Parameter Extraction and Performance Analysis of 3D Surface Reconstruction Techniques

Richha Sharma[1], Pawanesh Abrol[2]
Dept. of CS & IT, University of Jammu, Jammu, India[1, 2]

***Abstract*—Digital image-based 3D surface reconstruction is a streamlined and proper means of studying the features of the object being modelled. The generation of true 3D content is a very crucial step in any 3D system. A methodology to reconstruct a 3D surface of objects from a set of digital images is presented in this paper. It is simple, robust, and can be freely used for the construction of 3D surfaces from images. Digital images are taken as input to generate sparse and dense point clouds in 3D space from the detected and matched features. Poisson Surface, Ball Pivoting, and Alpha shape reconstruction algorithms have been used to reconstruct photo-realistic surfaces. Various parameters of these algorithms that are critical to the quality of reconstruction are identified and the effect of these parameters with varying values is studied. The results presented in this study will give readers an insight into the behaviour of various algorithmic parameters with computation time and fineness of reconstruction.**

Keywords—3D reconstruction; point cloud; feature detection; feature matching

I. Introduction

The 3D reconstruction process can generate the photo-realistic 3D surfaces of objects using their digital images to generate their sparse and dense 3D point clouds. The generated point cloud (PC) data is to be transformed into a substantial digital representation of the scanned objects to enable its use in numerous application fields including Computer-Aided Design, Medical Imaging, Reverse Engineering, Virtual Reality, and Architectural Modeling [1]. Since object surfaces in point clouds are to be reconstructed, various reconstruction approaches have been proposed over years to fast-track the research in this area. Earlier, 3D reconstruction was mainly applied in the field of Industrial and Architectural design, Animation industry, Medical Modeling, Robotics, etc. but with the advent of Machine Learning and 3D vision technology, the application range has widened to other fields like autonomous driving, remote sensing, 3D printing, and even online shopping. The sample points can be generated from the laser scanner, photogrammetry technique, or some mathematical function. In most cases, the sample points will elucidate the shape and topology of the object's surface. Algorithms for surface reconstruction (SR) are applied to the sample points, and to obtain a 3D model of the object [2]. 3D reconstruction can usually be done in three manners1) Using 3D modeling software like AutoCAD, FreeCAD, 3DMAX, etc. 2) Using 3D scanners to obtain models in real-time. 3) Using digital images with the help of RGB-D cameras. Close-range photogrammetry has been dealing with manual or programmed picture measurements for accurate 3D modelling. Although many applications utilize 3D scanners as a regular source of data input, image-based modelling is still the most thorough, affordable, portable, adaptable, and widely-used method. It is a low-priced and easy way to obtain depth images. Point cloud obtained from RGB-D data is of middle density with the advantages of being cheap and flexible along with the color information. It is limited to close-range objects with moderate accuracy in applications like Indoor reconstruction, object tracking, human pose recognition; etc. Laser-scanner-based 3D reconstruction of an object is quick and accurate and provides realistic textured 3D surfaces of a scene or object being studied. Because of the high cost and less availability, researchers have been inspired to use photogrammetry to produce realistic 3D surfaces from an object's image. It is used for mapping and modeling common indoor household objects. With the ease of availability of digital cameras and computers in homes and organizations, the processing cost of image-based reconstruction has diminished. The work in this paper evaluates the efficacy of various surface reconstruction techniques. It proposed a framework for reconstructing 3D surfaces that can generate a 3D model of typical indoor objects from their digital photos. Correspondence matching is done over the image sets to obtain a dense set of points that are projected to 3D space. Furthermore, by applying SR techniques to the resulting point cloud, a photo-realistic surface is formed.

The paper is structured as: Section II presents a summary of surface reconstruction algorithms and work related to 3D reconstruction. Section III presents the framework of the used methodology along with a description of the selected algorithms used for reconstruction. Section IV provides a discussion of the experiment with the obtained results. The conclusion and future outline of the work is given in Section V.

II. Related Work

Reconstructing a 3D surface from an order of digital images comprising a scene/object is a demanding and significant exercise in computer vision. Over the years, authors have employed several SR approaches to create a precise and photorealistic 3D surface from an image or group of images depicting a variety of objects. 3D reconstruction from images (photogrammetry), is a long-standing technique whose potential could only be seen after the development of computers and digital photographs and is widely used in entertainment to create scenarios for games, acquire human expressions, scanning products for commercials, etc.

