Title:
A Study on Junction Region Recognition for Shape Similarity Metric of Stone Tools Using 3D Measured Point Cloud

Authors:
Altansoyombo Enkhbaatar, s3321004@iwate-u.ac.jp, Iwate University, Iwate, Japan
Mengbo You, ymb@iwate-u.ac.jp, Iwate University, Iwate, Japan
Kouichi Konno, konno21@iwate-u.ac.jp, Iwate University, Iwate, Japan

Keywords:
Point Cloud, Feature Points, Edge Points, Junction Region, Stone Tool, Flake Surfaces

DOI: 10.14733/cadconfP.2025.71-76

Introduction:

Numerous stone tools are unearthed in archaeological excavations, providing essential evidence for studying human activity. Stone tools, crafted by hammering and polishing, were used as cutting tools or weapons and are key artifacts of prehistoric societies, offering insights into past cultures. 3D point clouds, which represent surface shape through spatial data points, have become essential in fields such as archaeology[12], architecture[1], and art[9].

Erdenebayar S. et al.[8] proposed a method for automatically extracting feature lines from stone tools represented by point clouds, aiming to reduce the time and expertise required for scale drawing. Their method uses the Mahalanobis distance metric to identify outline points and surface variation analysis to detect potential features, refined by Laplacian smoothing. Their method[8] extracts only edge shapes, which is insufficient information for shape matching. Edge relations such as junctions may be required for accurate shape matching.

Edge points lie along the boundary edges of the flake surfaces on a stone tool. Detecting junction regions, where edges share a common vertex, remains challenging due to noise and ambiguous geometry. If junction regions are not accurately recognized, the extracted features may fail to reflect the true shape of the tool, leading to incorrect shape similarity assessments. In previous work[2], edge points and junction regions were extracted from 3D point cloud, unfortunately, accuracy suffered from low-density points.

This paper presents a new method for recognizing junction regions, essential for identifying the distinct characteristics of stone tools. It combines feature line extraction[8] and edge detection methods to identify junction regions accurately. We evaluate our method by comparing extracted features with peakit[7] images of the stone tools.

Related works:

Feature extraction techniques can be divided into point-based and mesh-based methods[4]. Our method adopts a point-based approach, focusing on detecting junction regions where multiple sharp edges converge in the point cloud. Despite progress in curvature approximation, extracting junction regions accurately remains challenging, especially in complex structures.

Proceedings of CAD'25, Shenzhen, China, June 23-25, 2025, 71-76

Reverse engineering main steps involve: 1)capturing and preprocessing, 2)segmentation, 3)regions of interest, 4)classification of the regions, 5)surface generation, 6)CAD reconstruction. For stone tools, we use steps 1-4, but steps 5-6 are difficult due to the complexity and ambiguity of flake surfaces.

Yu Z. et al.[13] proposed a method for extracting and visualizing structural information from point clouds using immersive interfaces. The method[13] focused on extracting faces from unorganized point clouds and extraction of edge points based on which face they are adjacent to. Their method of extracting vertices is not suitable in stone tool, because of many small surfaces that give noise. In our method, prepare a point cloud with resampling to reduce noise before the region-growing algorithm, which improves the accuracy of edge point extraction.

Liu Y. et al.[10] introduced learning-based segmentation combined with geometric surface fitting to recover complete CAD models Point2CAD[10], including edges, junctions, and freeform surfaces. Their approaches are primarily optimized for industrial CAD applications with an F-score[10] of surface 0.947, edges 0.816, and corners 0.736. However, the irregular surfaces of stone tools are different from industrial objects, which makes it difficult to apply their methods directly.

Buonamici F. et al.[6] categorize reverse engineering approaches into surface-based, feature-based, and hybrid modeling, and evaluate existing tools for their strengths and limitations in handling complex geometries.

Zhang Z. et al.[14] propose a comprehensive CAD reconstruction pipeline that guarantees high-precision reconstruction while markedly diminishing the requirement for manual operations. Their framework includes the automatic determination of extrusion heights the algorithm that autonomously computes the height of extruded features, enhancing the accuracy and speed of the reconstruction process.

