

Extracting Scar and Ridge Features from 3D-scanned Lithic Artifacts

Eitan Richardson¹⁾, Leore Grosman²⁾, Uzy Smilansky²⁾ and Michael Werman¹⁾

The Hebrew University, Jerusalem, Israel

¹⁾ Department of Computer Science.

²⁾ Computerized Archaeology Laboratory, The Institute of Archaeology.

Abstract

We propose a method for computer analysis of 3D-scanned lithic artifacts. It provides a detailed description of the scars and ridges that were left on the surface of the stone tool while it was produced by knapping. Traditional documentation and research methods are based on hand drawn images of stone tools and manual measurements. Our algorithm enables automatic, objective and precise documentation of the tool surface, together with a quantitative analysis of the scar and the ridge networks. The proposed algorithm detects ridges on the surface of the scanned object, segments the surface to scars and calculates features that can be used for clustering and classifying lithic artifacts. We demonstrate our method on various lithic artifacts, present the extracted features and compare the results to traditional hand drawings.

1. Introduction

Much of what is known today about prehistoric periods derives from the study of stone artifacts manufactured during the Paleolithic period that spans over 99.5% of human history. The gradual evolution of stone technology from rough and simple forms to highly sophisticated and refined objects marks the cognitive evolution of the human brain, as well as the development of manual, technical and social skills. Prehistoric stone tools were manufactured by knapping flakes from a stone core and creating a scarred surface. The surface that is created after removing a flake is called a *scar* and the boundary of the scar a *ridge*. As more flakes are removed, ridges become borders between adjacent scars on the lithic artifact (Figure 1).

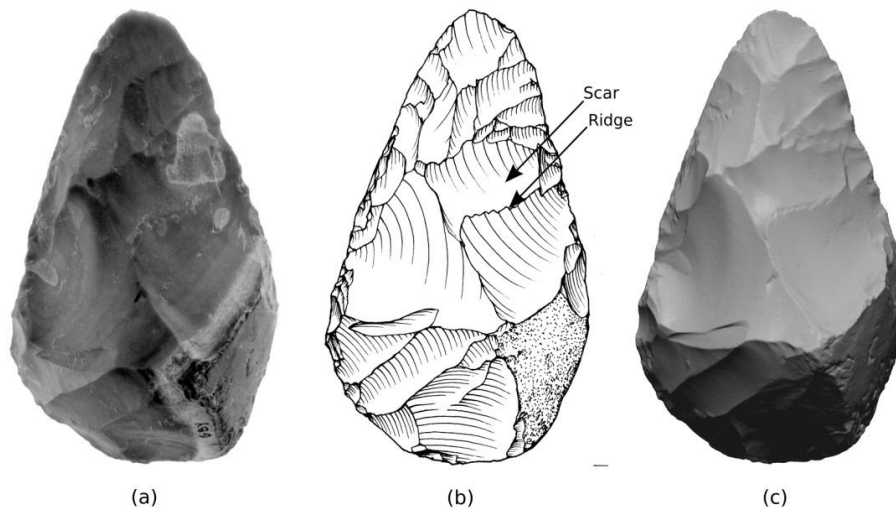


Figure 1: Lower Paleolithic Handaxe from the site NBA (after Grosman et al 2011). (a) A photograph of the stone tool (b) A traditional hand drawing with indication of a scar and a ridge (c) Rendering of the 3D-scanned tool

The most important and difficult task of the archaeologist is to decipher the mute messages embedded in the shapes and forms of the lithic artifacts and thus learn about different aspects of the lithic industry. Of prime importance are questions related to the production technology of the artifacts. The very nature of the irreversible process of knapping implies that the producer of a stone artifact had a clear idea not only of the final product he/she had aimed at, but also of the chain of operations which should be followed to achieve the desired goal. Hence, the mental process is engraved in the morphology of artifacts retrieved from prehistoric sites, which include the waste and the final products of knapping - the tools. This is the reason why a major effort in prehistoric research is devoted to the study of knapped stone morphology, aiming to deduce from the observed features the chain of actions and considerations involved in the production (Bar-Yosef and Van Peer 2009; Boeda et al 1990; Sellet 1993). The most prominent morphological feature of the stone tool surface is the distribution and hierarchy of scars and ridges which decorate it. They provide the only direct evidence for the succession of flake removals. This is why the study of the apparent scars and the ridges that form their boundaries was always considered of supreme importance in prehistory. Scars and ridges provide direct clues for the various phases in handling the raw material of the knapper and also hint to his/her technological skill.

In the present work we utilize a high precision optical 3D scanner that bridges between archaeology and computer graphics/vision methods enabling computer-based analysis of the morphology of prehistoric stone tools. This work relies on previous work that was aimed at positioning and extracting measurements of lithic artifacts using 3D-scanning models (Grosman et al 2008, Grosman et al 2011). We present an algorithm for identifying the ridge and scar pattern on the surface of the object. It enables the introduction of quantitative and intrinsic measures such as the number of scars, their surface area, depth and mean curvature.

The presentation is arranged along the following lines: Section 2 of the paper discusses prior work, both in the field of prehistoric archaeology and in computer vision. In section 3, we provide a detailed description of the scar segmentation algorithm. The extracted features are described in section 4 and some results are presented in section 5, including comparison to hand drawings and to other mesh segmentation methods. In the last section, we discuss the results and possible future work.

2. Related Prior Work

Most of the archaeological research which focused on scars and ridge patterns has been rather intuitive and descriptive. Several attempts were made to explain prehistoric assemblage variability by trying to quantify the extent, nature and variability of the reduction chain (e.g. Davis and Shea 1998; Eren et al 2005, Eren and Sampson 2009; Kuhn 1990).

