Interactive Segmentation of Rock-Art in High-Resolution 3D Reconstructions

Matthias Zeppelzauer*, Georg Poier†, Markus Seidl*,
Christian Reinbacher†, Christian Breiteneder‡, Horst Bischof† and Samuel Schulter†

*Media Computing Group, St. Pölten University of Applied Sciences, Austria, Email: m.zeppelzauer@fhstp.ac.at
†Institute for Computer Graphics and Vision, Graz University of Technology, Austria, Email: poier@icg.tugraz.at
‡Interactive Media Systems Group, Vienna University of Technology, Austria, Email: cb@ifs.tuwien.ac.at

***Abstract*—Petroglyphs (rock engravings) are important artifacts for the documentation and analysis of early human life. Recent improvements in 3D scanning and 3D reconstruction enable the accurate 3D reconstruction of petroglyphs from rock surfaces at sub-millimeter resolution. To enable the indexing, matching, and recognition of petroglyphs in petroglyph databases, the shapes must first be segmented from the reconstructed rock surface. The absence of robust 3D segmentation methods for petroglyphs leaves a gap in the digital processing workflow. In this paper, we present a semi-automatic method for petroglyph segmentation for high-resolution 3D surface reconstructions. A comprehensive evaluation shows that our method is able to robustly segment petroglyphs with high accuracy and that the incorporation of 3D information is crucial to solve the segmentation problem. The presented method represents a major step towards the completion of a full 3D digital processing workflow of petroglyphs.**

Fig. 1. The traditional approach to acquire data about the shapes and locations of petroglyphs on a rock panel is time-consuming. The panel is covered with transparent foil on which the petroglyphs are traced with a marker pen. Image copyright by Alberto Marretta.

I. Introduction

Petroglyphs have been scratched, carved and pecked into rock panels all over the world resulting in a vast number of figures on rock surfaces. These figures represent an important artifact for the documentation and study of human history and development. Traditionally, petroglyph shapes were documented (together with their locations on a rock panel) by manually tracing the shapes as illustrated in Figure 1. This is however an extremely time-consuming task considering the sheer number of petroglyphs that have to be documented. In recent years 3D scanning of rock surfaces has seen increasing activity. Although questions regarding long-term storage and handling of the vast amounts of digitized 3D data still have to be answered, we observe that 3D reconstruction gains increasing importance for the documentation and conservation of rock-art. The detailed 3D scanning and reconstruction further facilitates the development of (semi-)automated tools for analysis and indexing of petroglyph shapes to support rock-art researchers in their daily work and to make petroglyphs easier accessible to the public.

Existing work on the automated analysis of rock-art deals mostly with the recognition and classification of *pre-segmented* petroglyph shapes [23], [24], [8], [6], [7], [18], [17]. The automatic *segmentation* of petroglyph shapes is an important pre-processing step in this context which is still unsolved and even considered infeasible [23]. Only a few works on the segmentation of petroglyph shapes exist. For instance, Deufemia et al. use the manual tracings (as shown in Figure 1) as input for segmentation [7]. To automate documentation and to reduce the efforts for manual tracings, Seidl & Breiteneder proposed a method that is able to segment petroglyphs in photographs of a rock surface [16]. To our knowledge, the segmentation of petroglyphs has not been performed on 3D data so far.

In this paper, we present a method for segmentation of rock-art from high-resolution 3D surface reconstructions. The method is able to automatically segment petroglyphs from a given 3D reconstruction and further allows the user to interactively refine the segmentation. To accelerate processing, we map the 3D input data (usually a point cloud) to image space (a depth map) and extract the surface topography by enhancing the geometric micro-structure in the depth map. The resulting enhanced map is input to a classifier which estimates the probability that a given pixel belongs to a pecked region. Next, we apply an energy minimization to reduce noise and improve the smoothness of the contour of the segmented shapes. Finally, we incorporate user-input in terms of scribbles to iteratively refine the segmentation. Figure 2 illustrates the segmentation process from the input point cloud (left), to the fully automatic segmentation (middle), to the interactively refined segmentation (right). Our quantitative and qualitative evaluation shows that the proposed method provides fully automatic segmentations with high-quality and that interactive refinement is useful to improve results. Moreover, an imple-

Fig. 2. The proposed semi-automatic segmentation process: The left image shows the input point cloud viewed from projection direction. The middle image shows the result of the initial fully automated segmentation. The image on the right shows the final result after manual refinement by using scribbles. The green scribbles represent foreground, i.e., pecked areas and the red scribbles background, i.e., natural rock surface.

mentation of the method using modern graphics cards enables immediate user feedback in less than one second for a one mega pixel image. We further show that segmentation on 3D data strongly outperforms pure 2D segmentation. This underlines the necessity of 3D representations for the documentation and representation of rock-art.

The paper is organized as follows. In Section II we review related work on the automatic analysis of rock-art. Section III describes our 3D segmentation approach. We introduce the material used in our evaluation and the experimental setup in Section IV. Section V presents quantitative and qualitative results. Finally, we draw conclusions in Section VI.

II. Related Work

The workflow of automated petroglyph analysis ranges from (i) the acquisition of rock-art, to (ii) the segmentation of individual petroglyphs, to (iii) the classification and recognition of petroglyph shapes. As petroglyphs are three-dimensional artifacts, it is most reasonable to apply this workflow in their natural domain.

