

Visualization and Extraction of Carvings for Heritage Conservation

Lawonn, Kai; Trostmann, Erik; Preim, Bernhard; Hildebrandt, Klaus

DOI

[10.1109/tvcg.2016.2598603](https://doi.org/10.1109/tvcg.2016.2598603)

Publication date

2017

Document Version

Accepted author manuscript

Published in

IEEE Transactions on Visualization and Computer Graphics

Citation (APA)

Lawonn, K., Trostmann, E., Preim, B., & Hildebrandt, K. (2017). Visualization and Extraction of Carvings for Heritage Conservation. *IEEE Transactions on Visualization and Computer Graphics*, 23(1), 801-810. <https://doi.org/10.1109/tvcg.2016.2598603>

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

Visualization and Extraction of Carvings for Heritage Conservation

Kai Lawonn, Erik Trostmann, Bernhard Preim, and Klaus Hildebrandt

Abstract— We present novel techniques for visualizing, illustrating, analyzing, and generating carvings in surfaces. In particular, we consider the carvings in the plaster of the cloister of the Magdeburg cathedral, which dates to the 13th century. Due to aging and weathering, the carvings have flattened. Historians and restorers are highly interested in using digitalization techniques to analyze carvings in historic artifacts and monuments and to get impressions and illustrations of their original shape and appearance. Moreover, museums and churches are interested in such illustrations for presenting them to visitors. The techniques that we propose allow for detecting, selecting, and visualizing carving structures. In addition, we introduce an example-based method for generating carvings. The resulting tool, which integrates all techniques, was evaluated by three experienced restorers to assess the usefulness and applicability. Furthermore, we compared our approach with exaggerated shading and other state-of-the-art methods.

Index Terms—Feature extraction, heritage preservation, Frangi filter, surface analysis, feature filtering

1 INTRODUCTION

In the 13th century, artists have carved illustrations of the episodes from the church history and depictions of important persons into the plaster of cathedrals and churches. One prominent example are the carvings on the walls of the cloister of the Cathedral of Saints Catherine and Maurice in Madgeburg. Due to the open structure of the cloister, these carvings were subjected to weathering. Nowadays, the illustrations in parts of the plaster are hard to recognize and many structures are only visible with extreme side light. Therefore, restorers and historians are in need of a software tool that allows to analyze digital copies of the plasters, and to explore and edit the carvings virtually. In addition, virtual images visualizing the carvings and illustrations of the original shape of the carvings are attractive for visitors of the historic site.

In this paper, we propose techniques addressing the technical challenges that have to be solved for creating such a tool and we present an evaluation of the associated software by the relevant target group.

We want to contrast our approach from methods for general sunken-relief extraction. The carvings we consider are special since the structures are canal-like and flat. Aging and weathering have additionally flattened the structures so that they can barely be perceived with the naked eye. Furthermore, the surface of the plaster exhibits irregular structures and the data contains noise. The amplitude of the noise and the height of the structures almost match the depth of the carvings. Both factors complicate the extraction of the carving structures.

To enable the processing of sunken-relief extraction, we first digitalize the plasters and carvings using a passive stereo vision system, resulting in a triangle mesh approximating the surface of the plaster. For analyzing the digitized plaster, we developed a detection filter for carvings in surfaces that searches for parts with a distribution of principal curvatures that exhibits characteristic for carvings, i.e., locally cylinder-like structures. This approach is based on the Frangi filter [9] from medical image processing, which is used for detecting vessel-like structures in images. We adapt this method to detect carvings in 3D surfaces. Our method is specifically designed for detecting canal-like structures and is robust against noise. For filtering the set of detected

carving structures and selecting the most salient structures, we propose a scheme that combines connected component analysis and an importance measure for the saliency of the carvings. Finally, visualization techniques are employed to emphasize the detected carving structures, see Figure 1. The software implementing these methods has been tested by domain experts in a user study that highlights the effectiveness and utility of our approach. Our experiments demonstrate that state-of-the-art methods for relief extraction, as well as feature line extraction and exaggerated shading, fail on the type of data we are considering. In summary, we make the following contributions:

- An algorithm for detecting carving structures in surfaces based on an adaptation of the Frangi filter and a scheme for ordering connected components of the detected carving structures by saliency
- A framework for interactive exploration of carving structures in surfaces that allows for selecting and deselecting structures, adding new carvings and visually highlighting them

2 RELATED WORK

For the illustration of carvings, we analyze the field of illustrative visualization. One relevant topic in this field is the depiction of strong features via lines.

Feature Lines Methods for feature line extraction can be divided into view-dependent and view-independent approaches. Although, for carving extraction it is important to illustrate all features independent of the camera position, it is worth mentioning view-dependent approaches as well. Ridge and valley lines have the property that every point is an extremum of the principal curvature on the corresponding curvature line [15, 29]. The computation of these lines is delicate since it involves estimating third-order derivatives on the surfaces. Prominent approaches use surface fitting [28, 39] or a combination of discrete curvature approximation and smoothing techniques [14]. A technique for view-dependent feature line detection was introduced by DeCarlo et al. [7]. They extended the contour definition to determine salient features. Another view-dependent feature line approach was presented by Xie et al. [38]. Their method depends on the underlying surface shading and allows for compensating noise by adding light sources. This approach was later improved [41]. A method that combines ridges and valleys with a view-dependent approach was proposed by Judd et al. [16]. They defined view-dependent curvature quantities to apply their technique. Kolomenkin et al. [18] proposed a view-independent approach to determine the transitions from flat regions to ridges. Their approach can be used to enhance small features on archeological objects. Zhang et al. [42] introduced Laplacian lines, which are used to illustrate prominent features. The Laplacian of the surface normal field is pre-computed and the result is used for illustrating the features. Weinkauff and Günther [37] used the Morse–Smale complex of a curvature-based indicator function to construct networks of feature-lines. Robustness against noise is improved by simplifying the Morse-

-
- Kai Lawonn is with the University of Koblenz-Landau, Germany. E-mail: lawonn@uni-koblenz.de.
 - Erik Trostmann is with Fraunhofer Institute for Factory Operation and Automation IFF, Germany. E-mail: erik.trostmann@iff.fraunhofer.de.
 - Bernhard Preim is with the University of Magdeburg, Germany. E-mail: preim@isg.cs.uni-magdeburg.de.
 - Klaus Hildebrandt is with Delft University of Technology, The Netherlands. E-mail: k.a.hildebrandt@tudelft.nl.

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxx/TVCG.201x.xxxxxx



Fig. 1. On the left the original model with standard headlight. On the right the same dataset with our visualization technique.

Smale complex using the concept of persistence [8]. Rieck et al. [30] developed persistence-based filters and topological signatures for the multivariate analysis of data and applied the techniques to the analysis of cultural heritage artifacts. While the method performs very well on artificial data, the processing of 3D-scans requires manual segmentation of the digital object and adjustment of parameters (see Sections 6.2.1 and 6.2.2 in [30]). A technique to illustrate surface features and enhance the spatial impression with lines was presented by Lawonn et al. [21]. For a recent overview on the extraction of feature lines, we refer to [22].