TABLE I. DESCRIPTION OF SURFACE RECONSTRUCTION ALGORITHMS

Authors	Year	Algorithm	Description
Edelsbrunner and Mucke [3]	1994	Alpha shape algorithm	It is a heuristic approach and performs well in uniform sampling and has the complexity of $O(n^2)$
Bernardini *et al* [4]	1999	Ball-Pivoting algorithm	Forms a 3D mesh by joining the triangles created by rolling the ball through points. It is data-driven and is sensitive to noise and is not expensive in terms of time and memory and results are of favorable quality.
Amenta *et al.* [5]	2000	The Crust algorithm	Delaunay Triangulation is estimated employing the Voronoi diagrams over the point cloud forming a smooth surface from unstructured points. It is robust and does not require a hole-filling mechanism. It has the complexity of O (n log n)
Amenta *et al.* [6]	2001	Power Crust algorithm	It gives approximated polygon surface with the amalgamation of median balls. The algorithm is robust and no hole-filling structure is needed in reconstruction as the factual geometry of the object is captured.
Amenta *et al.* [7]	2001	Cocone algorithm	Delaunay-based algorithm for surface reconstruction with added post-filtering restrictions. It produces an incomplete surface with a low-density point cloud.
Dey and Goswami [8]	2003	Tight Cocone algorithm	The reconstructed surface uses the 'In' marked computed Cocone tetrahedral triangles. This method is not robust to a noisy and sparse point cloud and has a time complexity of $O(n^2)$.
Dey and Goswami [9]	2004	Robust Cocone algorithm	It can reconstruct surfaces from noisy points and is also considered a labeling algorithm as it marks cocone triangles as "in' and 'out'.
Kazhdan *et al.* [10]	2006	Poisson Surface Reconstruction	The watertight surface is recreated with Marching Cubes (MC) algorithm and stored in an octree. The algorithm is resilient to noise.
Kazhdan *et al.* [11]	2013	Screened Poisson Surface Reconstruction	It incorporates point constraints in solving the screened Poisson equation to compose the solved indicator function more accurately.

A. Surface Reconstruction Techniques

From input data, which is classically obtained from 3D scanners or photogrammetry as scattered points in 3D space, the process of SR creates a 3D surface of the object [12]. The surface is then often recreated in the triangulated form (points joined in the form of triangles with sharing edges and/or vertices), depending upon the type of point data employed [13] making a visible, 3D model of a real object. In the field of computer graphics, numerous surface reconstruction techniques have been developed and used. In this paper, a few of the well-known surface reconstruction algorithms are briefly discussed and presented in Table I. A brief description of a few of the algorithms is given in hierarchical order in terms of the year in which they were introduced.

The primary input of the SR process is a digital image set of the object being reconstructed. This is done by detecting features from the image that will help in the process of reconstruction. These feature points are searched for correspondence; projection of the same 3D point of the object across the images [14]. Two feature detection algorithms have been used. SURF (Speeded up Robust Features) focuses on blob-like structures in the image [15]. It is fast, robust, and invariant to rotation and scale which makes it a good feature detection algorithm [16]. Min Eigen Features detect the corners (locations with steep intensity variations in any direction) in an image. Feature matching, generally known as image matching is performed in many computer vision applications like image registration, object identification, and object recognition. It consists of detecting a set of interest points each associated with image descriptors from image data. After the extraction of features from the image set, corresponding feature points are matched. Matching shows that features are from the corresponding locations from completely different images.

There are certain issues and constraints while dealing with point data. The unstructured-ness of data makes it difficult to implement and reconstruct the surface because of the non-uniformity of input data. Dataset size is another major constraint and can lead to a tradeoff between accuracy and performance. Performance has been always a bottleneck in all computer-based applications in terms of time, memory efficiency accuracy, and completeness.

III. METHODOLOGY

A digital image sequence of the scene or item being recreated serves as the primary input to the SR process. Fig. 2 demonstrates the process used to recreate an object's 3D surface from a series of digital images. A set of distinguished feature points are found in the image sequence and used for correspondence matching across the photos to begin the 3D reconstruction process. The exhaustive matching of all image points is considerably reduced by feature detection, making it computationally cheap. The robust feature points are also easily recognizable and image transformation-invariant. Consequently, by identifying feature points in the photos, the computation time for correspondence matching is significantly decreased.

The dataset used in the work comprises images of irregular indoor household objects to be reconstructed. It consists of first-hand image data comprising 4 image sets taken from an android phone having a (16+20) MP dual camera placed at a distance of 25 cm from the object. Images were preprocessed to a standard form for further use and Table II shows the parameter values of preprocessed images. Fig. 1 presents the image dataset used as input in this paper for the implementation purpose and the attributes of the images are given in Table II. Image features were detected and extracted from all the images containing the object of interest using the SURF (Speeded Up Robust Features) algorithm. The SURF-identified feature points are readily distinct and invariable to image transformations [17] and hence are then used for correspondence matching.