Existing 3D reconstruction algorithms based for example on structure-from-motion (SfM) (e.g., [10], [1]) and structured light scanning (SLS) (e.g., [22]) enable the accurate reconstruction of surfaces at sub-millimeter resolution. Thus, 3D reconstruction has become a promising technique for the digitization of rock-art. An early example in the rock-art domain is presented by Landon & Seales who propose a system for 3D acquisition and presentation of Puerto Rican petroglyphs [11]. A more recent approach that considers multiple scales to acquire 3D data of petroglyphs together with their surrounding rock panel has been proposed by Alexander et al. in [2].

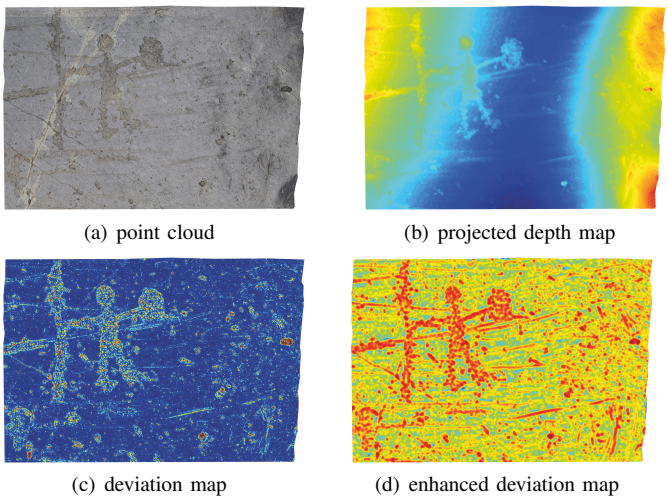
After 3D reconstruction, the detection and segmentation of pecked surface areas is necessary to enable further processing. Most existing methods skip this segmentation step due to its high complexity [23] and apply shape analysis and classification directly on pre-segmented or manually segmented figures. Zhu et al., for example, use a modified Generalized Hough Transform (GHT) for the mining of large petroglyph shape datasets [24]. Deufemia et al. apply the Radon transform as a shape descriptor for unsupervised shape recognition via self-organizing maps (SOM) [8]. In a second step, they use a fuzzy visual language parser to solve ambiguous interpretations by incorporating archaeological knowledge. Additionally they propose a two stage classification of petroglyphs in [6]. The authors employ shape context descriptors to generate an initial clustering with SOMs. In the second step, they use an image deformation model to classify petroglyph shapes. Subsequently, Deufemia et al. use Fourier descriptors to detect and classify petroglyphs from full scenes which stem from manual tracings [7]. Seidl et al. combine skeletal descriptors with shape descriptors for petroglyph classification [18], [17].

The automatic segmentation of rock surfaces containing petroglyphs has rarely been addressed so far. Deufemia and Paolino [7] apply segmentation to digitized manual tracings. This is a rather straight-forward task since manual tracings are binary images that already represent a manual pre-segmentation of the figures. A first attempt towards the segmentation of petroglyphs directly from images of natural rock surfaces has been proposed by Seidl & Breiteneder [16]. The authors employ high-resolution digital photos of rock surfaces as input and extract a set of texture features from the images. Next, they train an ensemble of SVM classifiers on the features and propose a fusion of the classifier outputs that enables the interactive fine-tuning of the segmentation by the user.

In contrast to the approach of Seidl & Breiteneder, we use 3D point clouds of the rock surface instead of images. In images the geometric surface structure is represented in the image texture. Image texture is, however, highly dependent on illumination. Light (almost) parallel to the surface, for example, creates shadows which to some degree reflect the surface geometry, while orthogonal light makes the surface structure practically invisible. Our approach is independent of lighting as the 3D data explicitly represents the geometric structure of the surface. Thus, we expect the 3D data to be a well-suited basis for the detection and segmentation of petroglyphs shapes.

III. 3D Segmentation Approach

Our approach first maps the 3D surface reconstruction to a two-dimensional depth map to obtain a compact representation of the surface geometry. Next, we enhance the depth map to extract and emphasize the fine geometric structures related to the peck-marks that make up petroglyph shapes [21]. A classifier is trained on the enhanced map that generates probabilities for each pixel. Akin to Santner et al. [15], the resulting probability map is input to an energy minimization that generates a binary segmentation of the input with a smooth contour. To leave the final decision of the segmentation to the user, we incorporate a mechanism for iterative refinement of the segmentation result. The additional user input enables our method to learn incrementally and to improve future segmentations.



(a) point cloud (b) projected depth map

(c) deviation map (d) enhanced deviation map

Fig. 3. A 3D reconstruction of a rock surface in which the shape of a human figure has been engraved. (a) The colored point cloud from Figure 4(a) viewed from projection direction; (b) The projected depth map; (c) The deviation map; (d) The deviation map with enhanced topographic structure that emphasizes peck-marks that make up the engraving.

A. Data preprocessing and enhancement

The extraction of surface topography directly from the 3D Point cloud is computationally demanding and impractical for larger point clouds. For a more efficient processing, we generate a 2D heightfield and then extract topographic information from the projected depth map. This mapping can be done without loss of information, as there are usually no self-occlusions in pecked rock surfaces. Thus, the depth map is able to represent the full geometric information of the surface.

To compute the mapping, we estimate a support plane for the input point cloud that minimizes the least squares distances to the point cloud. Next, we estimate the location of each 3D point on the support plane by an orthographic projection with projection direction set to the normal direction of the support plane. We map the signed distances between the 3D points and the support plane to the respective projected location on the support plane. The result is a 2D depth map that represents the distances of each point to the least square surface. Figure 3(b) shows an example depth map obtained for the point cloud in Figure 3(a).