Feature-based shading A technique for enhancing surface features by modifying the surface normals was presented by Cignoni et al. [5]. Their idea is to smooth the normals and to compare them with the original ones: the larger the differences, the more salient the underlying geometric feature. Another approach for enhancing geometric features by modifying the surface normals was introduced by Rusinkiewicz et al. [32]. They iteratively smooth the surface normals and store the results. Afterwards, the stored sequences of normals are combined to determine the shading. Lee et al. [23] used an approach that automatically determines lighting positions to enhance features. Here, the surface is first segmented using a curvature-based watershed method. These patches are used for the placements of lights. Vergne et al. [35] introduced a local shape descriptor that enhances salient features. They warped the incoming lighting in order to increase the view-dependent surface curvature information. This leads to a perceptually enhanced surface shape. For analyzing salient features on archaeological surfaces, Kolomenkin et al. [20] presented an approach to automatically process scanned artifacts. They defined a *prominent field*, which is a smooth field with certain properties at surface features to enhance geometrical features. Vergne et al. [36] presented a technique called *Radiance Scaling* that improves the classical shading by enhancing features on the surface. They scaled light intensities based on curvature and material characteristics. An exaggeration tech-

nique to enhance relief structures was proposed by Miao et al. [25]. This approach is based on three features maps comprising local height, normal difference and mean curvature. Furthermore, the normals are adjusted to gain an enhanced rendering of the surface. Ammann et al. [2] introduced an approach that analyzes several surface features per vertex. These features are used to combine different shading techniques to enhance salient structures on the surface. As this approach enhances all features, it may be prone to noise and even highlight these undesired regions. For the illustration of complex archaeological artifacts, Gilboa et al. [12] reported on the development of computerized automatic systems. Through automatic feature detection, they highlighted salient areas as line drawings, colored images or enhanced 3D models. Sharma and Kumar [33] presented a way to investigate artifacts with machine-learning techniques. Their system employs the user's preferred visualization settings to generate a reasonable result. Several techniques that depend on various vertex features, e.g., curvature, ambient occlusion factor, are applied to gain an appropriate shading result.

Relief extraction and generation There is a large body of literature on relief extraction and generation and we would like to point to the survey by Kerber et al. [17] for an overview. A general introduction to 3D shape analysis for archaeological surfaces was provided by Tal [34]. Closest related to our work is the relief extraction scheme by Zatzarinni et al. [40]. They represented the surface containing the relief as a height field over a base surface. The relief is found through an analysis of the height field. In contrast to our scheme, their method aims at reliefs with enough height (or depth). In the case of the carvings considered here, the ratio between depth of the reliefs, amplitude of noise and natural structures in the surface and the (very low, but present) curvature of the surface make it difficult for their method to extract the carvings.

3 DATA ACQUISITION

The incisions cover an area of 37 m by 5 m outside on a wall to the courtyard of the Magdeburg cathedral. Because of current restoration work, we could use an existing scaffold for data capturing. Varying lighting conditions and unavoidable vibrations of the scaffold require a rapid measurement process. We decided to use a passive stereo vision system because it requires only a pair of synchronously captured images.

The measurement area was chosen based on the required resolution of less than 0.1 mm. Using two grayscale cameras with a resolution of 2048×2048 pixels the system was designed to capture an area of about 170 mm by 170 mm, see Figure 2. The calculation of calibration parameters was accomplished with the OpenCV library. We used a precisely measured calibration plane with circular marks. The centers of the images of the circular marks can be localized with high accuracy. This leads to a reliable estimation of the intrinsic and extrinsic camera parameters. Testing the spatial intersection of the calibrated stereo vision system using a stochastic pattern on a plane surface (calibration standard) results in residual errors up to a maximum of 0.02 mm, which is acceptable for this application.

To compute pairs of corresponding points in both images of one capture, we start with an evenly distributed field of points on one image. Then the task is to find the best correlation between a squared mask around that point on image one and the projection of this mask along the epipolar line on image two. To save computation time and to increase robustness, this line is trimmed by taking into account the depth of field (sharpness). Because of the slightly different viewing angles and varying imaging characteristics, the correlation algorithm has to be robust against differences in dynamics and the average intensity. Along with other approaches, the normalized cross-correlation introduced by Moravec [27] gives reliable results. It is invariant against additive and multiplicative intensity changes in the search window. The spatial intersection for all rays of corresponding points for one capture leads to a resulting 3D point cloud of up to 4 million points.

To reconstruct a larger area, we have to combine the results of a number of captures. This is accomplished by manually picking three corresponding 3D points within two patches and a subsequent refinement using the Iterative Closest Point algorithm [3]. Since the

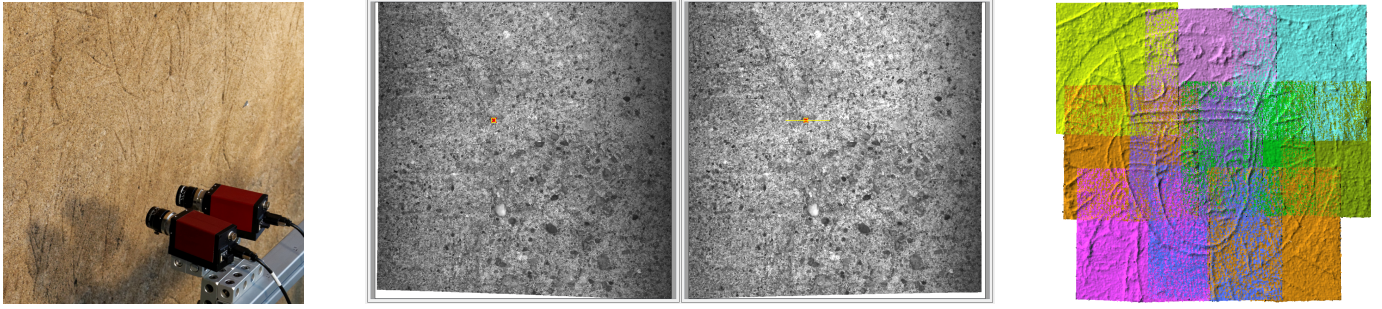


Fig. 2. Stereo camera setup (left) and corresponding images (middle) for the computation of a 3D point cloud (patch). On the right the registration result for nine false-colored patches is shown.

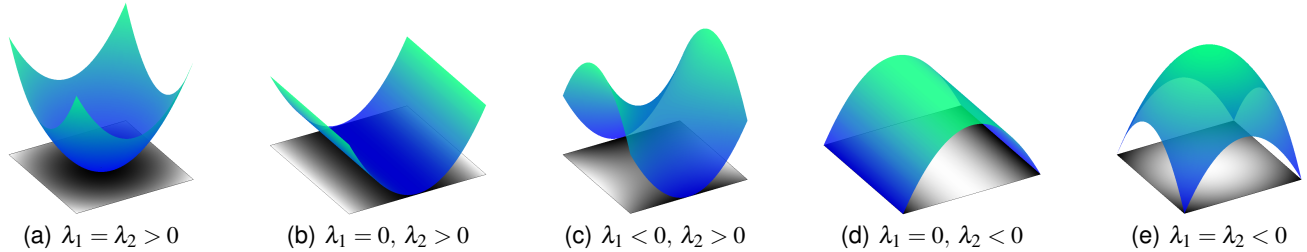


Fig. 3. Different greyscale images are illustrated with their corresponding height surfaces. The height surfaces' principal curvatures are listed below. Note that for $\lambda_1 = \lambda_2 = 0$ the image is constant and the height map is a plane.

patches overlap by about 40%, there are enough unique features in the point set to ensure a robust registration. A drawback of this method is that the overall shape of the wall becomes slightly deformed after registration of a large number of patches. This is not affecting the visual perception of the result, but a comparison to point data captured by 3D laser scanning shows a significant deviation of several centimeters on a wall with 2.5 m in length.

4 REQUIREMENT ANALYSIS

Based on a literature search and discussions with domain experts, we devised the following list of requirements:

- A method for detecting carving structures
- Interaction techniques that allow the user to modify the set of carving structures, e.g., erasing wrongly detected carvings or adding carvings
- A visualization technique that enhances the carvings

In the real-world plaster, the carving can barely be perceived with the naked eye. Only under special lighting conditions, the restorers can get an impression of the carving structures. To circumvent these difficulties, we decided to digitalize the whole plaster with high resolution and to allow the domain experts to process the data such that the carvings can be illustrated. Since with standard visualization of the scan data, it is still difficult to see carvings, the experts asked for visualization methods that can emphasize the carvings. The experts are interested in interactively modifying and controlling the visualization, e.g., by selecting additional structures to be emphasized or manually removing falsely detected carving structures.