Attempts to quantify flake scar patterning were based primarily on two dimensional projections (e.g., Conrad et al 2004), drawing the scars manually and subjectively, thus failing to capture the complexity of the 3D nature of stone knapping. Recently Clarkson et al (2006) suggested a 3D analysis of flake scars using a 3D measurement tool. The 3D data was recorded manually and the scars were described in terms of directed lines (vectors) which connect the extreme points of the scar – a rather incomplete description which fails to convey the complexity and detail of the network of scars and their characteristics.

Scar detection, from the computer graphics/vision point of view, belongs to the intensively studied field of image segmentation. In recent years, with the increased availability of three-dimensional data, there has been much research in the area of 3D surface segmentation. One of the challenges in surface (or “mesh”) segmentation is to define the criteria for evaluating the quality of the resulting segmentation. In most cases there is no objective principle which establishes a correct or preferred segmentation. In (Shamir 2008), the author defines two types of mesh segmentation methods, which aim to achieve two different goals. *Part-type* segmentation methods aim to segment the 3D object into meaningful parts, for example, segmenting a 3D human figure into head, arms, legs etc. The goal in *surface-type* segmentation is to segment the surface of the object into patches according to some

criteria. The problem of segmenting the scanned lithic artifact belongs clearly to the latter type. In addition, the underlying knapping process, which ideally creates well-defined surface scars, should provide proper criteria for the success of a given segmentation: it should be consistent with the expert archaeological evaluation.

Surface-type mesh segmentation algorithms are based on various surface attributes. Some examples are surface curvature and geodesic distances, which we discuss in more detail in section 3. Other possible attributes are the normal directions of the triangular faces, the dihedral angle between adjacent faces and the planarity and convexity of the segments. A popular family of segmentation methods is Watershed (Mangan and Whitaker 1998), a region-growing method that starts with local-minima points of some attribute function of the mesh and grows regions until local-maxima points are reached. In section 5, we compare the results of our method to those of Watershed.

Methods for detecting ridges and valleys on a surface embedded in 3D draw on techniques developed for the related problem of edge detection in images. In our work, we use an existing ridge-detection method (Yoshizawa et al. 2005). Applying a ridge detection method to the surface of a scanned stone tool does not solve the scar segmentation problem by itself, the method does not detect all the scar-ridges, and on the other hand, it detects many false ridges. The reason for this failure is that ridges on stone tools have a specific archaeological definition – they are the intersection of two scars. In our method, we utilize ridge detection as an intermediate step towards the scar segmentation as described in section 3.5.

3. Detecting Scars and Ridges

3.1 Formal Definition of the Segmentation Problem

The main goal of our algorithm is segmenting the surface of the scanned tool into scars. The stone tool is scanned using a structured-light 3D scanner, generating a triangular mesh (Figure 2), which is a surface in three-dimensions defined by small triangles – a set V of n vertices $v_1...v_n$ in R^3 and a set F of m triangular faces $f_1...f_m$, each face defined by an ordered set of three vertices. The goal of the segmentation task is to divide the set F of m faces into k disjoint sets $F_1...F_k$, such that each F_i corresponds to a single scar on the surface of the object. It should be noted that the number of segments k is not an input to our algorithm – it is detected automatically. Once the mesh has been segmented into scars, the ridges can be defined as the borders between scars (the polygons connecting the ordered set of vertices that form the border of each segment). For rendering purposes, these polygons can be smoothed to form a smooth ridge curves.

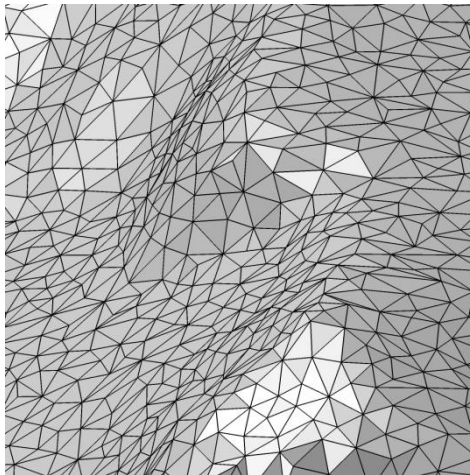


Figure 2: A close-up look on a scanned object, represented as a three-dimensional triangular mesh.

3.2 Outline of the Algorithm

Our method is based on the importance of high-curvature ridges for identifying borders between scars and on the other hand by the need for a robust method that will ensure the spatial consistency of the scars.

The surface curvature value at each point is first calculated. This value measures how strongly the surface bends at each point. Curvature ridges are then detected. These are continuous lines on the surface which follow local maximum points of the curvature value. The curvature-ridges mostly coincide with the actual scar-ridges. The curvature values and ridges are then utilized in a clustering process which breaks the surface of the object into small sub-scars. The clustering process is done in a way that ensures that a single cluster will not cross a high-curvature area or a ridge. In the next step, the border between each adjacent pair of clusters is analyzed, clusters with a weak border between them are merged together. At the end of this process, the small sub-scars have been merged to become complete scars. A final optimization step is then performed to fine-tune the segmentation. At the end of the optimization step, the final scars are defined and the border between each adjacent pair of scars is defined as a ridge.

The list below outlines the main steps in our algorithm. Each step is further detailed in the sections below. Figure 3 shows the intermediate results.