The values in the depth map reflect both the global curvature of the rock surface and the geometric fine structures of the engravings. The global curvature, however, masks out the fine structure due to its much larger variation. To compensate the global curvature we first apply a local mean adjustment to every pixel location of the depth map. The result is a mean adjusted map (the deviation map) which has zero mean locally. The deviation map (Figure 3(c)) better expresses the local geometric fine structure than the depth map (Figure 3(b)). To enhance local structures of peck-marks we locally filter the deviation map by a Gaussian filter, see [21] for a detailed description. Locations where the filter matches well (e.g. small valleys that relate to peck-marks) yield a stronger response than, for example, rather flat areas where the filter yields a weaker concensus. The result is an enhanced deviation map (EDM) which emphasizes structures that resemble the appearance of peck-marks, see Figure 3(d) for an example.

B. Classification

In order to provide a pixel-wise segmentation of the EDM we estimate the probability for each pixel as belonging to either foreground (pecked rock surface) or background (natural rock surface). The final image segmentation algorithm, which will be described in Section III-C, integrates this information to find an optimal segmentation which stays close to the classifier output and minimizes the boundary length of the segmentation, leading to smoother contours. In order to provide a pixel-wise segmentation of the EDM we estimate the probability for each pixel as belonging to either foreground (pecked rock surface) or background (natural rock surface). The final image segmentation algorithm, which will be described in Section III-C, integrates this information to find an optimal segmentation which stays close to the classifier output and minimizes the boundary length of the segmentation, leading to smoother contours. To obtain this estimate we train a classification model that takes a pixel and its spatial neighborhood as input and outputs the desired foreground probability with respect to the learned model.

We rely on Random Forests [3], [4], [5] as a classification model. Random Forests have been employed for various tasks in different computer science communities, in particular in computer vision [19], [9]. Random Forests are fast to train and allow for iterative refinement which makes them well-suited for our task, where we first want to train a classifier offline on existing labeled data and then adapt the pre-trained model in an online fashion by data obtained from user interactions.

Training Random Forests: A Random Forest consists of a number of decision trees. Each decision tree is a hierarchical classifier that separates the input feature space into non-overlapping cells and assigns each cell a probability distribution over the class labels, foreground and background in our case. Given a set of training examples (vectorized image features of pixels with the corresponding class labels), a decision tree is trained by recursively splitting the data into smaller parts such that one class becomes more and more dominant in the distribution. This objective can be formulated as a maximization of the information gain:

$$G(\mathcal{X}, \Theta) = H(\mathcal{X}) - \sum_{c \in \{l,r\}} |\mathcal{X}_c| \, H(\mathcal{X}_c). \qquad (1)$$

This objective expresses the change in information entropy $H(.)$ when splitting the data into two sets, $\mathcal{X}_l$ and $\mathcal{X}_r$, which are assigned to two child nodes in the tree hierarchy. The splitting function Θ, which typically involves a simple feature value comparison, is chosen as to maximize the information gain. This optimization is typically done via a randomized grid search, where F_s splitting tests are randomly sampled and the best is selected based on the objective. Such a procedure has been proven effective to enforce the decorrelation between the individual trees in the forest [4]. The training/splitting procedure continues recursively until some stopping criteria are met. These criteria typically include a maximum tree depth or a minimum number of data samples per node. If one of the criteria is met, a leaf node is created and the class distribution is calculated given the training data falling into this leaf node.

Prediction with Random Forests: Given a set of T fully trained decision trees, inference with Random Forests is highly

efficient. A new image feature is routed from the root node to the corresponding leaf node by evaluating the learned splitting functions Θ along this path, which is done for all trees. Thus, an image feature ends up with T probability distributions from the T trees. These distributions are simply averaged over the differently trained decision trees to obtain the final prediction of the Random Forest for a single pixel in the input image.

C. Optimization of Segmentation

The output of the classifier described above is a pixelwise probability of belonging to either foreground or background. The simplest way to turn that information into a binary segmentation would be to threshold the probability at some user defined threshold. However, this usually yields a deficient result since the decision is made for each pixel individually and, thus, neglects the structure which is generally contained in natural images at this point. To make use of the underlying structures of natural images and to improve the segmentation result we cast the problem of segmenting an image into two disjoint regions as an energy minimization problem as defined in e.g. [15] of the form:

$$\min_{u\in[0,1]} \left\{ \frac{1}{2} \int_\Omega g(x) \left|\nabla u(x)\right| dx + \lambda \int_\Omega u(x) f(x) dx \right\}, \quad (2)$$

where $\Omega \subseteq \mathbb{R}^2$ is the image domain, $g(x)$ is an edge detector function and $f(x)$ represents the output of the classifier described above. The first term is called regularizer (or smoothness term) and ensures that the solution is piecewise constant, which means that small errors caused by the data term are corrected while sharp edges in the segmentation are retained. The weighting term $g(x)$ enforces the segmentation result to be aligned with dominant edges in the input image. The second term is called data term and ensures that the solution stays close to the output of the classifier. Factor λ allows for steering the relative influence of the regularizer.

To minimize the objective we rely on a convex relaxation approach [14]. For efficiency reasons this is implemented on the graphics processing unit (GPU). The result of the minimization is an indicator function u, which is 1 for foreground regions and 0 for background.