5 EXTRACTION AND GENERATION OF CARVINGS

For the analysis and processing of carvings in plasters, we developed schemes for carve extraction and ranking as well as methods for generating new carvings.

5.1 Carve Extraction and Ranking

Though the plaster that we want to analyze is attached to one wall of the cloister, it is not planar. Therefore, the carvings cannot be described by a height field over the plane. We treat the digitalized plaster as a curved surface, in which we want to detect the carvings.

Curvature computation For detecting carvings, we analyze the distribution of the curvatures over the surface. Since we are working with triangle meshes, we need to compute the principal curvatures and curvature directions of triangle meshes. Following Rusinkiewicz [31], we first compute a discrete shape operator in every triangle, which measures the deviation of the normals across the edges of the triangle. Then, a discrete shape operator at the vertices is constructed by transforming and averaging the shape operators of the triangles. The eigenvalues of the shape operator are the principle curvatures λ_1, λ_2 , which are ordered such that $|\lambda_1| \leq |\lambda_2|$. When computing the discrete shape operator (in particular for noisy data), it is recommended to smooth the surface beforehand. We used bi-Laplacian smoothing, which results in a robust curvature computation in our experiments. An alternative to smoothing the surface is to use larger neighborhoods for the curvature computation, see [6, 13].

Frangi filter for surfaces The carving structures we want to extract are “small canals” in the shape. At the bottom of each canal, the geometry equals that of a round tube or a cylinder. This means one principal curvature vanishes and the other one is approximately one over the radius of the cross section of the carving, i.e., $\lambda_1 = 0$ and $\lambda_2 \neq 0$. Figure 3 shows examples of surfaces and their principal curvatures. Since the carvings are engraved, the non-vanishing principal curvatures of all carvings have the same sign. Whether the sign is positive or negative depends on the orientation of the surface. We choose the orientation such that the surface normals point outwards (out of the wall) and the curvature values are positive. For detecting the carving (or vessel) structures, we use, analogous to the Frangi filter [9], the quantities $\mathcal{S} := \sqrt{\lambda_1^2 + \lambda_2^2}$ and $\mathcal{R}_{\mathcal{B}} := \lambda_1 / \lambda_2$ and define the *vesselness*

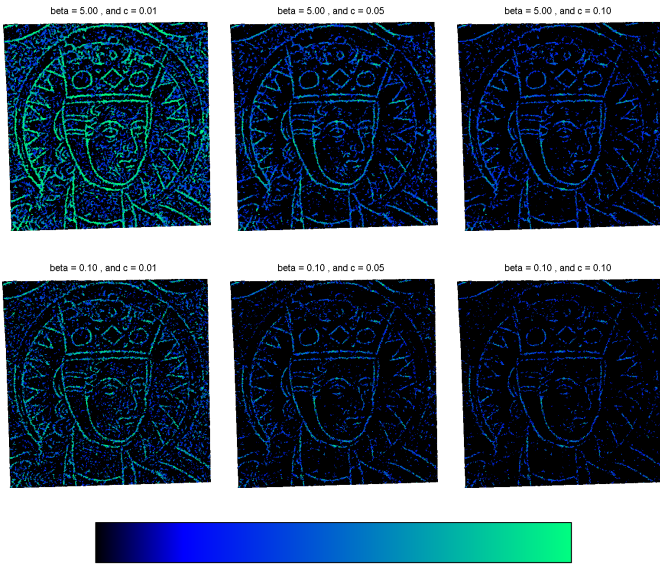


Fig. 4. Results of the vesselness measure with different parameter settings. The colormap is shown at the bottom, where values range linearly from 0 to 1, which corresponds to black to cyan, respectively.

measure as

$$\mathcal{V}_0(s) = \begin{cases} 0 & \text{if } \lambda_2 < 0, \\ \exp\left(-\frac{\mathcal{R}_s^2}{2\beta^2}\right) \cdot \left(1 - \exp\left(-\frac{\mathcal{I}_s^2}{2c^2}\right)\right). & \end{cases} \quad (1)$$

In the discrete setting, we compute the principal curvatures at the vertices of the mesh and use Eq. 1 to get a value of the vesselness measure at every vertex. Linear interpolation in the triangles yields a scalar field representing the vesselness measure on the triangle mesh. We used a blue to cyan colormap to display the influence of the parameters β and c . To depict the threshold, we additionally displayed all values that are smaller than the threshold in black.

Parameter adjustment The vesselness measure, Eq. 1, involves two parameters, β and c . To help the users in choosing values for the two parameters, we create preview galleries. The users can choose preferred parameter settings by selecting an image. An example of an image gallery is shown in Figure 4. After a preview image has been selected, the corresponding parameter setting can be modified to fine-tune the result. Our user-interface for this step is inspired by the work of Mönch et al. [26]. When pressing the right mouse button, the user can alter the two parameters by mouse movements. Horizontal movements affect only one of the parameters and vertical movement the other parameter. The benefit of this interaction scheme is that the user can maintain focus on the image while changing the values of parameters.

Ranking the carvings Once the parameters β and c are set, the vesselness measure can be computed. For extracting the carvings, we use a connected component analysis, which clusters the vertices into components. Starting with an arbitrary vertex with $\mathcal{V} > 0$, we iteratively add all neighbors in the one-ring with $\mathcal{V} > 0$ to the same component until every vertex with positive vesselness measure is assigned to a component. For any component C_i , we define its saliency as

$$\mathcal{V}_{C_i} := \sum_{\mathbf{p} \in C_i} \mathcal{A}_V(\mathbf{p}) \cdot \mathcal{V}(\mathbf{p}), \quad (2)$$

where $\mathcal{A}_V(\mathbf{p})$ denotes the Voronoi area of vertex \mathbf{p} , see [24]. Finally, we rank the components by their saliency.



Fig. 5. Selection of carving structures and visualization with emphasized carvings. A standard rendering of the captured plaster is shown in (a). To help the user in selecting carving structures, our tool displays connected components of the detected carvings. Each component receives a color reflecting its saliency (b). The colormap is shown at the bottom, where values range from gray (minimum value) to red (maximum value). The minimum, maximum value correspond to the minimum, maximum saliency (cf. Eq. 2) in the set of detected components. The values in between are linearly interpolated. The user can activate the n components with highest saliency value. The result for $n = 4$ is shown in (c). Activated components are shown in cyan. To fine-tune the result, the user can further modify the selection set. In the final rendering, the selected carvings are emphasized (d).

Activating the carvings After connected components in the carvings structures are determined and the components are ranked by saliency, the user can use this information for selecting the components that will be enhanced in the rendering. The selection proceeds in two phases, first the user specifies a number n of components to be selected. Our tool then selects the n component with highest saliency value. In the second phase, the user has full control over the selection and can manually add components to and remove components from the selection. To help the user in modifying the selection, our tool assigns colors to all components. Selected components are displayed in cyan and non-selected components receive a color that reflects their saliency value (cf. Eq. 2). The colormap we use for the non-selected components ranges from dark red to gray over orange, see Figure 5. Parts of the surface that do not belong to a carving structure receive a dark color. To further assist the user during the manual modification of the selection set, already activated components will be highlighted in red by a mouse over to indicate that clicking will delete the selection and not activated components will be colored in green to indicate that clicking will activate them.