1. Preprocess the mesh (3.3)
2. Estimate the maximum principal curvature at all vertices (3.4)
3. Detect curvature-ridges (local-maxima lines of the principal curvature) (3.5)
4. Perform initial geodesic clustering into sub-scars, using random seeds (3.6)
5. Merge adjacent clusters that have a weak border (3.7)
6. Fine tune the segments using graph-cut optimization (3.8)
7. Define the ridges as the borders of the segments (3.9)

List 1: Outline of the algorithm

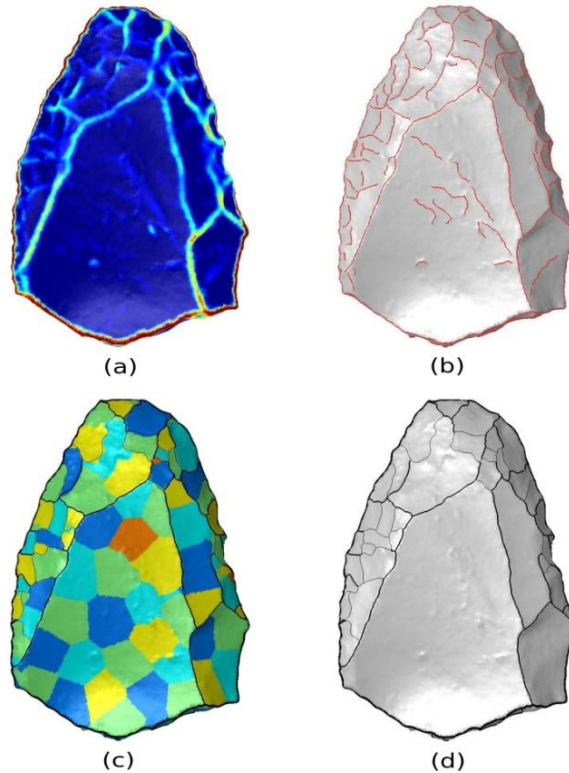


Figure 3: Intermediate steps results. (a) Estimated maximum principal curvature (b) Detected ridge lines (c) Geodesic clustering (segments indicated by colors) and post-merging (indicated by black borders). (d) The final scar-ridges after graph-cut optimization

3.3 Mesh Preprocessing

Artifacts can be scanned at different levels of detail, or resolution. As a pre-processing step, we apply a mesh simplification technique (Siek et al. 2002) to bring the resolution of all processed tools to a common value. In addition, we use Principle Component Analysis (Jackson 1991, Jolliffe 2002) to position the object in 3D so that the main axis of the tools is parallel to the x axis *etc.* This step is required for the proper calculation of the object’s dimensions.

3.4 Principal Curvature Estimation

The principal curvature values and directions at each point on a smooth surface embedded in a three-dimensional space are mathematically defined as the two eigenvalues (k_1 , k_2) and corresponding eigenvectors of the shape tensor at that point. The principal curvature on a triangular mesh is estimated using (Cohen Steiner and Morvan 2003). As can be seen in Figure 4(c), the maximum curvature value (k_1) is high along the ridges.

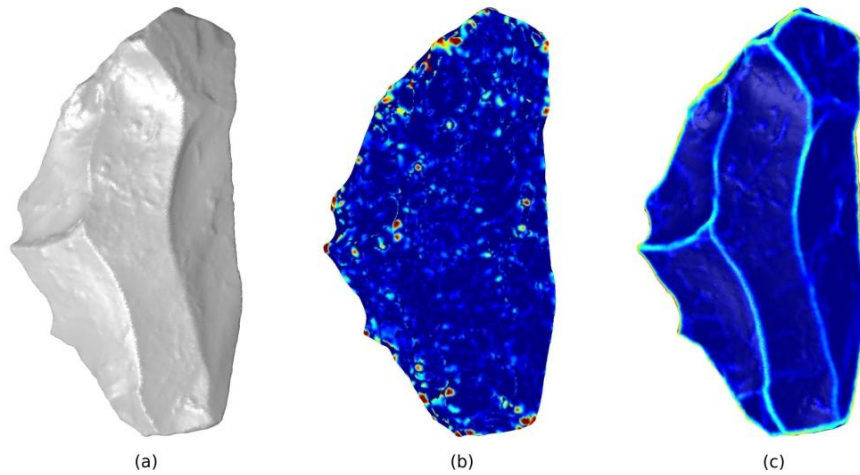


Figure 4: (a) The original scanned object. (b) The minimum principal curvature value - k_2 (not informative for detecting the ridges.) (c) The maximum principal curvature value - k_1 .

3.5 Curvature-ridges Detection

We apply a ridge-detection method (Yoshizawa et al. 2005) to detect curvature-ridges on the surface of the object. Curvature-ridges are defined as local-maximum points of the maximum principal curvature value (k_1) in the principal curvature direction (Koenderink 1990). We use the term “curvature ridges” to differentiate these ridges from the stone artifact ridges or “scar ridges”, which are the goal of our method. The differences are:

- Curvature ridge detection identifies many minor ridges, which are caused by imperfections in the shape of the object. Most of these ridges are irrelevant to our work and are considered false detection.
- In some cases, parts of the boundary of a scar will not be detected as a curvature-ridge. Our goal is to detect the entire boundary of each scar.

Note that (Yoshizawa et al. 2005) also detects valley lines on the mesh, which are points of negative curvature. Our method does not use the valleys. Figure 5 shows the detected ridge lines.

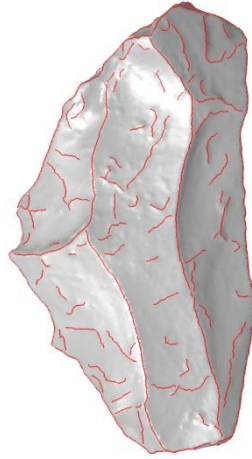


Figure 5: Detected curvature ridges.