D. Iterative Refinement

The manual segmentation of rock-art is often subjective and, thus, may vary between users. Additionally, segmentation errors may remain in the fully automatic segmentation. Thus, a user probably wants to add corrections and modifications to the automatic segmentation procedure. To take this into account our method provides the opportunity to refine the automatically generated results.

We provide two schemes of user interaction. On the one hand the user is able select additional training data for the classifier described in Section III-B. This will affect the results on a more global level, i.e., user interaction in one part of the image will likely affect the results on completely different locations of the image. This type of refinement enables, for example, the adaption of the classifier to a novel pecking style. On the other hand, the user is also able correct the results by making hard decisions. These hard decisions are integrated in the data term and will thus only affect the final optimization step. In both cases the user draws scribbles on the image and assigns a label, i.e., foreground or background to each of them.

In the first case these scribbles are used as new training data in order to adapt the classifier model online. To adapt the classifier we choose a rather straightforward approach. The training samples are traversed down the pre-trained trees in the forest and the class statistics at the leafs are updated, respectively. We opted for this simple procedure because it is extremely fast and offers a lot of flexibility.

In the second case the user's scribbles are integrated in the data term by setting the probability for the respective pixels to zero or infinite and, thus, forcing the optimization to make a specific decisions at these pixels.

IV. Dataset and Experimental Setup

The Camonica valley in Brescia, Italy is a world heritage site since 1979 that provides one of the largest collections of rock-art in the world[1]. According to UNESCO more than 140.000 engravings can be found in the Camonica valley. The rock-art employed in our work stems from different sites in the Camonica valley namely *Seradina*, *Foppe di Nadro* and *Naquane*.

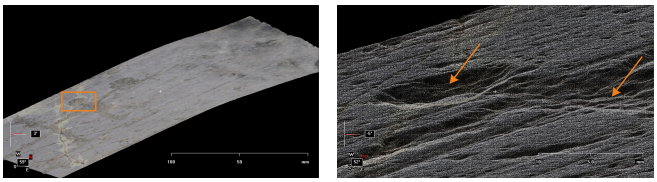
A. Dataset

For the accurate representation of rock-art, high resolution 3D reconstructions are necessary that provide sufficient detail to capture the individual peck-marks that make up a petroglyph shape. We employed structured light scanning (SLS) and structure from motion (SFM) and generated high resolution scans of rock surfaces in the three locations mentioned above. The selected areas show individual motifs (such as antropomorphs, animals, abstract symbols) as well as entire scenes (e.g. hunting scenes). The shapes of the depicted figures are quite complex, consisting of fully pecked areas (inside of body) as well as thin structures (legs and antlers) as well as fine structures that were rather scratched than pecked (called "filiform"). Across different locations, we observe different pecking styles with different roughness and depth. Some areas are partly covered by moss which impedes the 3D reconstruction as well as the subsequent analysis. The figures are partly incomplete, due to damages of the rock surface (e.g. erosion) and due to occlusions by other figures which were pecked on top of others at a later time. These factors impede interpretation and thus also segmentation.

As a result of 3D reconstruction we obtain large unstructured point clouds of several millions of points. An example for a point cloud with 4.4M points together with a close up zoom into the cloud is shown in Figure 4. For each 3D point an RGB color value is available. Point coordinates are converted to metric units (meters). The obtained resolution is in the majority of the reconstructions beyond 0.1mm and thus sufficient to represent individual peck-marks. To account for reconstruction artifacts, we apply smoothing and removal of outlier points prior to automated processing.

A detailed annotation of all reconstructions has been performed by domain experts who labeled all pecked areas

[1] http://whc.unesco.org/en/list/94, last visited March 2015



(a) (b)

Fig. 4. (a) A point cloud obtained by 3D reconstruction; (b) A close up of the head of the antropomorph shown marked by the colored rectangle in (a). The point cloud has 4.4M points. The close up clearly shows the geometric surface structure of the pecked regions (e.g. in the head and torso of the figure, marked with arrows).

(a) (b)

Fig. 5. An orthophoto of a point cloud (a) and its corresponding ground truth labels (b). White areas mark pecked regions and black areas non-pecked regions. In the red area no information is available.

in the reconstructions. Figure 5 shows an example with an annotated hunter and a deer. Both figures are annotated, as well as individual peck marks in the surrounding and the bow represented as filiform. The detailed annotations enable both, the training and the objective quantitative evaluation of our segmentation method. For the annotation orthophotos have been generated from the point clouds by projecting the 3D point colors to a support plane (similarly to the depth map generation in Section III-A).

The selected dataset for our evaluation contains of $N = 26$ 3D scans of rock surfaces from *Seradina*, *Foppe di Nadro* and *Naquane*. Each site represents a different rock from which several detail scans were acquired. All scans together consist of 115.3M 3D points. The number of points per scan ranges from 800K to 10.6M with an average of 4.4M. All surface scans in the dataset were completely labeled as either being an engraved area (class 1) or the natural rock surface (class 2). All individual peck-marks in the scans were labeled, even if they do not spatially coincide with an engraved figure. Class 1 represents 20.9% of the data and is thus underrepresented.

B. Experimental Setup

Our evaluation covers different aspects: (i) we evaluate the performance differences between a full 3D approach (based on depth information) and the traditional 2D approach based on color photos as in [16]; (ii) we investigate the influence of the amount of training data on segmentation performance (generalization ability). For both investigations we use results obtained by the fully automatic segmentation to obtain comparable and objective results. (iii) We simulate interactive refinements and measure the impact on the segmentation performance and (iv) we investigate the benefit of energy optimization on the segmentation output compared to pure pixel-based classification. Ultimately, we present qualitative results to demonstrate the capabilities of our method.