TABLE II. PARAMETER VALUES OF IMAGES USED FOR IMPLEMENTATION

Parameter	Value
Size of image	1520 x 2048
Min-Quality	0.1
Confidence	99.90
Roi (x, y, width, height)	[30, 30, 2018,1490]

Fig. 1. The input image dataset.

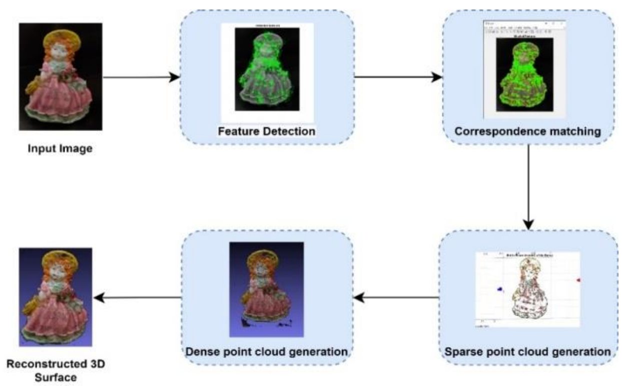

Fig. 2. Methodology of 3D surface reconstruction.

An approximate-nearest-neighbor (ANN) search discovers the matched point pairs. An image pair of the object with enough overlap has been selected for the purpose of matching. The features of the first image were inserted in the k-d tree which acts as queries to the feature points of other images of the pair. The k-d tree has been used for the efficient search of K-nearest neighbors of the feature points. The accuracy of the paired points is then tested using a RANSAC-based estimation. Matches that do not satisfy the basic matrix equation are discarded as non-fits.

SFM has been applied to build sparse point clouds (SPC) from huge image sets. In SFM, correspondence matching finds the matched points from the image set that are triangulated to obtain the 3D location of these points. The obtained point clouds are not dense and do not comprehend the object's geometry or specifics of the scene under reconstruction, hence are inadequate for 3D modelling [18]. Thus, dense correspondence matching must be done to build dense point clouds (DPC) from sparse points of the image sets [19], precisely unfolding the specifics of the scene/object being reconstructed and as shown in Fig. 3.

The pronounced methodology for 3D reconstruction used in this paper is grounded on several well-known algorithms in the fields of correspondence matching, SFM, SR, and so on. Rather than delving into the thorough mathematics of these techniques, the study focused on the application of existing techniques to create an inexpensive, robust, and simple answer to the problem of 3D reconstruction. Because of its robustness, swiftness, and capability to reconstruct surfaces with irregular geometry, the approach described in this paper is easily applicable to industry. The photogrammetry-based reconstruction method conversed here can efficiently substitute expensive laser scanners as prime 3D reconstruction methods in numerous application areas due to its little workforce requirement and reduced functioning cost. Three surface reconstruction algorithms were tested on a PC generated from digital images. The performance of these reconstruction algorithms is evaluated in terms of time, resource usage, reconstruction quality, and so on. The following section provides brief overviews of these algorithms.

Fig. 3. 3D point cloud of object Image (row1 sparse point cloud, row 2 dense point cloud).

A. Poisson Surface Reconstruction (PSR)

PSR intends to create a 3D mesh from a DPC by minimalizing the difference between the surface normal directions of the recreated surface and the 3D points [10] (Fig. 4). The surface normal of each point's surface for the input point cloud is found by computing the eigenvectors over the k-nearest neighbor of each point, and an octree of pre-defined depth is used for storing the reconstructed surface. A 3D indicator function x is defined as having the value of 1 inside and 0 outside the surface. ∇x (grad of the indicator function) is a vector field comprising non-zero values at the points near the surface where x can be varied.

For the points where ∇x is not zero, the value of ∇x equates to the surface normal vector of corresponding points and is stated as:

$$\nabla x = V \tag{1}$$

On both sides of the equation, the divergence operator is applied to modify the equation into a standard Poisson problem and can be articulated mathematically as:

$$\Delta x \equiv \nabla \cdot \nabla x = \nabla \cdot V \tag{2}$$

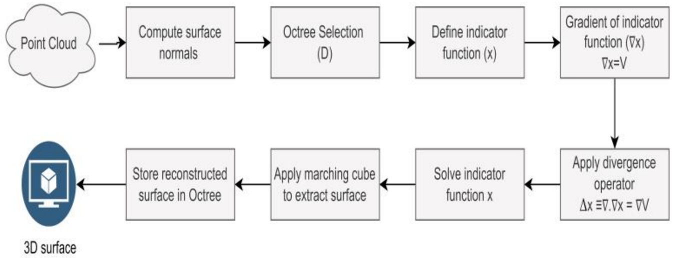

Fig. 4. Flow chart of Poisson surface reconstruction.