3.6 Geodesic Clustering

As a first step towards segmenting the scars, we perform initial clustering of the mesh faces. If we consider the original mesh as a clustering in which each face forms a cluster, our goal is to reduce the number of clusters from an order of 100K to several hundred. In addition, we require clusters to be connected component and we want the borders between adjacent clusters to be consistent with the curvature-ridges.

A clustering process assigns to one cluster elements that are close to each other according to some distance function. Figure 6 demonstrates three vertex distance functions. The Euclidean distance causes points on opposite sides of the mesh to be close and therefore is not suitable for mesh clustering. The geodesic distance measures distances on the surface of the mesh. We use the Fast Marching method (Sethian 1999) to estimate the geodesic distance between a starting point to all other points on the mesh. We use non-uniform edge weights, based on the maximum curvature value at each vertex, thus increasing the geodesic distance between a pair of points on the two sides of a high-curvature ridge, as can be seen on the right column of Figure 6. This encourages low-curvature areas to be clustered together and the border between clusters to coincide with the curvature-ridges.

Our clustering method proceeds along the following line: For each vertex we maintain three properties - a boolean flag indicating whether it is a part of a cluster, the assigned cluster (label) and the geodesic distance value to the segment center (or seed) point. We start with all vertices being unoccupied and having an “infinity” distance from a seed. We choose a random vertex (random seed) on the mesh and calculate its weighted geodesic distance to all other vertices. We then mark the geodesic neighborhood of the seed (all vertices whose geodesic distance from the seed is smaller than a predefined threshold) as occupied. We assign to the new seed all vertices that are geodesically closer to it than to their current seed. We then choose the next random seed that is not occupied and does not lie on a high-curvature ridge. The process continues until there are no available vertices.

Note that although the above method clusters the set of vertices and not the faces, it is easy to define a clustering of the faces based on the vertices clustering. We use majority voting of the clusters of the three vertices. In rare cases, the three vertices of a triangular face will belong to three different clusters. In this case, we choose the cluster of the first vertex.

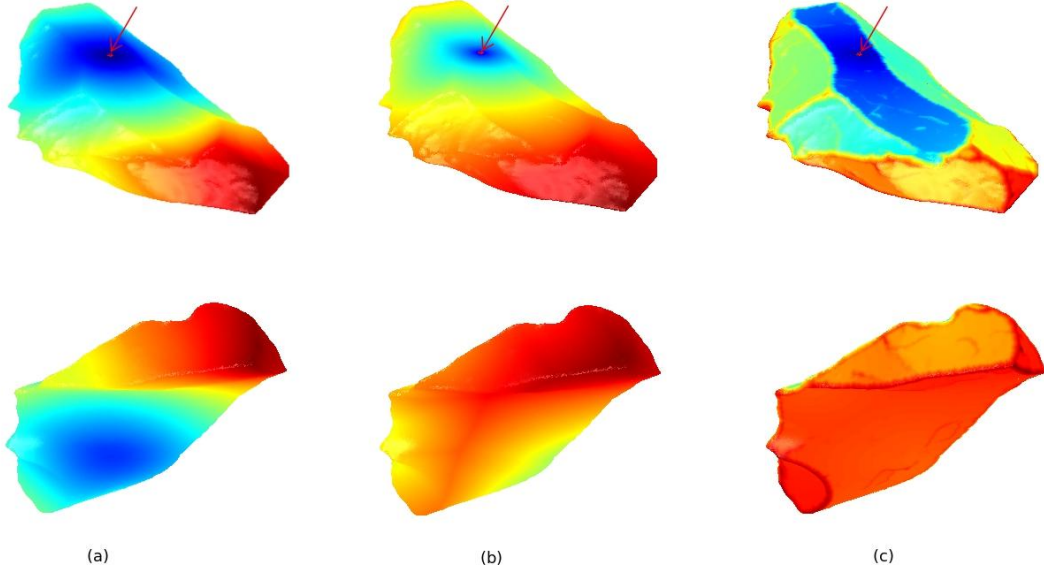


Figure 6: Distance functions. Each column shows the distance between the point marked by a red arrow in the top image to all other points on the object, according to a distance function. The distances are indicated using blue-to-red color coding. The objects on the bottom row are rotated to show the other side. (a) Euclidean distance. Note that points on the other side of the object are close to the marked point. (b) Geodesic distance. (c) Geodesic distance with a weight function based on the maximum curvature.

3.7 Post Merging

The geodesic clustering phase generates sub-scar clusters. These clusters (or segments) represent over-segmentation of the surface of the mesh, however, the segments do not cross the scar-ridges. The border between a pair of adjacent segments contains points that have equal geodesic distances to the two seeds. These borders fall into two categories - borders that strongly depend on the randomly chosen seeds of the clusters, and stable borders that do not depend on the exact location of the seeds. On flat areas (or areas with negative curvature), the points in which the geodesic distances from the two seeds are equal strongly depend on the selected random seed. However, if the two seeds fall on the two sides of a high-curvature ridge, the dependency will be much lower. Randomized Cuts (Golovinskiy and Funkhouser 2008) is a method that utilizes this feature to identify meaningful borders, by running the clustering algorithm multiple times and keeping only the repeating (or stable) borders. We found that in our problem, it is sufficient to identify the meaningful borders by calculating the median maximum-curvature value along the border and merging adjacent segments with a low-curvature border. Figure 7 shows the results of the geodesic clustering and post-merging.