If not mentioned otherwise, we used the following settings throughout all experiments. We employ k-fold cross-validation where k, the number of folds, is 4. This means that the dataset of $N = 26$ scans is divided randomly into k approximately equally sized subsets. Next, we run k evaluations, where each subset is used once as test set while the remaining subsets serve as training set. Finally, the results of all k evaluations are averaged to obtain an estimate of the overall accuracy.

We use a standard patch based approach for learning the Random Forest model [12], [20], where each training sample (pixel) is represented by a square patch (pixel neighborhood). In our case the patches are approximately of size 10×10 mm and sampled from the plain enhanced deviation map, EDM, in the standard configuration of our approach. For evaluation we employ other representations than the EDM (e.g. the depth map or a color image) as input as baselines.

For training, we randomly sampled 4000 patches per class from each scan in the training set. For the entire training set (according to our cross-validation protocol) we obtain approx. 152K training patches to learn our Random Forest model. We train ensembles with 10 trees and stop training if less than 5 samples arrive in a node of a tree or a maximum depth of 18 is reached. The number of sampled tests F_s for optimization of the splitting functions depends on the overall number of possible tests F. We set $F_s = \sqrt{F}$. For the final segmentation we set the parameter λ, which trades off the data term and the smoothness term, to 0.6.

To measure the segmentation quality we employ the *Dice Similarity Coefficient* (DSC) which is a standard measure for the evaluation of segmentation approaches. DSC measures the mutual overlap between the automatic segmentation X and a manual labeling Y:

$$\mathrm{DSC}(X, Y) = \frac{2|X \cap Y|}{|X| + |Y|}. \tag{3}$$

DSC is between 0 and 1 and 1 means a perfect segmentation.

V. Results

To assess the performance of our method we conducted quantitative and qualitative evaluations on the material from Section IV.

A. Quantitative Results

In the sequel we present results of all evaluated aspects described in Section IV-B and point out our contributions.

1) 2D vs. 3D Information: First, we investigate whether the acquired 3D information proves useful for segmentation. We compare the segmentation results when using color information (photographs) with that of a full 3D approach incorporating depth information. To start with, we learned models solely based on either color or depth information. The results are shown in rows 1 and 2 of Table I.

We observe that using color information the results are relatively poor, which is reflected by an average Dice coefficient of about 0.39. However, by employing depth information (from the depth map) a relative improvement of about 50% is achieved, yielding an overall Dice coefficient of 0.59. The average performance is further improved when relying on the enhanced deviation map (EDM) as input representation

Representation	Seradina	Foppe di Nadro	Naquane	Average (over scans)
Color (Plain)	0.394	0.363	0.388	0.389
Depth Map (Plain)	0.581	0.747	0.543	0.590
EDM (Plain)	0.567	0.758	0.619	0.603
Color (LBP)	0.346	0.373	0.372	0.356
EDM (LBP)	0.556	0.753	0.611	0.594

TABLE I. DICE COEFFICIENTS FOR DIFFERENT SETUPS, COMPARING THE CAPABILITIES OF COLOR AND DEPTH INFORMATION. THE RESULTS ARE DEPICTED FOR THE INDIVIDUAL SITES FROM WHICH THE SCANS ORIGINATE AS WELL AS OVER ALL SCANS IN OUR DATASET.

for segmentation as proposed by our approach. These results show (i) that 3D outperforms 2D information and (ii) that the enhancement of the depth map is beneficial for segmentation.

Additionally, to taking color image, depth map, and EDM directly as input, we extract texture features (Local Binary Patterns, LBP [13]) from the representations and use them as input for segmentation instead. Rows 4 and 5 in Table I show the corresponding results. We observe that although the use of texture features improves performance on some of our scans, overall results do not benefit from using LBP features. At first sight this might seem surprising since texture represents an important cue for discriminating areas with different surface geometries. From our experiments, however, we observe that the Random Forest together with the patch-based approach sufficiently captures texture information. This is indeed a well known property of this approach and was also exploited in prominent prior work [20].

Note that, since we were focusing on different representations which mainly affect the classifier, we skipped energy optimization for these experiments. Instead, we directly used the *maximum-a-posteriori* (MAP) estimate of the Random Forest classifier directly to label each pixel as either belonging to a pecked or non-pecked area.

2) Generalization Ability: Next, we investigate the generalization ability of our method. The goal is to gain further insights about the amount of necessary manual labeling effort needed for the classifier to learn a reasonable model of foreground and background. For this purpose we split our dataset into differently sized training and test sets, i.e. we randomly sample a number of m surface scans from our dataset of N scans, train a model on randomly selected samples of the m scans and evaluate it on the remaining $N - m$ scans. We iterate this procedure for different sample sizes m, focusing on small $m \ll N$. Figure 6 shows the results as a function of m, where $m \in \{1, 2, \ldots, \frac{N}{2}\}$. To avoid a bias coming from a particular sample selection, we run these experiments 15 times by sampling different random subsets of the scans for each distinct m. The final results shown in Figure 6 are averaged over all runs.

We observe that the depth map enhancement described in Section III-A strongly improves generalization ability to unseen images when the training set is small. The segmentation performance is almost independent of the size of the training set. Without enhancement the method requires a much higher number of training samples to achieve similar results.