Given these existing limitations, our study proposes a novel junction region recognition method to enhance structural analysis, offering accurate edge connectivity representation for improved stone tool classification and similarity metrics. The methods [13], [11], and [5] all use feature extraction to work with and extract faces, edges, and contour points. However, our novel approach using junction regions for stone tool identification has never been used before.

Proposed Method:

Junction regions are points in a point cloud where three or more segmented regions or edge point pairs converge. Edge points represent the convex shape of an object, and refer to a point that lies on the edge of a flake surface. These points are essential for extracting junction regions and identifying stone tools.

Our proposed method identifies edge points and junction regions and involves several key steps: 1) resampling a point cloud by the VoxelGrid filter for calculating cost, 2) applying a region-growing algorithm, 3) extracting edge features, and 4) extracting junction regions, 5) constructing topological structure. Our method assumes that a point cloud of the stone tool is separated front and back sides.

A region-growing algorithm of [5] method works with full stone tool point cloud and makes segmentation by the looped ridge lines. The algorithm has been modified to function with half-stone and recognize the stone tool's outline as a limit of the point cloud, creating a loop with ridge lines. A region-growing algorithm is used to identify and segment the smooth areas in the point cloud that represent the flake surface.

Fig.1(a) shows a stone tool, and Fig. 1(b) shows the stone tool point cloud located on the x,y,z coordinate system. Our method assumes that the half stone tool point cloud is located on a flat surface indicated by the x,y plane as shown in Fig. 1(b). The different flake surface colors show the result of region-growing and red points show the boundary of each region.

Data Structure:

This section explains the data structure used for efficient junction region extraction. Each point $\mathbf{p}_i$ as

Proceedings of CAD'25, Shenzhen, China, June 23-25, 2025, 71-76

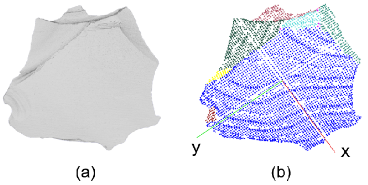

Fig. 1: Stone tool point cloud on coordinate system.

Point set	
p_i	(x, y, z), (RGB), (n_x, n_y, n_z), InP/BP, notEP/EP, EPP_id, notJP/JP, JP_ids

Fig. 2: Data structure of point set.

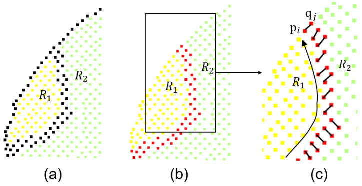

Fig. 3: Edge points between R_1 and R_2.

Region set	
R_i	(R, G, B), Boundary Point List, Edge Point Pair List

Fig. 4: Data structure of regions.

shown in Fig.2, includes coordinates x, y, z, RGB color values, normal vectors $\mathbf{n}_x$, $\mathbf{n}_y$, $\mathbf{n}_z$, a binary flags indicating boundary, edge, and junction point status, an edge point pair ID (EPP_id) if it is an edge point, and junction point IDs (JP_ids) if it is part of a junction region.

The data structure of the region R_i shown in Fig.4, includes the region's RGB color values, a list of its boundary points (Boundary Point List), and a list of edge point pairs associated with the nearest neighboring region (Edge Point Pair List).

Extract edge features:

To extract the edge feature re-sample the point cloud and simplify using the VoxelGrid filter and remove the back side of the stone tool by point cloud z-axis to reduce noise for the next step.

To extract edge points: 1) use a region growing algorithm with a correction process of region boundaries[5] and extract edge points by choosing points where the angle difference between the normal vectors of adjacent points exceeds a specified threshold, as described in [2]. 2) arrange boundary points of each region in the counterclockwise direction.

Edge points represent the boundary of the flake surface of the stone tool. In other words, edge points are located between two regions within a point cloud, as shown in Fig. 3(a), which shows edge points between R_1 and R_2.