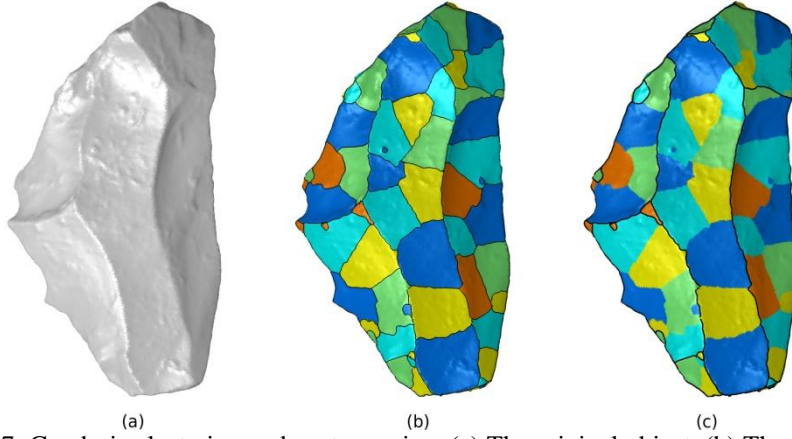


Figure 7: Geodesic clustering and post-merging. (a) The original object. (b) The geodesic clustering result. (c) Post-merging: The colors indicate the original geodesic clustering and the black borders indicate the result of the post-merging (random borders removed).

3.8 Graph-Cut Optimization

Graph-cut is a familiar method for image and for mesh segmentation (Boykov and Jolly, 2001). We use graph cut as a final post-processing segmentation phase. Because we want to improve the quality of the border between the segments in addition to improving the segmentation, we use graph-cut at the individual triangular faces level. The graph-cut algorithm solves the min-cut optimization problem of partitioning a graph into two disjoint groups of vertices while cutting edges with a minimal total weight. In our case, we use a variant of graph-cut (alpha-expansion), which allows partitioning the graph into a larger number of segments. The input to graph-cut is the weighted face-adjacency graph. Each face forms a node in the graph and an edge connects each adjacent pair of faces (creating a 3-regular graph). We now add a node for each segment label and an edge between each face-node to its initially-assigned segment. The weights of the edges connecting face-nodes are the *smoothness terms*. They define the cost of cutting the mesh into two segments at each edge. The weights between the face-nodes to the segment-nodes are the *data term* and represent the cost of changing the original assignment of a face to a segment. In our method, we base the smoothness term on the maximum-curvature and on the detected ridges. An edge that passes through a detected ridge line will have zero smoothness term (thus encouraging graph-cut to segment the mesh at these edges). Other edges will have a weight that is proportional to $1/\text{max-curvature}$, so that the cost of partitioning at a flat area is high.

3.9 Defining the Ridges

The final scar ridges are the borders between scars. We define the border between each adjacent pair of scars as a ridge line. We assign a weight to each ridge line, based on the median of the maximum-curvature along the ridge line. This weight can be used to visually emphasize major scars by drawing them with a thicker line. In addition, we apply a smoothing function to the ridge lines. Figure 8 shows the final ridges, which are the result of the graph-cut optimization and the ridge definition phases.

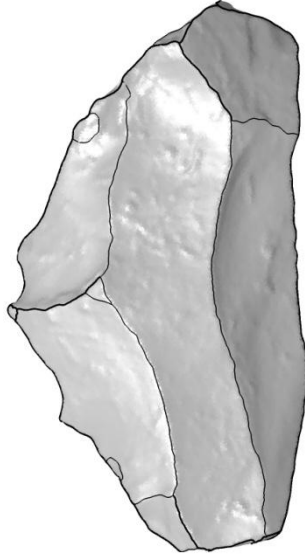


Figure 8: The final scars and ridges.

4. Feature Extraction

As presented in section 1, one of the goals of our work is to calculate features that can be used for automatic classification and clustering of scanned artifacts. Clustering and classification algorithms require features that are descriptive and discriminative (can differentiate between classes of objects). (Tangelder and Veltkamp 2007) define several categories of 3D shape matching methods, for example, global descriptors and local features. Both these methods “take into account only the pure geometry of the shape”. In contrast, graph based methods “attempt to extract a geometric meaning from a 3D shape using a graph showing how shape components are linked together”.

In the case of lithic artifacts, proper segmentation of the surface into meaningful scars allows generating a graph-based descriptor, which is rich and descriptive on one hand and with a reasonable dimensionality (k) on the other hand. Figure 9 shows the scar adjacency graph diagram super-imposed on the drawing of the object. Note that each scar node can contain several attributes of each scar, such as:

- Area of the scar
- Parameters of best-fitting plane, planarity value
- Mean normal direction
- Histogram of curvature values (1D or 2D)
- Shape of scar (use some shape feature) / of ridge

In addition, edges can hold information about the borders between scars, for example, the mean curvature value along the border. Usage of graph-based descriptors require a definition of a metric and some graph matching algorithm such as (Melnik et al. 2002) in the classification or clustering phase.

In addition to the scars graph-based descriptor, global features such as the ones described in (Tangelder and Veltkamp 2007) can be calculated for the entire object:

- The 3D shape histogram, as defined by (Otagiri et al. 2002)
- Histogram of the principal curvature (1D or 2D)
- Histograms of normal distances from centroids
- The volumetric moments of inertia (Zhang and Chen 2001)

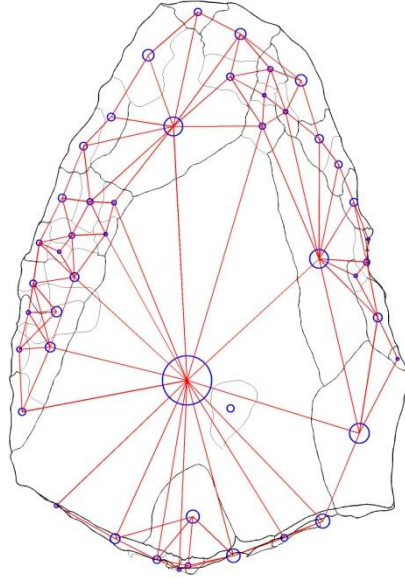


Figure 9: Lithic artifact represented as an adjacency graph of the scars. Size of the scar nodes in the drawing (blue circles) is proportional to the diameter of the scar. Edges (red) connect adjacent scars. The edges connecting to main scar on the back of the object are not shown.