These experiments advance a strong argument in favor of our method. Especially since one of the main rationales behind designing and implementing a system for automatic segmentation of petroglyphs is to reduce tedious manual labeling effort, the ability to generalize from a small amount of training data is crucial.

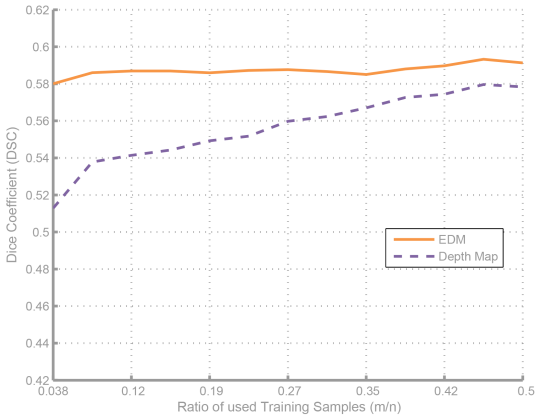


Fig. 6. Segmentation performance as a function of training set size for two input representations: the depth map and the enhanced deviation map. Using a large number of training samples our method performs similar for both input representations. When the number of training samples is small, the enhancement of the depth map strongly improves results.

Training Set: / Test Set:	FdN+N / Seradina	Seradina / FdN	Seradina / Naquane	Average (over all scans)
Depth Map	0.547	0.734	0.535	0.565
EDM	0.558	0.756	0.610	0.595

TABLE II. DICE COEFFICIENTS SHOWING THE RESULTS OBTAINED FOR SCANS FROM *Seradina* WHEN THE CLASSIFIER IS TRAINED ON SCANS OF ONLY *Foppe di Nadro* AND *Naquane* ("FDN+N"), AS WELL AS RESULTS FOR SCANS FROM *Foppe di Nadro* ("FDN") AND *Naquane* WHEN THE CLASSIFIER IS TRAINED ONLY FROM SCANS OF *Seradina*. BEST RESULT IN BOLD FACE.

Another question in this context is whether models learned from scans of one site can generalize to other sites. This is especially interesting since rock-art at different sites may exhibit different peck styles, e.g. because different tools were used for engraving. To answer this question we separated the dataset into two sets, one containing the scans from *Seradina* and the other one the scans from *Foppe di Nadro* and *Naquane* (which are both in neighboring locations and thus similar). Next, we trained our model solely on the scans from one of the two sets and evaluated it on the other, and vice-versa.

Table II shows the results for both partitions of our dataset. We observe that segmentation on the depth map does not generalize well. The average performance drops from 59% (see Table I) when data from all sites is used for training to only 56.5%. The performance difference for the EDM, however, is negligible (59.5% vs. 60.3% when data from all sites are used). These results confirm that the enhancement of the depth map is important for the generalization ability of rock-art segmentation.

3) Interactive Segmentation: Objective evaluations of interactive methods are difficult. The two main issues are that (i) a large user study with a significant number of participants needs to be performed to obtain meaningful results, and (ii) the metrics to measure the outcome of such a study often leaves room for interpretation. For instance, one could examine the number of time or scribbles a user needed to get a "satisfying" result. However, at which level a result is considered satisfying

Setup	Seradina	Foppe di Nadro	Naquane	Average (over scans)
No interaction	0.567	0.758	0.619	0.603
Simulated interaction	0.604	0.810	0.636	0.637

TABLE III. RESULTS FOR SIMULATION OF INTERACTIVE SEGMENTATION. WE PROVIDE THE DICE COEFFICIENTS FOR THE INDIVIDUAL SITES AS WELL AS OVER THE ENTIRE DATASET. THE SIMULATED USER INPUT IMPROVES SEGMENTATION PERFORMANCE.

Methods	Seradina	Foppe di Nadro	Naquane	Average (over scans)
Non-Optimized	0.567	0.758	0.619	0.603
Optimized	0.580	0.781	0.635	0.618

TABLE IV. SEGMENTATION PERFORMANCE (DCE) WITH AND WITHOUT OPTIMIZATION. OPTIMIZATION SLIGHTLY BUT CONSISTENTLY IMPROVES THE SEGMENTATION FOR ALL SITES AND IN AVERAGE.

is certainly subjective.

To circumvent the mentioned problems, but still evaluate the potential of interactive segmentation, we perform a quantitatively measurable simulation of user input. For this purpose, we randomly select a small number of samples from the manual labeling of a given scan and provide it as additional "user input" to our method. Using this information as input we can re-train the classifier, apply it to the same scan and measure the obtained performance difference. For this experiment we sampled 500 pixels for each class per scan, which is a relatively small number considering that the average number of pixels in the projected images is about 29 million. Table III shows that our method is able to improve by adding additional clues for segmentation.

4) Optimization: In a final experiment we evaluate the effect of optimization (Section III-C) on the segmentation performance. As a baseline we evaluate the plain output of the classifier (without interaction), where each pixel is labeled with the class exhibiting the maximum likelihood. Results in Table IV show that the optimization consistently improves the results by a small margin.

The optimization favors larger coherent parts and tends to remove fine grained details. This in average improves the performance in terms of DCE. In practice, however, this may mean additional refinement work necessary by the user because the refinement of the fine details is usually more tedious and time-consuming than adding or removing larger parts. To give the user more control over the segmentation result, we allow the user to steer the optimization by adapting the λ parameter, which indeed trades off data fidelity and cohesiveness. See Figure 7 in Section V-B for visual examples that demonstrate the effect of varying λ.