Visualizing the carvings To emphasize the selected carving structures in the rendering, the shading of the selected components is modi-

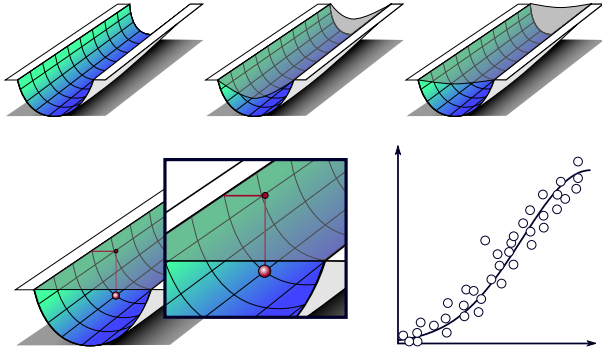


Fig. 6. First, the carving is smoothed and the boundary is fixed (top row). Afterwards, every point on the smoothed carving is assigned the height distance to the original carving. The values are ordered according to their height and plotted. Finally an exponential function is fitted to the ordered height data.

fied. The idea is to use the vesselness scalar field to weight the lighting in order to enhance the carving cues. Let \mathbf{l} denote the light vector. Then the shading is computed using $s = \langle \mathbf{l}, \mathbf{n} \rangle^{1+\alpha \cdot \mathcal{V}_0}$, where α expresses how strongly the carving cues are emphasized. We used $\alpha = 2$ for all results. Using a significantly higher value of α would result in stronger, but unnatural looking carvings. In addition to emphasizing the selected carvings, our tool allows users to deemphasize the non-selected structures. This is achieved by isotropically smoothing the normals of the surface (using 10 iterations of Laplace smoothing) and using the smoothed normal vectors to compute the shading everywhere except for the selected carving structures.

5.2 Carve Generation

In addition to visualizing the carvings, we want to provide a tool for generating carving structures. For example, to help with the recreation of lost parts of an illustration by adding new virtual carvings. Since editing the surface itself is a challenging problem which requires sophisticated modeling tools [10], we decided to preserve the original surface and add virtual carvings overlaying the original plaster. This is realized by modifying the surface normals in the areas where new carvings are added. To specify the location and width of the virtual carvings the user is provided with a brush tool. The size of the brush specifies the width of the carving. The result of the brushing is stored in a scalar field on the surface. Initially, this scalar field is 0 for every vertex. Whenever the brush is used, every vertex inside the brushed region in screen space is assigned a new scalar value. Denoting the distance of a vertex to the center of the brush by $dist$, we assign the value $r_n = \frac{r-dist}{r}$, which ensures that the midpoint assumes the maximum value. Whenever the user draws over existing brushed vertices, the already stored values are compared with the potential new values. The greater value is then assigned to the vertex. We use a scalar function $f(r_n)$ to determine the modified normals of the surface. The function serves as a desired height (or offset in normal direction) over the original surface. The height value at any point depends only on the value r_n assigned to the point during brushing. The function f depends on the carvings observed in the data. The goal is to make the newly created carvings look similar in style to the ones that are already in the data.

Height Field Calculation To obtain the generative model for new carvings, we analyze the existing carvings, which are extracted using the scheme described in Sec. 5.1. The generative model represents the carvings as a height field over a smoothed version of the surface. The profile of the height field is analyzed and used to generate new carvings with a similar profile. The first step is to construct a base surface over which the carvings are represented using a height field. For this, we copy the surface and heavily smooth the carvings by applying implicit Laplacian smoothing, i.e., mean curvature flow. The implicit smoothing

is defined by the iteration

$$\mathbf{p}_i = (\mathbf{I} - \mu \mathbf{L}) \mathbf{p}_{i+1}, \quad \mathbf{p} \in \mathcal{V}_\ell, \quad (3)$$

where \mathbf{I} is the identity matrix, \mathbf{L} is the Laplacian matrix, and μ is the weighting factor, which we set to 0.05. The iteration starts with the original vertex positions \mathbf{p}_0 . After each smoothing step, the Euclidean distance between the new and the old positions of the vertices is computed

$$D^{i+1} = \sum_{\mathbf{p} \in \mathcal{V}_\ell} \|\mathbf{p}_{i+1} - \mathbf{p}_i\|.$$

The smoothing flow converges when the quotient D^{i+1}/D^i tends to 1. We stop the smoothing if $D^{i+1}/D^i > 0.99$. In practice, the smoothing process is terminated after 30-50 iterations. During the smoothing, the carvings evolve into their surface normal directions and towards the minimal surface spanned by the constrained boundaries.

We measure the distance of the carving to the minimal surface, see Figure 6. Explicitly, we look for the intersection of the line generated by the point on the smoothed carving and the new normal with the triangles on the original carving. The intersection point and the point on the smoothed carving yield the height distance. In case two or more intersections occur, we assign the shortest distance to the point. Every point of the smoothed carving is assigned a distance, i.e., the height scalar field. For every carving component, we scale the distance to the x -axis in a range of $[0, 0.5]$.

Then a Gaussian distribution $f(x) = a \cdot \exp(-(x-0.5)/c)^2) + d$ in the range $x \in [0, 0.5]$ is fit to the data, see Figure 6 bottom right. The resulting function can be used as the input for the drawings on the surface: $f(r_n)$. This yields the scalar field \mathcal{C} . Based on this scalar field \mathcal{C} , we determine the gradient per triangle. Given a triangle $\Delta = \{i, j, k\}$, the gradient of every triangle is determined by:

$$\nabla \mathbf{g} = (\mathcal{C}_j - \mathcal{C}_i) \frac{(\mathbf{p}_i - \mathbf{p}_k) \perp}{2A_\Delta} + (\mathcal{C}_k - \mathcal{C}_i) \frac{(\mathbf{p}_j - \mathbf{p}_i) \perp}{2A_\Delta}, \quad (4)$$

where A_Δ is the area of the triangle Δ and \perp denotes a counterclockwise rotation by 90° in the plane generated by the triangle, see [4]. The gradient for every vertex is obtained by transforming and averaging the gradients of the triangles. This yields the vertex gradient of the scalar field \mathcal{C} .

The vertex gradients of the scalar field can be used to calculate the new normals of the carving structure. Given is a scalar field \mathcal{S} defined on a plane by $\mathcal{S} = f(x, y)$, $x, y \in \mathbb{R}$. It generates a height surface by setting $z - f(x, y) = 0$. Then, the unit normal vector $\mathbf{n}(x_0, y_0)$ at (x_0, y_0) is

$$\mathbf{n}(x_0, y_0) = \frac{1}{\sqrt{1 + \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}} \cdot \begin{pmatrix} -\frac{\partial f}{\partial x} \\ -\frac{\partial f}{\partial y} \\ 1 \end{pmatrix} \quad (5)$$

This can be simplified by determining the gradient of f : $\nabla \mathbf{g} = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)^T$ then, the normal is

$$\mathbf{n}(x_0, y_0) = \frac{1}{\sqrt{1 + \langle \nabla \mathbf{g}, \nabla \mathbf{g} \rangle}} \cdot \begin{pmatrix} -\nabla \mathbf{g}_x \\ -\nabla \mathbf{g}_y \\ 1 \end{pmatrix}. \quad (6)$$

As we considered the height field on the plane, the normal of this plane is $\mathbf{n}_p = (0, 0, 1)^T$. Therefore, we can determine the angle γ between the plane normal \mathbf{n}_p and the normal determined by the height surface $\mathbf{n}(x_0, y_0)$:

$$\begin{aligned} \gamma &= \arccos(\langle \mathbf{n}_p, \mathbf{n}(x_0, y_0) \rangle) \\ &= \arccos\left(\frac{1}{\sqrt{1 + \langle \nabla \mathbf{g}, \nabla \mathbf{g} \rangle}}\right). \end{aligned} \quad (7)$$

Thus, this γ depends on the length of the gradient only. Hence, we can calculate the normals of the generated carvings with the scalar