Edge point pair is determined as following steps:

1. Extract boundary points for each region and make a boundary point list into the region structure. In addition, store boundary points flag into a point set structure. For example all boundary points of R_1 are shown in Fig.3(a)

2. Traverse boundary point list of R_1 and find edge point for each boundary point. Candidate of edge point is boundary point lists without R_1's list. For example, edge points of R_1, denoted $\mathbf{p}_i$, and edge points of R_2, denoted $\mathbf{q}_j$, are shown in Fig.3(c).

3. Extract edge points of R_1 and R_2 and make edge point list for intermediate information.

4. Calculate the average distance between $\mathbf{p}_i$ and $\mathbf{q}_j$, derived by step 3 denoted by ϵ.

5. If the distance is less than the average distance ϵ, make a point pair ($\mathbf{p}_i$, $\mathbf{q}_j$) as shown in Fig.3(b).

6. After making point pairs ($\mathbf{p}_i$, $\mathbf{q}_j$) add them to the edge point pair list as shown in Fig.4

Junction region recognition:

Junction regions in a point cloud represent areas where three or more edge point segments meet. They are crucial for understanding the structure and shape of stone tools, offering insight into the connectivity between flake surfaces and contributing to shape similarity metrics. As key surface markers, they help characterize stone tools geometry.

How to find the junction is as following steps:

1. Traverse edge point pair list that belongs to region R_i. For example, the edge point pair list of R_1 is traversed as shown in Fig.5.
2. Find the edge point pair where the region of the pair is changed. For example, ($\mathbf{p}_3$, $\mathbf{q}_4$) to ($\mathbf{p}_4$, $\mathbf{r}_3$) as shown in Fig.5 is a place of changing the region pair. If the changing pair is detected, the edge point pairs are added to the edge point pair of junction list.
3. Get junction region by ordering junction points in a clockwise direction, such as ($\mathbf{p}_i$, $\mathbf{p}_{i+1}$), ($\mathbf{p}_{i+1}$, $\mathbf{r}_k$), ($\mathbf{r}_k$, $\mathbf{r}_{k+1}$), ($\mathbf{r}_{k+1}$, $\mathbf{q}_j$), ($\mathbf{q}_j$, $\mathbf{q}_{j+1}$), ($\mathbf{q}_{j+1}$, $\mathbf{p}_i$). For example, Fig.6 shows the junction region whose start point is p3, represented by ($\mathbf{p}_3$, $\mathbf{p}_4$), ($\mathbf{p}_4$, $\mathbf{r}_3$), ($\mathbf{r}_3$, $\mathbf{r}_4$), ($\mathbf{r}_4$, $\mathbf{q}_3$), ($\mathbf{q}_3$, $\mathbf{q}_4$), ($\mathbf{q}_4$, $\mathbf{p}_3$)
4. Generate a list of all junction regions by traversing each edge point pair in R_i.
5. Repeat the process for every region edge point pair to find all junction regions.

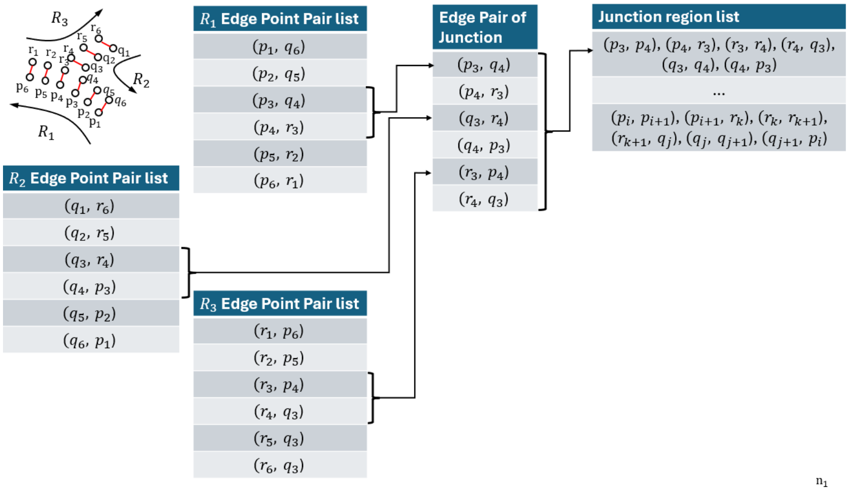

Fig. 5: An example of the extracted junction region.