5) Runtime: Semi-automatic segmentation methods like the one proposed in this work require interaction between user and algorithm. Therefore, the response time to new user input should be minimal. All components of our method are implemented using modern parallel computing architectures provided by graphics cards. The runtime of our method is linear with the number of pixels in the input image. For a 1 MP image, user feedback can be provided within 600 ms, with feature extraction and classification contributing 100 ms and optimization 500 ms. This runtime analysis excludes the projection of the point cloud to a depth map and the creation of the EDM which has to be done only once.[2]

B. Qualitative Results and Iterative Refinement

To complement our evaluation, we present qualitative results obtained with our method. Figure 7 illustrates the segmentation workflow for two different rock surfaces. We provide the input depth map, the enhanced deviation map, the manual labeling, the probabilistic map from the classifier, as well as the automatically generated segmentation and the manually refined segmentation. For both scans different settings for λ are used ($\lambda = 0.6$ and $\lambda = 0.1$ for the first and second surface). A comparison of the final segmentations with manual labeling shows that the proposed method achieves accurate segmentation results.

[2]All timings have been measured on a recent Linux computer using a Nvidia GeForce GTX780 GPU

VI. CONCLUSION

In this paper we have presented a novel method for the segmentation of petroglyph shapes from 3D reconstructions. The approach first enhances the geometric micro-structure of the surfaces and uses the enhanced representation as input to probabilistic classification. Additionally, we apply energy minimization to enhance the contour of the segmented shapes. To enable refinements and the correction of errors, we provide an intuitive method for interactive refinement that in turn lets the method learn and thus improve future segmentations. Our evaluation shows that the proposed method yields accurate segmentations over a large dataset of 3D surface reconstructions. The method clearly outperforms segmentation in 2D (on color images) and yields a high generalization ability even with very limited training data. From our experiments we conclude that the proposed segmentation method has the potential to bridge the gap that exists between 3D reconstruction of rock surfaces and higher-level shape analysis of petroglyphs. Thus, we expect the proposed method to facilitate the documentation of rock-art in future.

ACKNOWLEDGMENTS

The work for this paper has been carried out in the project 3D-Pitoti which is funded from the European Communitys Seventh Framework Programme (FP7/2007-2013) under grant agreement no 600545; 2013-2016. We thank Alberto Marretta for providing image material.

REFERENCES

[1] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. Seitz, and R. Szeliski. Building Rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

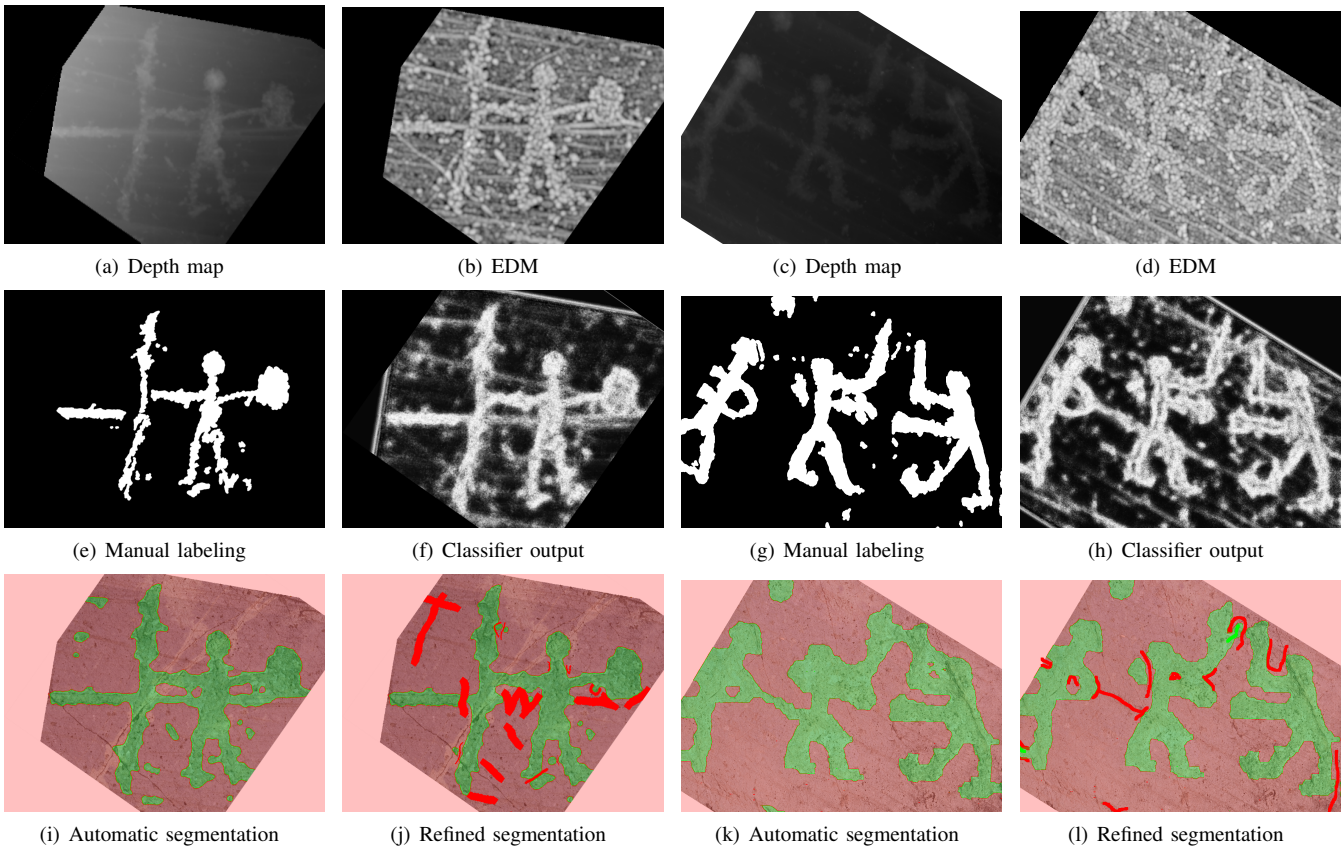
[2] C. Alexander, A. Pinz, and C. Reinbacher. Multi-scale 3d rock-art recording. *Digital Appl. in Archaeology and Cultural Heritage*, 2015.