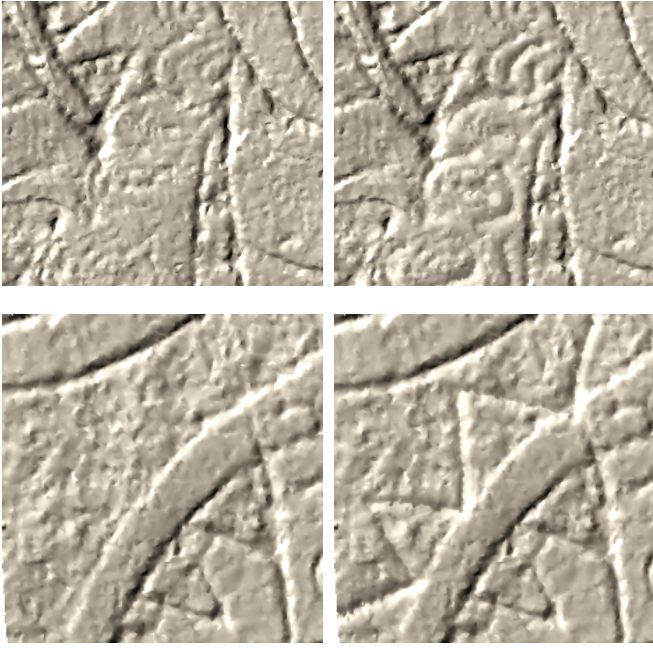


Fig. 7. Two examples where carvings were added (right).

field. After we determined the vertex gradients, we used the smoothed normal to obtain the normals at the carvings. We apply Eq. 7, which yields the angle. This angle is used to determine how much we need to rotate the smoothed normal towards the vertex gradient in order to get the normal \mathbf{n}_c at the carving. Instead of using \mathbf{n}_c , we mix it with the original normals. Otherwise, we would obtain unnatural looking carvings. We use a linear transformation of \mathbf{n}_c and \mathbf{n} to obtain the new normal $\mathbf{n}' = \mathcal{C} \cdot \mathbf{n}_c + (1 - \mathcal{C}) \cdot \mathbf{n}$, where \mathcal{C} denotes the scalar field obtained by the drawing on the surface $f(r_n)$. Thus, we obtain a more natural looking carvings, see Fig. 7.

6 EVALUATION WITH DOMAIN EXPERTS

To assess the utility of the proposed methods, we conducted an informal evaluation with three domain experts. Two of them are monument conservators and one is an expert in the digitization of cultural heritage sites and artifacts. The evaluation was conducted in three steps. First, the participants answered general questions concerning their experience in this field, and provided some personal data. In the second step, we explained to the experts the possibilities that the proposed framework offers. Then, they could use the software for analyzing the data set, selecting and editing carvings, and generating new carvings. During this process, we encouraged them to think out loud and recorded their comments. After testing the tool, the participants filled out a questionnaire concerning their assessment of the usefulness of the tool for their work and their impression of the usability and handling of the software. In the last step, we asked them to mention positive as well as negative aspects about our tool.

General Aspects The general aspects comprise questions about age, gender, and experience. The participants (P1, P2, P3) are 65, 38, and 41 years old, respectively. P1 and P3 are males and P2 is female. (P1,P2) work as monument conservators and have experience for 46 and 20 years, respectively. (P3) is working for an institute that regularly acquires 3D data from cultural heritage objects. He has 20 years of experience. He is also responsible for preparing the data before the monument conservator can use it. Furthermore, we asked about the usefulness of having a tool that identifies the carvings. We also inquired about the participants' background in computer-assisted analysis of carvings and about their experience in analyzing 3D digitizations of carvings. For most of the questions, we prepared 5-point Likert scale answers (+, +, +, -, -). The question about the importance of the

Table 1. Timings of the preprocessing steps.

# Vertices	85k	40k	20k	10k	5k
Time (s)	198	95	48	24	12

analysis of the carvings using digital 3D models was answered with '+ + +' by all participants. The last question in this category was about the importance of the analysis of the carvings with our tool, which was rated with ++ by P2 and P3 and with + by P1.

Framework Aspect After giving the participants the opportunity to use our tool, we asked several questions concerning the usefulness and handling of our tool. In particular, we asked about the possibilities for highlighting carvings and about the editing of the carvings. All participants stated that enhancing the visibility of the carvings is useful. P1 and P2 rated it with ++ and P3 with +. Next, we inquired whether the functionality to delete parts of the detected carvings is helpful. All participants rated this with ++. Another feature is that the illumination of the digital plaster can be modified by the positions and directions of the light sources to improve the perception of the carving structures. This feature was rated as useful and all participants judged it with +. All participants also confirmed the usefulness of the tool for creating new carvings. Here, P1 and P3 rated it with ++ and P2 with +. The question whether the participants find the framework useful for analyzing the carvings was confirmed with ++ by P1 and P2 and with + by P3. Finally, we wanted to know if our method can reliably detect the carvings. P1 and P2 rated it with +, whereas P3 refrained from answering as he does not have enough knowledge about the datasets.

Comments In the last stage of the evaluation, we asked whether the experts would use our framework for analyzing the carvings in a professional setting. All participants replied with 'yes'. The question whether they could use the tool for exploration of digitized plasters was also answered with 'yes' by all three participants. Finally, we asked the experts for additional feedback and for ideas and suggestions for additional features. They emphasized that some *annotation tools* would be helpful for sharing results with other domain experts and for working collectively on a dataset. Moreover, the experts would like to have a *history listing* who made which changes and the possibility to restrict the user's access, such that an inexperienced user could only analyze but not edit the data.

7 RESULTS

The plaster we consider is on a wall with width of 2.22 m and height of 0.98 m. After digitization of the plaster, the data comprises around 10 million vertices and around 21 million triangles. To handle the data set, we split the digitalized plaster in three parts and analyze them separately. For the sake of efficiency, we did not work on the fully resolved data set in most experiments. To create lower resolutions, we simplified the mesh using Edge Contraction with Quadric Error Metrics [11].

Resolution experiments The experiments were performed on a mid-class notebook with an Intel Core i5 with 2.6 GHz, 8 GB RAM, and an nVidia GeForce 870M with 4 GB. In Table 1, we show run times for different resolutions of the dataset. The preprocessing includes the curvature estimation, the computation of the vesselness measure and the determination of the ordered connected components of the carvings. As expected, the run times are linear to the number of vertices. On different datasets with approximately the same number of vertices, we obtained similar run times. Figure 8 shows a comparison of results obtained using different resolutions of the mesh.

Robustness experiments To test the robustness, we applied our method to a parametric surface: $z = 2^{-k} \cos(2 \cdot \sqrt{x^2 + y^2})$, with $x, y \in [-10, 10]$. Depending on k , the height varies ranging from $-1/2^k$ to $1/2^k$. For different k , we tested if our algorithm can detect the carved structures, see Fig. 9 top row. The surface is depicted as well as the detected carvings top left. Thus, for different heights the carvings can be detected well. Afterwards, we added noise in normal direction, i.e.,

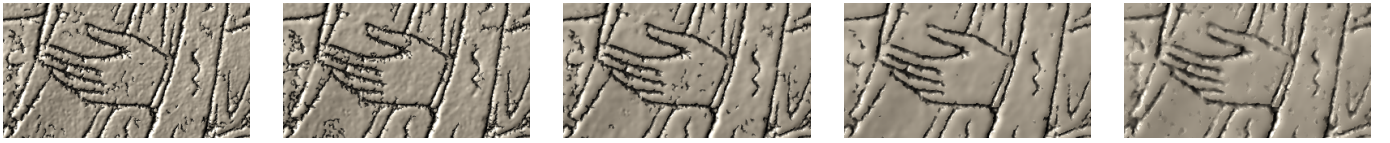


Fig. 8. A cutout of a model with different resolution, which corresponds to the number of vertices mentioned in Table 1.

