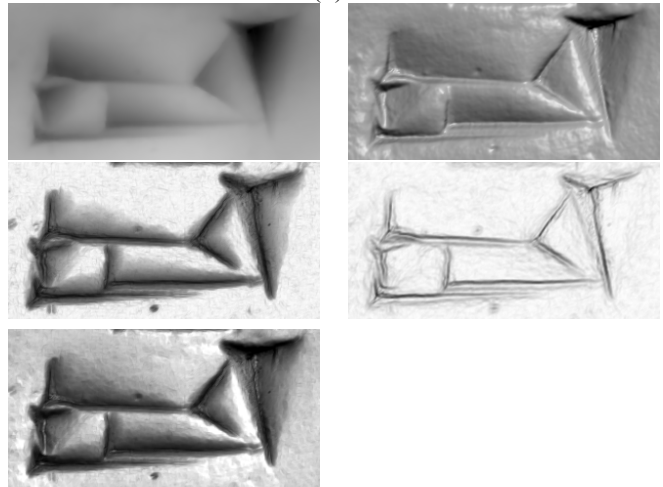
(c)



(d)



(e)



(f)

Figure 7. Image (a) is the unwrapped displacement map. In image (b), the same displacement map is directionally lit with a raking light disposed at the upper right; image (c) has only accessibility coloring; (d) has only curvature coloring; (e) is the product of the previous three, being accessibility colored and curvature colored and it is also illuminated with a raking light. Images (f) magnify the upper centers of (a)-(e). The maximum sphere radius in all accessibility colorings used for these examples is 5% of the width of the model; the sphere is offset by 0.1% of the model width. The composite image (e) is the product of curvature and accessibility values that are first gamma-mapped by 3 and 0.4, respectively. Images (c) and (d) were also gamma-mapped for visualization, but with different values.

Results and Discussion

Examples of our pipeline may be seen in Figures 1(d), 5(c), 7(e), 9, and 10(d). In our opinion, the combination of curvature coloring, accessibility coloring, and raking light yields the best enhancement of the inscriptions. By comparison with Figures 1(a), 2(a), and 5(a), each mark is clearly visible and readable in these figures. We have shown these figures to several cuneiform scholars, who agree. In fact, the visualization on Figure 7(e) sparked a lively debate on the meaning of this particular tablet. The unwrapped inscriptions are visually comprehensible from a single image because darkening by accessibility coloring indicates their depths and curvature coloring enhances the 2D shapes of their troughs.

Despite these successes, our pipeline has several limitations. First, there is some geometric error introduced at each step in the process. The raw scans from the scanner are accurate to 50 microns. Provided they are resampled at a resolution of 25 microns or less when merged into a full model and also when creating the spring meshes, the values in the displacement map should be accurate to around 50 microns.

Second and most significantly, there is distortion inherent in flattening curved surfaces onto planes (as seen in Figure 10). Our use of tensor product surfaces to fit tablet surfaces assumes that they are well approximated by a box. To the extent this assumption is untrue, the unwrapped inscriptions appear warped, particularly around the corners of the displacement map image. This distortion may decrease the legibility of individual marks, which were made by pressing a physical stylus into a soft *curved* surface. We don't currently have a quantitative measure for this distortion, but in the worst case, it could conceivably transform the mark into a different word. However we have not seen such severe distortions in our examples.

We might employ an "interrupted" projection onto a plane, like the Interrupted Goode Homolosine Projection for mapping the Earth. As seen in Figure 10(c), such a projection would show the surface split apart with seams occurring along the tablet's "ledger lines" (the lines separating vertically adjacent rows of marks). This projection would more evenly distribute the distortion amongst the unwrapped characters. Achieving this projection would require a way to flatten the coarse B-spline surfaces, retaining their boundary shape. We have not tried this.

To further control distortion, Paraform feature curves may be drawn along the ledger lines in the cuneiform text. These curves are treated as isoparametric lines in the fitting process, thereby forcing the ledger lines to be horizontal in the generated displacement map. This helps reduce the distortion in the output. We used feature curves in a couple places of Figure 10(d) to partially improve the output, but the distortion from tablet curvature is too severe to be fixed.

A third limitation of our pipeline is that large inscriptions pull the coarse B-spline surface (which is an approximation of the tablet's surface) down inside them. As a result, the measured depths of some inscriptions in the displacement map are less than they should be. We would prefer to have the B-spline surface be the original surface of the clay, before the stylus had penetrated it. By using what is known as the *alpha hull* of the mesh, we may better approximate this virgin surface (see Figure 8). The *alpha hull* [Ede92], a generalization of the convex hull, is defined as the complement of all spheres of radius *alpha* that do not intersect the surface. We have not tried this either.