Topology map using junction regions:

A topology map represents the structural connectivity of an object by identifying and linking key feature points. For stone tools analyzed through 3D measured point clouds, junction regions serve as critical markers that define the connections between surfaces. The following steps outline the process of constructing a topology map using junction regions.

How to construct topology map:

1. Visualize the junction region connections using a graph representation.
2. Nodes represent junction regions $\mathbf{n}_i$ and regions R_i, while edge points represent adjacency and shared boundaries as shown in Fig.6(b).
3. The resulting topology map provides a clear structure of how regions and junctions are connected as shown in Fig.6(c).

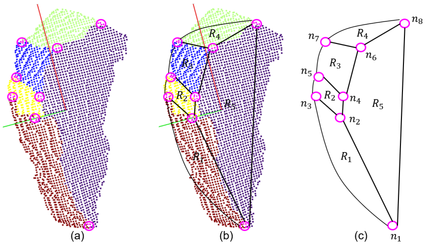

Fig. 6: Topology map.

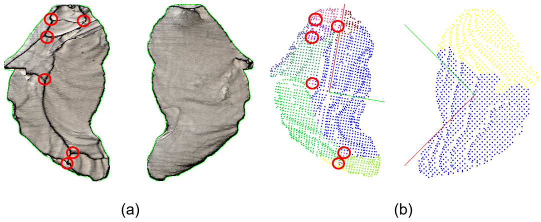

Fig. 7: Stone tool number 45 from Group 3 with extracted junction regions.

Results:

The proposed method includes junction region and edge point extraction. The proposed method was implemented in C++ programming language used for the recognition and visualization of the stone tools with Visual Studio 2019 and Windows 10 Pro Education. The experiment was performed on an Intel Core i5-11400 2.60 GHz machine with 16 GB of RAM.

The 3D scan data utilized in this study were acquired through a 3D surface reconstruction technique employing a four-directional measurement machine [3].

The proposed method was implemented on all 3D stone tools of Group 3, flakes with id numbers 39-46, resulting in the extraction of their junction points. Following this, top and bottom surfaces of the flakes with id numbers 39-46 were extracted and the results of the edge points and junction regions. The results were verified by comparing them with peakit[7] images of the stone tools.

As shown in Fig. 7, Fig. 7(a) is the peakit[7] image of the stone tool, and Fig. 7(b) is a stone tool with a point density of 5308 points. Fig. 7(b) shows the result of our method with the same number of junction regions as Fig. 7(a). Simplifying the point cloud into a regular point cloud improves the recognition of flake surfaces and produces more accurate results.

We compared the number of junction regions identified in Group 3 stone tool point clouds and peakit[7] images. While peakit[7] images contained 27 junction regions, our method identified 29. Upon examining their positions, we confirmed that 27 junction regions matched those in the peakit[7] images. However, the two junction regions are different. resulting in 93% of the junction regions extracted. This demonstrates our method's effectiveness in extracting junction regions.

Conclusions:

This paper proposed edge feature and junction region extraction from a 3D point cloud of stone tool. Our method offers a novel approach using junction regions derived from extracted edge point sequences. We verify the result of edge points and junction regions by observation. In future work, we intend to apply our method to stone tool identification.

Acknowledgement:

A part of this work was supported by JSPS Grant Number JP22K00998, and JP25K00537.