5. Results

Figure 10 shows the output of our method on 13 different stone tools. The figure shows the scar ridges (the borders between the detected scars).

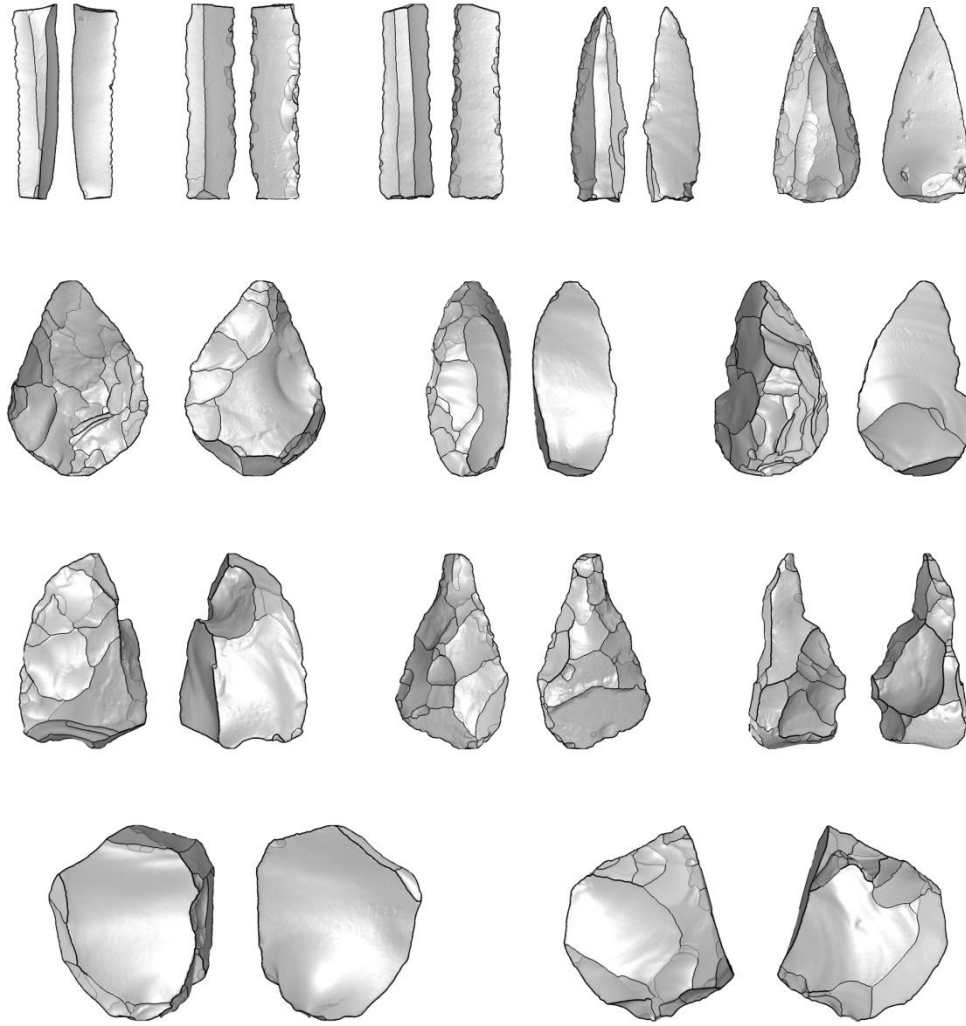


Figure 10: Scar segmentation results.

Next, we compare our results to hand-drawings (Figure 11). As can be seen, our method created a segmentation which is quite similar to the segmentation that is expressed in the hand-drawing, although in some cases, our method broke a single scar to two or more parts. This is further discussed in section 6.