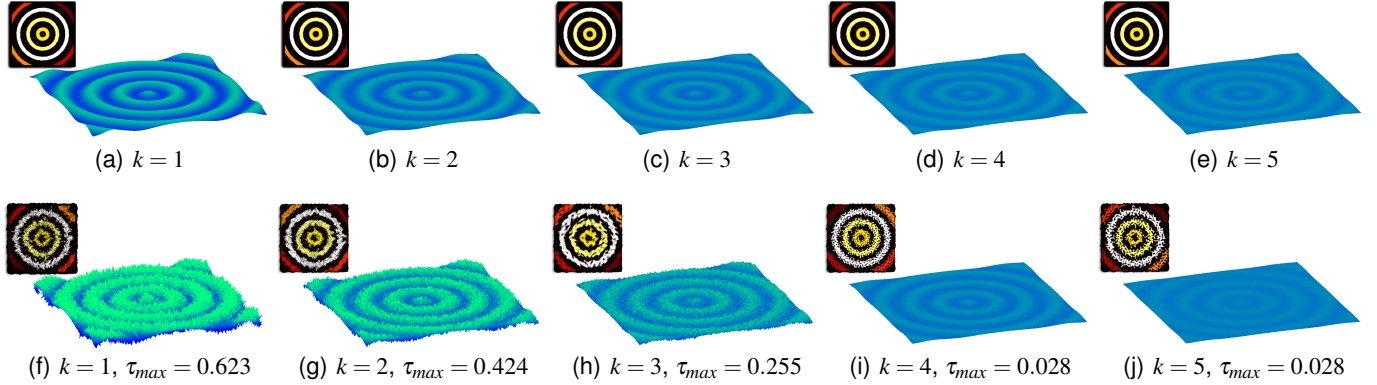


Fig. 9. A parametric surface with different heights without noise (top row) and with noise (bottom row). Our algorithm can detect the carvings well, see the small figures in the upper-left corner of the individual examples. The detected carvings are indicated by colors.

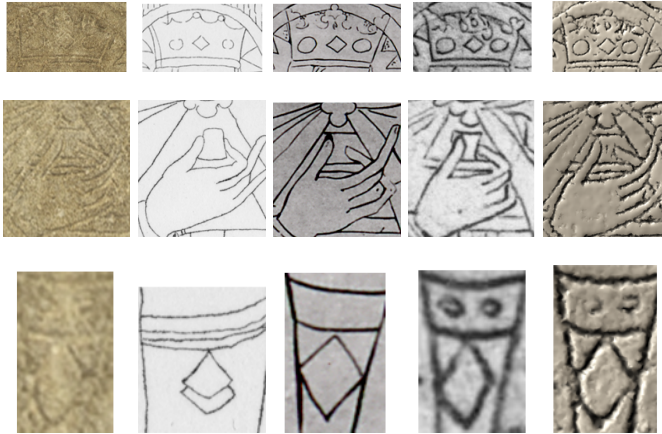


Fig. 10. Different cutouts of the carvings. From left to right: photo, steel engraving (1852), painting (1891), painting (1998), and our approach. First row: the four little circular carvings around the rhomb in the crown can only be perceived with our approach. Second row: based on the domain experts, the fingers of the painting from 1891 and our approach are correct. Third row: our approach and the painting from 1998 are correct.

the points are moved along the normals: $\mathbf{p}_{new} = \mathbf{p}_{new} + \tau \mathbf{n}$, where τ is a random value in $[-\tau_{max}, \tau_{max}]$. Even in the presence of noise, our algorithm can detect the carvings, see Fig. 9 bottom row.

Comparisons with paintings In the last century, artists and photographers tried to depict the carvings in their own work. These images mostly inherit additional knowledge about the carvings. If a carving seems to be discontinuous, the artist can retouch the missing part. Unfortunately, this leads to interpretation and sometimes to wrongly repainted results. In Figure 10, different cutouts are shown: a crown, a hand, and a necklace. As our methods are free from personal interpretations, we can depict the correct illustrations. Note that we did not generate additional carvings in the results. The other paintings may lead to misinterpretations. These paintings were used to show what

the carvings on the wall looked like. Mostly, experts relied on these wrongly painted illustrations. Our domain experts confirmed that these illustrations give rise to false interpretations and that our approach is quite useful to detect the carvings well. They also stated that they want to use our results to publish new insights in the future.

Different datasets For further evaluation of our approach, we applied our method to different datasets. We like to acknowledge the Laboratory of Computer Graphics & Multimedia at the Technion and to the Zinman Institute of Archaeology, University of Haifa for providing us the datasets. In Figure 11, we applied our method to different datasets. Thus, not only carvings can be detected and visualized, but also the letters and details on the ancient stone plates can be enhanced.

8 COMPARISON TO ALTERNATIVE METHODS

We compared our approach with the exaggerated shading technique by Rusinkiewicz et al. [32] and with the D.o.R System [1], which is based on the work of Gilboa et al. [12] and Kolomenkin et al. [19]. The exaggerated shading emphasizes the features in the rendered surfaces well, and thereby highlights the carvings. However, since the technique highlights any surface feature it also emphasizes other parts in the digital plaster like noise or irregularities in the plaster. In contrast, our method separates the noise and natural structures in the plaster from the carvings and emphasizes only the carvings. Figure 12 shows a comparison of the plaster visualized with both approaches. Closest to our work is the method presented by Zatarinni et al. [40]. Their technique works well for coarser carvings, but in case of strong noise and slightly bended surfaces their method cannot fully extract the carvings. For extracting the carvings, they proposed an automatic approach to set the threshold for distinguishing between foreground and background structures. Since this approach may be error-prone with noisy data, we determined the threshold manually to obtain more reliable results. Both our and Zatarinni et al.'s approach are representing the surface containing the carvings using a height field over a base surface. To better explain why our approach gives the expected result, we visualize the height fields and the components, see Figure 13 and 14. Whereas our approach detects the carvings correctly, the approach by Zatarinni et al. results in transition of the height from left to right due to the noisy data and the slightly bended surface. Even if we set different thresholds, not all carvings can be detected.



Fig. 12. Exaggerated shading [32] applied to the dataset (left) and the DoR System [1] (middle). Details can be depicted well, but noise is also enhanced. With our method (right) the carvings (shown in the inset on the figure) are detected and can be used to decently enhance the features.



Fig. 11. Various models with normal shading (left) and our method (right). The last two models show the identified carvings in different colors.

In addition to the comparison to alternative exaggerated shading techniques and the method by Zatarinni et al., we tried feature line techniques to test whether these can be used for extracting and visualizing the carving structures. Our experiments indicate that such methods are too sensitive to noise to compete with the presented method for the extraction of carving structures. Also, enhancing the carvings with feature line methods did not yield results comparable to the proposed scheme. Examples of this kind are shown in Figure 15. The images illustrate that some features can be depicted well, but in all cases noise is also enhanced.