References:

[1] Abreu, N.; Pinto, A.; Matos, A.; Pires, M.: Procedural point cloud modelling in scan-to-bim and scan-vs-bim applications: A review. ISPRS International Journal of Geo-Information, 12(7), 2023. ISSN 2220-9964. http://doi.org/10.3390/ijgi12070260.

[2] Altansoyombo, E.; Mengbo, Y.; Kouichi, K.: Edge feature extraction for shape similarity metric of stone tool by 3d measured point cloud. In Nicograph International 2024, vol. 1, 32–35, 2024. http://doi.org/10.1109/NICOInt62634.2024.00015.

[3] Altantsetseg, E.; Muraki, Y.; Chiba, F.; Konno, K.: 3d surface reconstruction of stone tools by using four-directional measurement machine. International Journal of Virtual Reality, 10, 37–43, 2011. http://doi.org/10.20870/IJVR.2011.10.1.2800.

[4] Altantsetseg, E.; Muraki, Y.; Matsuyama, K.; Chiba, F.; Konno, K.: Feature extraction and modification for illustrating 3d stone tools from unorganized point clouds. The Journal of the Society for Art and Science, 12(1), 36–47, 2013. http://doi.org/10.3756/artsci.12.36.

[5] Amgalan, A.; You, M.; Konno, K.: A study on automatic flake surface segmentation of stone tools by calculating shape features. In International Workshop on Advanced Imaging Technology (IWAIT) 2023, vol. 12592 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, 2023. http://doi.org/10.1117/12.2666684.

[6] Buonamici, F.; Carfagni, M.; Furferi, R.; Governi, L.; Lapini, A.; Volpe, Y.: Reverse engineering modeling methods and tools: a survey. Computer-Aided Design and Applications, 15, 1–22, 2017. http://doi.org/10.1080/16864360.2017.1397894.

[7] Chiba, F.; Yokokoyama, S.; Kaneda, A.; Konno, K.: Development of network-type archaeological investigation system. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-5/W7, 99–103, 2015. http://doi.org/10.5194/isprsarchives-XL-5-W7-99-2015.

[8] Erdenebayar, S.; Konno, K.: Feature line extraction of stone tool based on mahalanobis distance metric. The Journal of the Society for Art and Science, 18(1), 51–62, 2019. http://doi.org/10.3756/artsci.18.51.

[9] Ivsic, L.; Rajcic, N.; McCormack, J.; Dziekan, V.: The art of point clouds: 3d lidar scanning and photogrammetry in science & art. In ACM Association for Computing Machinery, vol. 1, 404–411, 2021. http://doi.org/10.1145/3452853.3452855.

[10] Liu, Y.; Obukhov, A.; Wegner, J.; Schindler, K.: Point2cad: Reverse engineering cad models from 3d point clouds. In Conference on Computer Vision and Pattern Recognition (CVPR). Seattle, WA, USA, 2024. http://doi.org/10.1109/CVPR52733.2024.00361.

[11] Sawada, Y.; Kinoshita, T.; Renchin-Ochir, A.; Chiba, F.; Konno, K.: Stone tool identification method based on measured points by rgb-d camera and points of stone tool database. The Journal of the Society for Art and Science, 21(4), 213–224, 2022. http://doi.org/10.3756/artsci.21.213.

[12] Yang, S.; Hou, M.; Li, S.: Three-dimensional point cloud semantic segmentation for cultural heritage: A comprehensive review. Remote Sensing, 15(3), 548, 2023. http://doi.org/10.3390/rs15030548.

[13] Yu, Z.; Lin, T.; Gumhold, S.; McGinity, M.: An immersive method for extracting structural information from unorganized point clouds. In Nicograph International 2024, vol. 1, 24–31, 2024. http://doi.org/10.1109/NICOInt62634.2024.00014.

[14] Zhang, Z.; Zhao, M.; Shen, Z.; wang, Y.; Jia, X.; Yan, D.: Interactive reverse engineering of cad models. Computer Aided Geometric Design, 111, 2024. http://doi.org/10.1016/j.cagd.2024.102339.