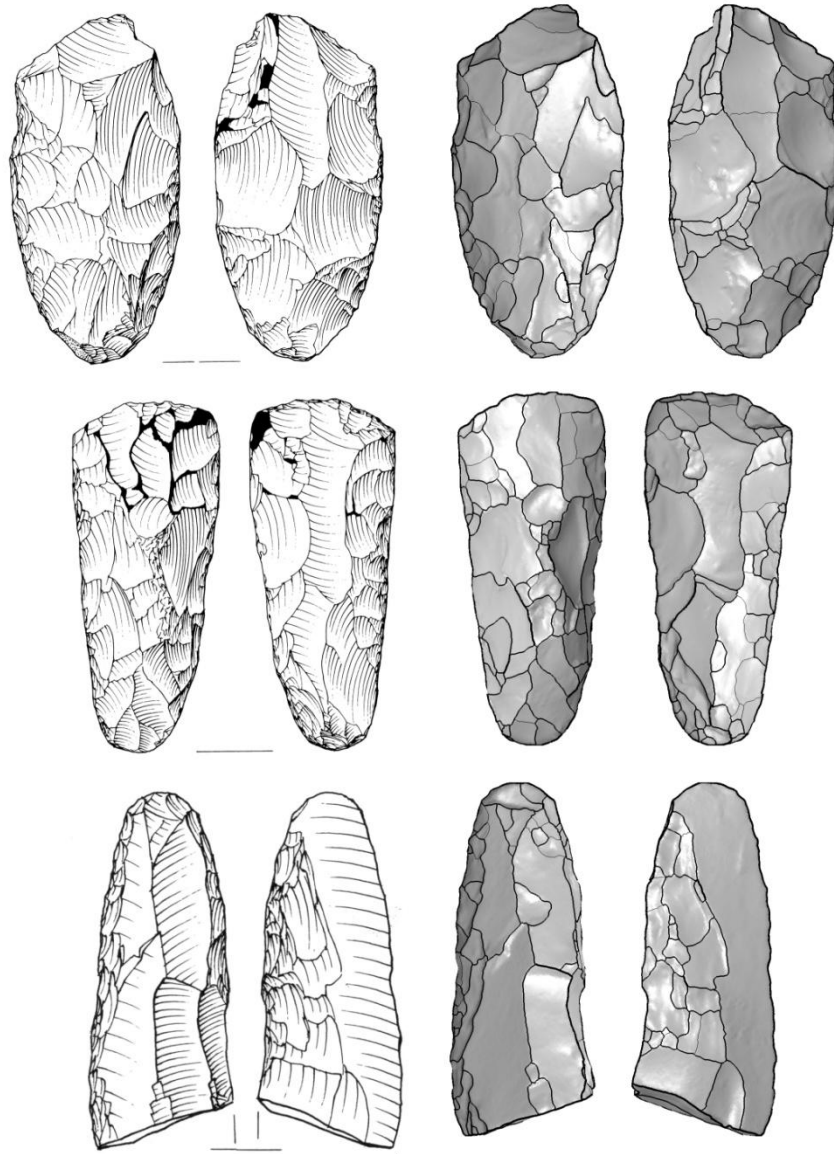


Figure 11: Segmentation results compared to traditional hand drawings (our results shown on the right.) Tools are from the Neolithic site of Abu-Gosh, excavated by Ianir Milevski, Drawn by Leonid Zeiger, Israel Antiquity Authority.

We compared the performance of our method to Watershed (Mangan and Whitaker 1998), which is considered to be one of the more robust mesh segmentation methods. We ran an implementation of Watershed that uses the principal curvature value as a height function. As can be seen in figure 12, the out-of-the-box performance of Watershed is quite good, but compared to our method, it produces less accurate ridges and fails to segment some of the scars.

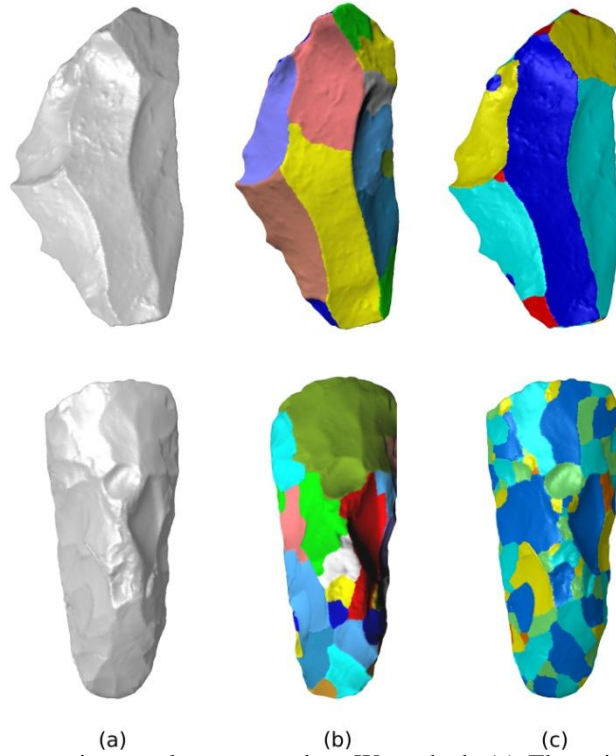


Figure 12: Segmentation results compared to Watershed. (a) The original objects. (b) Watershed results. The result is less consistent with the actual scars (e.g. on the right side of the object on the top row, at the top of the object in the second row). (c) Our method

6. Discussion and Future Work

The surface scars and ridges of stone tools are highly valuable for morphological study and for documentation. We presented a method for automatic detection of the scars and ridges on 3D-scanned lithic artifacts and for extraction of quantitative global and scar-related features, which can be used for clustering and classification of prehistoric artifacts. As presented in section 5, the scar segmentation result is consistent with prior methods, however, it is not always identical to the manual segmentation, which is expressed by the hand drawing. For example, our method might segment a scar into two parts because of some irregularity of the scar surface, while in the manual segmentation, the archaeologist can decide to define a single scar due to some higher-level archaeological considerations such as the position of the point of percussion. For the purpose of quantitative research (for example, clustering and classification of tools from several assemblages), which is the main goal of our work, we believe that the quality of our scar segmentation is sufficient, since it is performed in an objective and consistent manner on all scanned objects and it is mostly consistent with the archaeological scars and ridges definition.

We are now integrating the automatic segmentation algorithm in a computer code which will be used for documenting lithic tools in archaeological reports and articles. The purpose is to enable the archaeologist to mark selected scars and ridges on the 3D model, and produce 2D views of the artifacts which include the marked features. The markings should reflect the degree of confidence or importance attributed to the feature, such as i.e., using thicker lines to delineate the more outstanding ridges. Moreover, it will enable the archaeologist to introduce his/her judgment in cases of archaeologically uncertain situations as for example when two neighboring scars should be merged to a single one.

Future work can aim at enhancing the scar segmentation by introducing constraints or regularization terms on the shape of the scars (for example, requiring each scar to be convex) or by analyzing the surface of each scar. Detection of the conchoidal ripples on the surface of the scars can be performed

by means of Fourier analysis. This can enhance both the quantitative description of the scar and the generated drawing.