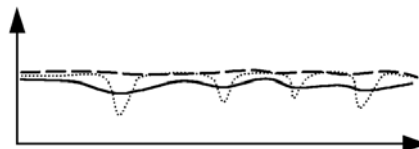


Figure 8. The true surface is drawn with dotted curve; it is coarsely approximated by a B-spline surface, shown with the solid curve. But to approximate the virgin (unmarked) surface, we really want the alpha hull, drawn with the dashed curve. The difference between this curve and the true surface curves represents the inscriptions better.

The fourth limitation is the inefficiency of the process of virtually repairing broken tablets by interactively adding geometry to the scanned 3D model. The process might be aided by using symmetries in a tablet's shape to approximate the missing geometry. A more comprehensive solution would permit arbitrarily shaped patches to be input and output, rather than forcing them to be rectangular. An energy function could be defined for the mapping and minimized through an optimization process [Ben91].

Conclusions and Future Work

We have described methods for unwrapping and visualizing inscriptions on cuneiform tablets. These techniques should be applicable to other inscribed objects and artifacts as well. For example, the bas-reliefs on Trajan's column (in Rome, Italy), which has many

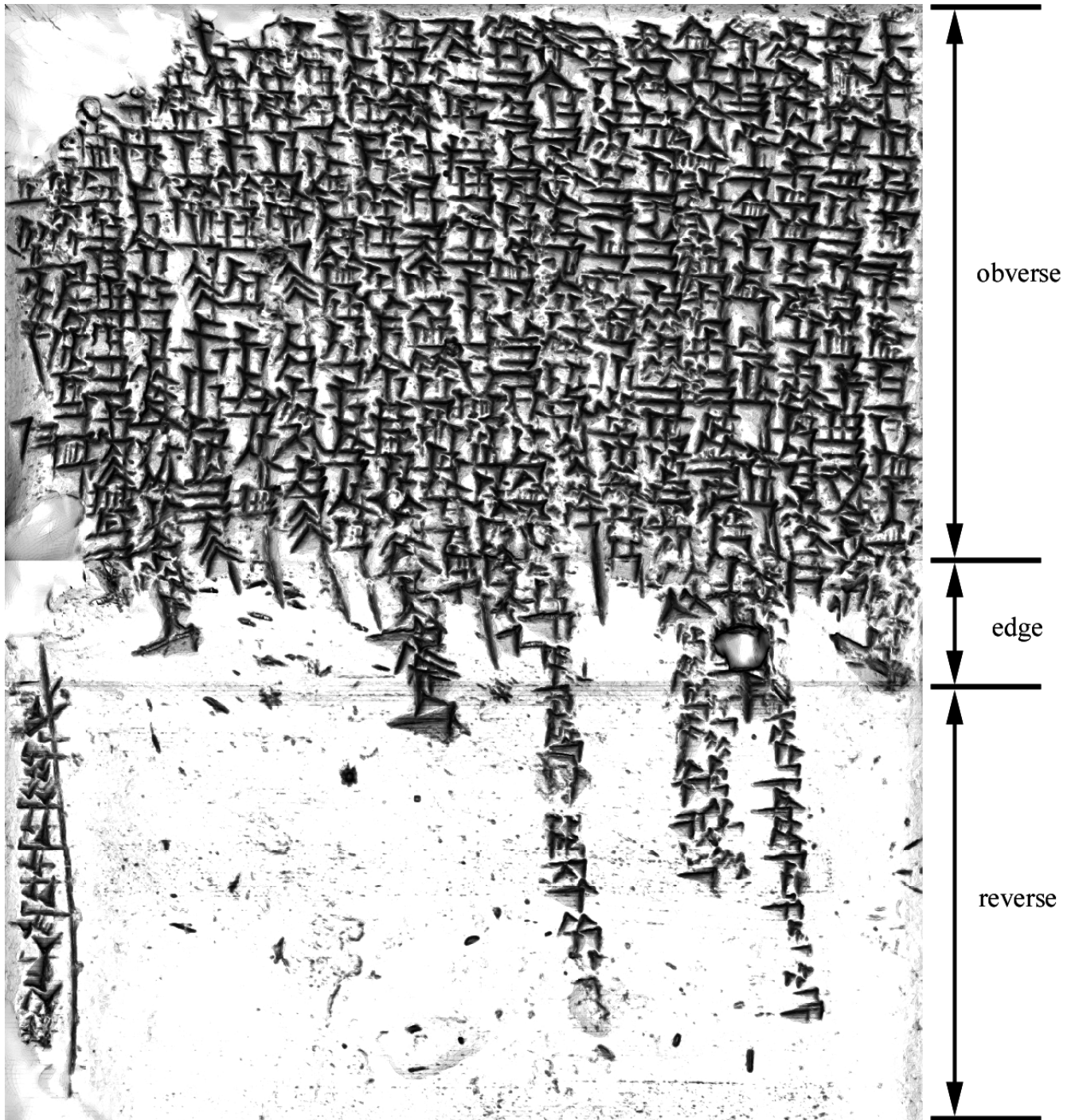


Figure 9. The inscriptions shown above, unwrapped and visualized using our pipeline, would be difficult to depict using photography. They would also be difficult to transcribe, due to the large number of small marks.

small carvings over its surface, might be better visualized by employing the unwrapping and coloring approach presented here.