The indicator function (x) is represented in 3D space for solving the Poisson equation. Octree represents the indicator function and its leaf nodes store x at various points over the reconstructed surface. The MC algorithm extracts the surface from the indicator function. The algorithm works by marching through the points in a 3D point cloud following the indicator function having a value of 1 close to the surface and 0 afar from the surface. By interpolation of points among the vertices of the cube during the traversal of the octree, a triangulated 3D mesh is generated and stored in the octree. The quality of reconstruction, computation time, etc. of PSR depends upon several parameters like Octree depth (D), Samples-per-node (S_n), and Surface offsetting (S_{off}). An octree with depth D generates a 3D mesh having resolution ($2^D \times 2^D \times 2^D$) that rises with the rise in D value and hence, the memory usage also rises severely. S_n indicates the least number of points that are allocated to individual leaf nodes of octree by the MC algorithm. If the data is noisy, a greater number of points are allocated as the surface is interpolated with the entire data points. S_{off} specifies a correction value of the threshold for the reconstructed surface. S_{off} =1, no correction required, S_{off} <1 signifies internal offsetting, and for external offsetting S_{off} >1.

B. Alpha-shape

It contains points, edges, triangles, and tetrahedrons. A subgroup of the triangles from the Delaunay triangulation is carefully selected to obtain a surface. The initial triangle with the property of minimum area is considered as the 'seed' which is further used to propagate the whole surface [3]. A subset of the 3D Delaunay triangulation of P is the α-complex of a set of points (P) for a given value of the parameter, with the α-shape serving as its underlying space. As the value of α varies, different α-shapes can be generated. For α = 0 the α-shape is P itself and when α =∞, the resultant shape is the convex hull of the point set. Changing the value of α from 0 to ∞, different α-shapes can be attained.

C. Ball Pivoting Algorithm (BPA)

It is an effective surface reconstruction technique in which a ball of the pre-defined radius is rolled across the point cloud [4]. Triangular inter-connected 3D mesh is generated as the ball moves through the cloud, connecting the 3D points. The process is repeated until all the points are linked to a triangle. The algorithm is divided into two steps. The primary step is to find a seed triangle, and the second step is to expand the seed triangle to create a 3D surface, as shown in Fig. 5 and 6. For the input points, the surface normal for all points is computed and a point (p) is selected.

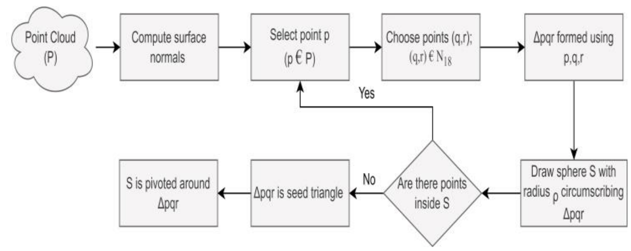

Fig. 5. Flow chart of ball pivoting algorithm (Step I).

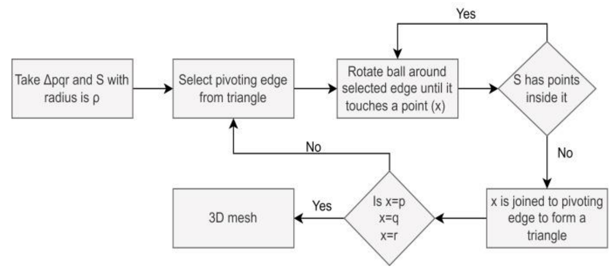

Fig. 6. Flow chart of ball pivoting algorithm (Step II).

A triangle is formed by picking two nearest neighbors' triangle if the sphere (S) has no point inside it else, a new point is chosen from the point cloud. The seed triangle is extended by rolling the ball that is pivoted to the chosen edge of Δpqr to find an alternative point in the point cloud. Another triangle is formed by joining the new-found point to the pivoting edge. The process is repeated until all the points have been navigated and linked by triangles to form a 3D mesh, as shown in Fig. 7. (q,r) of p such that (q,r) € N18 resulting in Δpqr and its circumsphere is searched for points. Δpqr is the seed triangle.