As a next step towards utilization of the quantitative scar and ridge features, we plan to apply machine learning methods to automatically cluster and classify stone tools from several sites and from several periods.

Acknowledgements

We would like to acknowledge support from the Israel Science Foundation Center of Excellence (Grant 300/6) and from the Faculty of Humanities and the R&D Authority of the Hebrew University.

References

- Bar-Yosef, Ofer, and Philip Van Peer. 2009. "The Chaîne Operatoire Approach in Middle Paleolithic Archaeology." *Current Anthropology* no. 50 (1):103-131. doi: doi:10.1086/592234 %U <http://www.journals.uchicago.edu/doi/abs/10.1086/592234>
- Boykov, Yuri Y, and Marie-Pierre Jolly. 2001. Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images. In *IEEE International Conference on Computer Vision*.
- Clarkson, Chris, Lucio Vinicius, and Marta Mirazon Lahr. 2006. "Quantifying Flake Scar Patterning on Cores using 3D Recording Techniques." *Journal of Archaeological Science* no. 33 (1):132-142.
- Cohen-Steiner, David, and Jean-Marie Morvan. 2003. Restricted Delaunay Triangulations and Normal Cycle. In *Proceedings of the nineteenth annual symposium on Computational Geometry*. San Diego, California, USA: ACM.
- Davis, Zachary J., and John J. Shea. 1998. "Quantifying Lithic Curation: An Experimental Test of Dibble and Pelcin's Original Flake-Tool Mass Predictor." *Journal of Archaeological Science* no. 25 (7):603-610.
- Eren, Metin I., Manuel Dominguez-Rodrigo, Steven L. Kuhn, Daniel S. Adler, Ian Le, and Ofer Bar-Yosef. 2005. "Defining and Measuring Reduction in Unifacial Stone Tools." *Journal of Archaeological Science* no. 32 (8):1190-1201.
- Eren, Metin I., and C. Garth Sampson. 2009. "Kuhn's Geometric Index of Unifacial Stone Tool Reduction (GIUR): does it measure missing flake mass?" *Journal of Archaeological Science* no. 36 (6):1243-1247.
- Golovinskiy, Aleksey, and Thomas Funkhouser. 2008. "Randomized Cuts for 3D Mesh Analysis." *ACM Trans. Graph.* no. 27 (5):1-12. doi: 10.1145/1409060.1409098.
- Grosman, Leore, Smikt, Oded, and Uzy Smilansky. 2008. "On the Application of 3-D Scanning Technology for the Documentation and Typology of Lithic Artifacts." *Journal of Archaeological Science* no. 35:3101-3110.
- Grosman, Leore, Gonen Sharon, Talia Goldman-Neuman, Oded Smikt, and Uzy Smilansky. 2011. "Studying Post Depositional Damage on Acheulian Bifaces using 3-D Scanning." *Journal of Human Evolution* no. 60 (4):398-406.
- Jackson, J. Edward. 2004. *A User's Guide to Principal Components*: John Wiley & Sons, Inc.
- Jolliffe, Ian T. 2002. *Principal Component Analysis*, 2nd Edition: Springer.
- Koenderink, Jan J. 1990. *Solid Shape*: The MIT Press.
- Kuhn, Steven L. 1990. "A Geometric Index of Reduction for Unifacial Stone Tools." *Journal of Archaeological Science* no. 17 (5):583-593.
- Mangan, Alan P, and Ross T Whitaker. 1998. "Surface Segmentation using Morphological Watersheds." *IEEE Visualization 1998 Late Breaking Hot Topics*.
- Meignen, Liliane, Jean-Michel Geneste, and Eric Boëda. 1990. "Identification de Chaines Operatoires Lithiques du Paleolithique Ancien et Moyen." *Paleo* no. 2:43-80.
- Melnik, Sergey, Hector Garcia-Molina, and Erhard Rahm. 2002. Similarity Flooding: A Versatile Graph Matching Algorithm and its Application to Schema Matching. In *International Conference on Data Engineering*, 2002: IEEE Computer Society.

- Ohbuchi, Ryutarou, Tomo Otagiri, Masatoshi Ibato, and Tsuyoshi Takei. 2002. Shape-Similarity Search of Three-Dimensional Models using Parameterized Statistics. In 10th Pacific Conference on Computer Graphics and Applications: IEEE Computer Society.
- Sellet, Frederic. 1993. "Chaîne Operatoire; the Concept and its Applications." *Lithic Technology* no. 18 (1-2):106-112.
- Sethian, James A. 1999. *Level Set Methods and Fast Marching Methods Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, Cambridge Monograph on Applied and Computational Mathematics: Cambridge University Press.
- Shamir, Ariel. 2008. "A Survey on Mesh Segmentation Techniques." *Computer Graphics Forum* no. 27 (6).
- Siek, Jeremy, Lee-Quan Lee, and Andrew Lumsdaine. 2002. *The Boost Graph Library: User Guide and Reference Manual*: Addison-Wesley.
- Tangelder, Johan, and Remco Veltkamp. 2008. "A Survey of Content Based 3D Shape Retrieval Methods." *Multimedia Tools and Applications* no. 39 (3):441-471. doi: 10.1007/s11042-007-0181-0.
- Yoshizawa, Shin, Alexander Belyaev, and Hans-Peter Seidel. 2005. Fast and Robust Detection of Crest Lines on Meshes. In *Proceedings of the 2005 ACM Symposium on Solid and Physical Modeling*. Cambridge, Massachusetts: ACM.
- Zhang, Cha , and Tsuhan Chen. 2001. Efficient Feature Extraction for 2D/3D Objects in Mesh Representation. In *IEEE International Conference on Image Processing*.