9 DISCUSSION & FUTURE WORK

We present techniques to support monument restorers working with heritage-protected carving structures in plasters. Our methods enable the analysis, processing and visualization of carving structures in digital surfaces. We propose a method for extracting carving structures in surfaces based on the Frangi filter from image processing. The idea is to search in the surface for curvature configurations with a vanishing

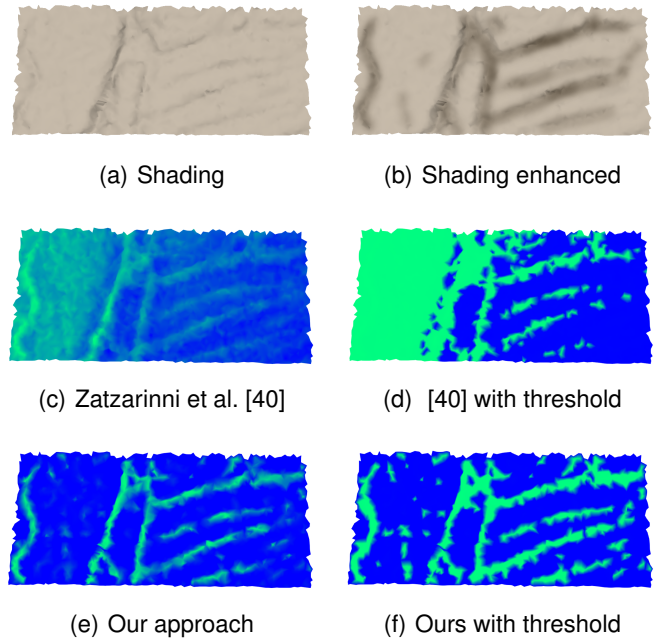


Fig. 13. In (a) a dataset with Phong shading is shown, in (b) the carvings were manually enhanced. In (c) and (d) the technique by Zatarinni et al. is applied. Here, a threshold is set to detect the carvings. In (e) and (f) our technique with correctly detected carvings is shown.

and a positive principal curvature. This concept is implemented by formulating a vesselness measure based on the principal curvatures. Furthermore, we propose techniques that support the user in finding appropriate parameters for the detection process. To robustly extract carvings, a connected component analysis is used in combination with a saliency measure for the components. The component analysis groups regions with the desired curvature pattern in connected carving structures. The saliency measure allows for ranking the groups. Setting a threshold on the saliency allows to robustly extract carvings. The component analysis ensures that coherent structures are obtained and noise is filtered out since it receives a low saliency value.

The carving structures are used as input for our visualization which highlights the carving structure and thereby increases their visibility. The resulting tool allows users to better perceive the carvings, in particular in cases where the structures have flattened due to aging and weathering. Moreover, the user can interact with the carving se-

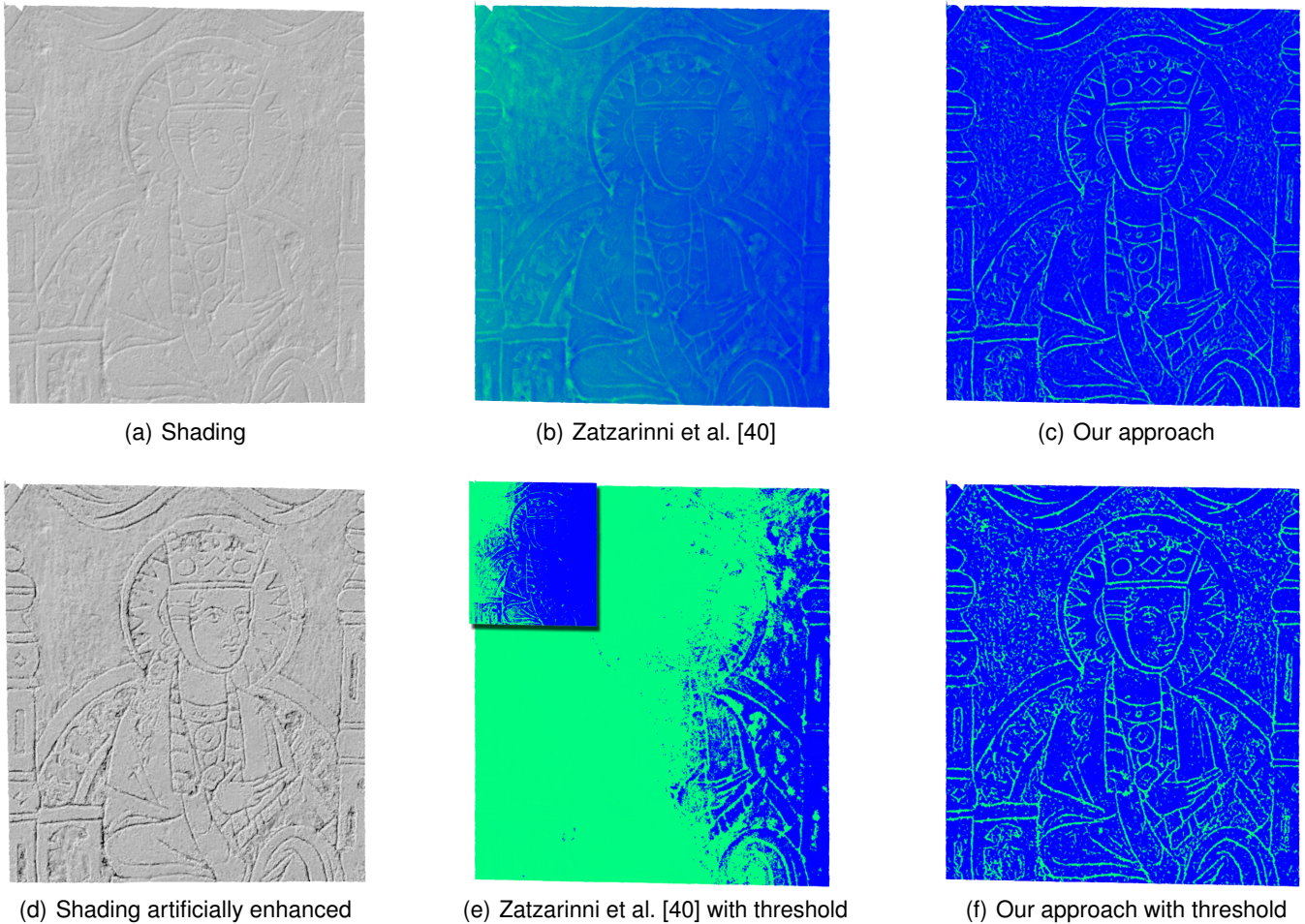


Fig. 14. In (a) a dataset with Phong shading is shown. We manually enhanced the carvings to perceive more easily the features (b). In (b) and (e) the technique by Zatarinni et al. is applied. Setting different thresholds will either detect the carvings left or right. In (c) and (f) our technique with a threshold is shown and it detects the carvings correctly.



Fig. 15. Different feature line methods are applied to the model. From left to right: shading, ridges and valleys, suggestive contours, apparent ridges, and photic extremum lines.

lection process by adjusting a simple and effective set of thresholds. Furthermore, users may wish to add new carvings. We propose an example-based approach for creating carvings in the “style” of the historic original carvings.

We confirmed the robustness of our approach by applying our techniques to artificially generated surfaces with different heights. With our method we were able to show what the carvings look like without risking misinterpretation from artists. As future work, the experts asked for several extensions. In particular, they asked for annotation tools. This would enable collaborative editing and exploration. Moreover, the experts would like to have a log listing who made which changes. The experts also suggested to extend our framework such that datasets with

textures are supported. Then, the analysis could also take the texture into account, in addition to the current geometric analysis.

Altogether, based on the evaluation and our experiments, we consider our work a successful step in the development of visualization techniques and tools for supporting monument restorers.

ACKNOWLEDGMENTS

The authors wish to thank Ernst-Thomas Groll, Claudia Böttcher, Michael Schiller, and the anonymous reviewer for their valuable comments.