[3] Y. Amit and D. Geman. Randomized inquiries about shape; an application to handwritten digit recognition. Technical Report 401, Department of Statistics, University of Chicago, IL, 1994.

[4] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[5] A. Criminisi and J. Shotton. *Decision Forests for Computer Vision and Medical Image Analysis*. Springer, 2013.

[6] V. Deufemia and L. Paolino. Combining unsupervised clustering with a non-linear deformation model for efficient petroglyph recognition. In *Advances in Visual Computing*, number 8034 in Lecture Notes in Comp. Science, pages 128–137. Springer Berlin, Jan. 2013.



(a) Depth map (b) EDM (c) Depth map (d) EDM

(e) Manual labeling (f) Classifier output (g) Manual labeling (h) Classifier output

(i) Automatic segmentation (j) Refined segmentation (k) Automatic segmentation (l) Refined segmentation

Fig. 7. Two segmentation examples: left (a,b,e,f,i,j) and right (c,d,g,h,k,l) with intermediate processing steps. In the depth maps brighter pixel values mean higher distance from the camera. The classifier output shows the foreground likelihood, i.e., the data term $f(x)$, where brighter values correspond to higher likelihood. For the refined segmentations, we superimposed the user input (scribbles). For comparison we set $\lambda = 0.6$ for the first example (i) and (j), and $\lambda = 0.1$ for the second example in (k) and (l). Decreasing λ yields smoother segmentations with a higher likelihood of merging larger regions. Images are cropped for better visibility.

[7] V. Deufemia and L. Paolino. Segmentation and Recognition of Petroglyphs Using Generic Fourier Descriptors. In *Image and Signal Processing*, pages 487–494. Springer, 2014. 00000.

[8] V. Deufemia, L. Paolino, and H. de Lumley. Petroglyph recognition using self-organizing maps and fuzzy visual language parsing. In *2012 IEEE 24th International Conference on Tools with Artificial Intelligence (ICTAI)*, volume 1, pages 852–859, 2012.

[9] M. Godec, P. M. Roth, and H. Bischof. Hough-based tracking of non-rigid objects. *Computer Vision and Image Understanding*, 117(10):1245–1256, 2013.

[10] R. Hartley and A. Zisserman. *Multiple View Geometry*. Cambridge University Press, 2003.

[11] G. V. Landon and W. B. Seales. Petroglyph digitization: enabling cultural heritage scholarship. *Machine Vision and Applications*, 17(6):361–371, Oct. 2006.

[12] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2005.

[13] T. Ojala and M. Pietikäinen. Unsupervised texture segmentation using feature distributions. *Pattern Recognition*, 32(3):477–486, 1999.

[14] T. Pock, A. Chambolle, D. Cremers, and H. Bischof. A convex relaxation approach for computing minimal partitions. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.

[15] J. Santner, T. Pock, and H. Bischof. Interactive multi-label segmentation. In *Proc. Asian Conf. on Computer Vision*, 2010.

[16] M. Seidl and C. Breiteneder. Automated petroglyph image segmentation with interactive classifier fusion. *In Proc. of the Indian Conference on Computer Vision, Graphics and Image Processing*, 2012.

[17] M. Seidl, E. Wieser, and C. Alexander. Automated classification of petroglyphs. *Digital Applications in Archaeology and Cultural Heritage*, 2015.

[18] M. Seidl, E. Wieser, M. Zeppelzauer, A. Pinz, and C. Breiteneder. Graph-based similarity of petroglyphs. In *Workshop Proceedings of "Where Computer Vision meets Art" ECCV - European Conference on Computer Vision - 2014 Workshops*, Zürich, CH, 09/2014 2014. Springer.

[19] J. Shotton, R. B. Girshick, A. W. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, and A. Blake. Efficient human pose estimation from single depth images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35(12):2821–2840, 2013.

[20] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2008.

[21] M. Zeppelzauer and M. Seidl. Efficient image-space extraction and representation of 3d surface topography. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE, 2015.

[22] S. Zhang and P. S. Huang. High-resolution, real-time three-dimensional shape measurement. *Optical Engineering*, 45(12), 2006.

[23] Q. Zhu, X. Wang, E. Keogh, and S.-H. Lee. Augmenting the generalized Hough transform to enable the mining of petroglyphs. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, page 1057–1066, New York, NY, USA, 2009. ACM.

[24] Q. Zhu, X. Wang, E. Keogh, and S.-H. Lee. An efficient and effective similarity measure to enable data mining of petroglyphs. *Data Mining and Knowledge Discovery*, 23(1):91–127, July 2011.