We envision several directions for future work. One is towards inexpensive high-resolution 3D scanners, which are currently very costly, because the number of

extant tablets is large (over 100,000), and archiving projects are typically not highly funded. A passive vision method, such as shape-from-stereo on a flatbed scanner [Sch00], might be capable of capturing the detail with sufficient resolution, and it would be inexpensive enough to enable almost anyone with ac-

cess to cuneiform tablets to create high-resolution models. However, we have experimented with this technique and were unable to reliably detect features, due to noise or the lack of sufficient distinguishable texture for the stereo algorithm to lock onto.

An improvement to the unwrapping phase would be to automatically compute the boundary curves of a patch on a 3D tablet model rather than requiring that a user draw them. However, such an automatic process would be feasible only if it included a method for minimizing distortion of the unwrapped tablet or for isolating the distortion to unimportant parts of the tablet.

Finally, having 3D models of the tablets may facilitate accurate cuneiform optical character recognition (OCR), which has been a dream of researchers. (Several papers [Dem94] have been written about fast automatic cuneiform character recognition using optical correlators, but their emphasis is on speed rather than automation.) For written text, abstract symbols (see Figure 2(b)) derived from OCR may be superior to our non-photorealistic rendering of displacement maps extracted from a 3D model.

Acknowledgements

We offer our thanks to Renee Kovacs, who suggested the project and provided us with cuneiform tablets to scan. We would also like to thank Szymon Rusinkiewicz and Matt Pharr for providing the accessibility and curvature coloring code that was used in producing the examples shown above. Also, we are grateful to Marc Rioux and Luc Cournoyer at the National Research Council of Canada for scanning a tablet for us, and to Paraform Corporation, which provided us with a version of its surfacing program to extract displacement maps of the tablet models. Lastly, Robert Englund provided valuable background information and was very encouraging of our research.

References

- [Ben91] Chakib Bennis, Jean-Marc Vézien, Gérard Iglésias, André Gagalowicz. "Piecewise surface flattening for non-distorted texture mapping." *Computer Graphics, Proceedings of SIGGRAPH 91*, August 1991, pp. 237-246.
- [Coo84] Robert L. Cook. "Shade Trees." *Computer Graphics, Proceedings of SIGGRAPH '84*, pp. 223-231, July 1984.
- [Cur01] Brian Curless. "Volumetric Range Image Processing." <http://graphics.stanford.edu/software/vrip/>
- [Dem94] N. Demoli, U. Dahms, H. Gruber, and G.K. Wernicke. "Use of a multifunctional extended optical correlator for cuneiform inscription analysis." *Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 2297, July 1994, pp. 278-287.
- [Ede92] Herbert Edelsbrunner and Ernst P. Mücke. "Three-dimensional alpha shapes." In A. Kaufman and W.E. Lorensen, editors, *Proceedings of 1992 Workshop on Volume Visualization*, October 1992, pp. 75-82.
- [Lev96] Marc Levoy and Patrick Hanrahan. "Light Field Rendering." *Computer Graphics, Proceedings of SIGGRAPH 96*, August 1996, pp. 31-41.
- [Kri96] Venkat Krishnamurthy and Marc Levoy. "Fitting Smooth Surfaces to Dense Polygon Meshes." *Computer Graphics, Proceedings of SIGGRAPH 96*, August 1996, pp. 313-320.
- [Mal01] Thomas Malzbender, Dan Gelb, and Hans Wolters. "Polynomial Texture maps." *Computer Graphics, Proceedings of SIGGRAPH 2001*, August 2001, pp. 313-320.
- [Mil94] Gavin S. P. Miller. "Efficient Algorithms for Local and Global Accessibility Shading." *Computer Graphics, Proceedings of SIGGRAPH 94*, August 1994, pp. 319-326.
- [Nis93] Hans J. Nissen, Peter Damerow, and Robert K. Englund. *Archaic Bookkeeping*. University of Chicago Press. 1993, pp. 45-46.
- [NRC] "National Research Council of Canada." <http://www.vit.iit.nrc.ca/>
- [Rus01] Holly Rushmeier, Laurent Balmelli, and Fausto Bernardini. "Horizon Map Capture." *Computer Graphics Forum*, vol.20, no.3, September 2001, pp. C/85-94.
- [Sch00] Richard Schubert. "Using a Flatbed Scanner as a Stereoscopic Near-Field Camera." *IEEE Computer Graphics & Applications*, vol. 20, no. 2, March/April 2000, pp. 38-45.



Figure 10. This example shows a highly curved tablet in top and side views (a) and (b), respectively. As a result of the high curvature, the unwrapped text, seen in (d), becomes severely warped. A solution to this problem might involve cutting the rows into a number of split sections, as in (c). In cartography, planar maps of the Earth that are split apart like this are termed “interrupted.”