REFERENCES

- [1] <http://webee.technion.ac.il/labs/cgm/Computer-Graphics-Multimedia/Software/EdgeDetection/EdgeDetection.html>.
- [2] L. Ammann, P. Barla, X. Granier, G. Guennebaud, and P. Reuter. Surface Relief Analysis for Illustrative Shading. *Computer Graphics Forum*, 31(4):1481–1490, June 2012.
- [3] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.
- [4] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Levy. *Polygon Mesh Processing*. AK Peters, 2010.
- [5] P. Cignoni, R. Scopigno, and M. Tarini. A simple normal enhancement technique for interactive non-photorealistic renderings. *Computer & Graphics*, 29(1):125–133, 2005.
- [6] D. Cohen-Steiner and J.-M. Morvan. Restricted Delaunay triangulations and normal cycles. *ACM Symposium on Computational Geometry*, pages 237–246, 2003.
- [7] D. DeCarlo, A. Finkelstein, S. Rusinkiewicz, and A. Santella. Suggestive contours for conveying shape. *Proc. of ACM SIGGRAPH*, pages 848–855, 2003.
- [8] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical morse complexes for piecewise linear 2-manifolds. In *Proc. of Symposium on Computational Geometry*, pages 70–79. ACM, 2001.
- [9] A. Frangi, W. Niessen, K. Vincken, and M. Viergever. Multiscale vessel enhancement filtering. In *Medical Image Computing and Computer-Assisted Intervention*, volume 1496, pages 130–137. 1998.
- [10] T. Funkhouser, M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin. Modeling by example. *Proc. of ACM SIGGRAPH*, 23(3):652–663, Aug. 2004.
- [11] M. Garland and P. S. Heckbert. Surface simplification using quadric error metrics. In *Proc. of ACM SIGGRAPH*, pages 209–216, 1997.
- [12] A. Gilboa, A. Tal, I. Shimshoni, and M. Kolomenkin. Computer-based, automatic recording and illustration of complex archaeological artifacts. *Journal of Archaeological Science*, 40(2):1329 – 1339, 2013.
- [13] K. Hildebrandt and K. Polthier. Generalized shape operators on polyhedral surfaces. *Computer Aided Geometric Design*, 28(5):321 – 343, 2011.
- [14] K. Hildebrandt, K. Polthier, and M. Wardetzky. Smooth feature lines on surface meshes. In *Proc. of Eurographics Symposium on Geometry Processing*, pages 85–90, 2005.
- [15] V. Interrante, H. Fuchs, and S. M. Pizer. Conveying the 3D Shape of Smoothly Curving Transparent Surfaces via Texture. *IEEE Transactions on Visualization and Computer Graphics*, 3(2):98–117, 1997.
- [16] T. Judd, F. Durand, and E. Adelson. Apparent ridges for line drawing. In *Proc. of ACM SIGGRAPH*, pages 19:1–19:7, 2007.
- [17] J. Kerber, M. Wang, J. Chang, J. J. Zhang, A. G. Belyaev, and H.-P. Seidel. Computer assisted relief generation - a survey. *Computer Graphics Forum*, 31(8):2363–2377, 2012.
- [18] M. Kolomenkin, I. Shimshoni, and A. Tal. Demarcating curves for shape illustration. In *Proc. of ACM SIGGRAPH Asia*, pages 157:1–157:9, 2008.
- [19] M. Kolomenkin, I. Shimshoni, and A. Tal. On edge detection on surfaces. In *Computer Vision and Pattern Recognition*, pages 2767–2774, June 2009.
- [20] M. Kolomenkin, I. Shimshoni, and A. Tal. Prominent field for shape processing and analysis of archaeological artifacts. *International Journal of Computer Vision*, 94(1):89–100, 2011.
- [21] K. Lawonn, T. Mönch, and B. Preim. Streamlines for Illustrative Real-time Rendering. *Computer Graphics Forum*, 32(3):321–330, 2013.
- [22] K. Lawonn and B. Preim. *Feature Lines for Illustrating Medical Surface Models: Mathematical Background and Survey*, chapter Visualization in Medicine in Life Sciences III, pages 93–132. Springer Verlag, 2016.
- [23] C. H. Lee, X. Hao, and A. Varshney. Geometry-dependent lighting. *IEEE Transactions on Visualization and Computer Graphics*, 12(2):197–207, 2006.
- [24] M. Meyer, M. Desbrun, P. Schröder, and A. H. Barr. Discrete differential-geometry operators for triangulated 2-manifolds. In *Proc. VisMath*, pages 35–57, 2002.
- [25] Y.-W. Miao, J.-Q. Feng, J.-R. Wang, and R. Pajarola. A multi-channel salience based detail exaggeration technique for 3D relief surfaces. *Journal of Computer Science and Technology*, 27(6):1100–1109, 2012.
- [26] T. Mönch, K. Lawonn, C. Kubisch, R. Westermann, and B. Preim. Interactive Mesh Smoothing for Medical Applications. *Computer Graphics Forum*, 32(8):110–121, Dezember 2013.
- [27] H. P. Moravec. Toward automatic visual obstacle avoidance. In *Proc. of Conf. on Artificial Intelligence*, pages 584–94, 1997.
- [28] Y. Ohtake, A. G. Belyaev, and H.-P. Seidel. Ridge-valley lines on meshes via implicit surface fitting. *Proc. of ACM SIGGRAPH*, pages 609–612, 2004.
- [29] I. R. Porteous. *Geometric Differentiation for the intelligence of curves and surfaces*. Cambridge University Press, 1994.
- [30] B. Rieck, H. Mara, and H. Leitte. Multivariate data analysis using persistence-based filtering and topological signatures. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2382–2391, 2012.
- [31] S. Rusinkiewicz. Estimating curvatures and their derivatives on triangle meshes. In *Symposium on 3D Data Processing, Visualization, and Transmission*, pages 486–493, 2004.
- [32] S. Rusinkiewicz, M. Burns, and D. DeCarlo. Exaggerated shading for depicting shape and detail. In *Proc. of ACM SIGGRAPH*, pages 1199–1205, 2006.
- [33] A. Sharma and S. Kumar. User-guided modulation of rendering techniques for detail inspection. In *Computer Graphics Theory and Applications*, pages 1–8, 2014.
- [34] A. Tal. 3D shape analysis for archaeology. In M. Ioannides and E. Quak, editors, *3D Research Challenges in Cultural Heritage*, volume 8355 of *Lecture Notes in Computer Science*, pages 50–63. 2014.
- [35] R. Vergne, R. Pacanowski, P. Barla, X. Granier, and C. Schlick. Light warping for enhanced surface depiction. In *Proc. of ACM SIGGRAPH*, pages 25:1–25:8, 2009.
- [36] R. Vergne, R. Pacanowski, P. Barla, X. Granier, and C. Shlick. Improving shape depiction under arbitrary rendering. *IEEE Transactions on Visualization and Computer Graphics*, 17(8):1071–1081, 2011.
- [37] T. Weinkauff and D. Gnther. Separatrix persistence: Extraction of salient edges on surfaces using topological methods. *Computer Graphics Forum*, 28(5):1519–1528, 2009.
- [38] X. Xie, Y. He, F. Tian, H.-S. Seah, X. Gu, and H. Qin. An effective illustrative visualization framework based on photic extremum lines (pels). *IEEE Transactions on Visualization and Computer Graphics*, 13:1328–1335, 2007.
- [39] S. Yoshizawa, A. G. Belyaev, and H.-P. Seidel. Fast and robust detection of crest lines on meshes. *Proc. of ACM Symposium on Solid and Physical Modeling*, pages 227–232, 2005.
- [40] R. Zattarinni, A. Tal, and A. Shamir. Relief analysis and extraction. In *Proc. of ACM SIGGRAPH Asia*, pages 136:1–136:9, 2009.
- [41] L. Zhang, Y. He, and H. S. Seah. Real-time computation of photic extremum lines (PELs). *The Visual Computer*, 26(6-8):399–407, 2010.
- [42] L. Zhang, Y. He, J. Xia, X. Xie, and W. Chen. Real-time shape illustration using laplacian lines. *IEEE Transactions on Visualization and Computer Graphics*, 17:993–1006, 2011.