BPA is sensitive to variations in ball radius (ρ) which acts as a critical parameter in defining the number of reconstructed faces. A small radius causes the model to be vulnerable to input noise while a high value of radius can result in missing details creating holes in the generated surface. Another parameter is the angle threshold which is the maximum permissible angle between the active and the new edge constructed by rolling the ball. The rolling of the ball is stopped for the region where the angle threshold is achieved. An upsurge in threshold angle increases the computation time. The clustering radius is the lowest permitted distance between a recently added point and the active edge point. If there is less space separating two locations than the clustering radius, the two are combined. This is done to prevent processing DPC from consuming too much memory as a result of creating too many little triangular meshes.

The methodology defined and implemented in this study is based on well-known algorithms in the area of correspondence matching, structure-from-motion, surface reconstruction, etc. Instead of delving into the intricate mathematics involved in creating these algorithms, the focus here is on using these techniques to create a practical, reliable, and straightforward solution to the 3D reconstruction task. The photogrammetry-based 3D reconstruction strategy detailed here can successfully replace the pricey laser scanners as the principal

3D mapping method in industries due to its minimal labor demand and nearly zero running cost [20].

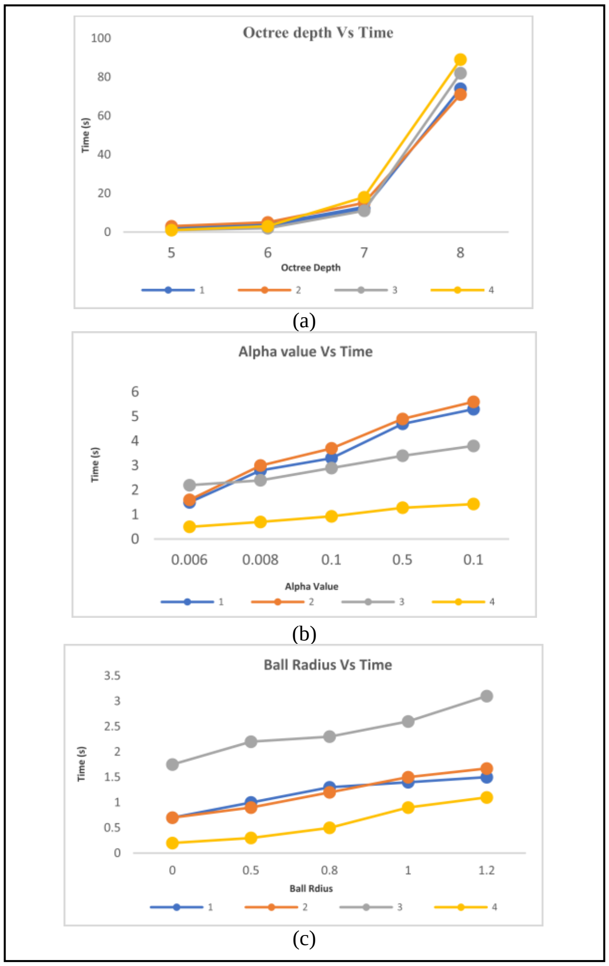

Fig. 7. Behavior of different parameters for selected algorithms with respect to time (a) PSR (b) Alpha Shape (c) BPA.

IV. Experimental Results

The experimental work has been performed on Intel Core (TM) i7 CPU with 20 GB RAM accompanied by NVIDIA GeForce 1050Ti. MATLAB 2021 on windows 10 has been used for implementation. To analyze how well the 3D reconstruction algorithms performed, some point clouds were produced from digital image sets of various objects. To create a realistic 3D surface of the acquired item, PSR, Alpha shape, and BPA algorithms have been applied to the point clouds. The topology surface, the number of faces and the amount of time the algorithm took to compute each 3D surface are all taken into consideration in the analysis. The parameters affecting the reconstruction process were inspected in detail, by altering the control parameters of respective algorithms in terms of the execution time and reconstruction fineness. Fig. 6 shows the graphs of selected algorithmic parameters with respect to time. The primary influencing parameter in PSR is Octree Depth (D), and execution time grows with depth as the resolution of the 3D mesh increases by $2^D x 2^D x 2^D$. It has been found that an increase in alpha value causes an increase in execution time in the Alpha shape technique, which is a critical parameter for the quality of the rebuilt surface. The ball radius is an important parameter in BPA, and it has been found that as grows in value, surface reconstruction takes longer to complete. The performance of selected reconstruction algorithms has been evaluated with the same input images and is compared in terms of how fine is the reconstruction. The final triangulated surfaces are investigated to measure the fineness of the surface. The more the number of faces, the finer is the reconstructed surface. The data in Table III shows the number of faces of the reconstructed surface for the corresponding input image for each of the algorithms. A graphical representation is given in Fig. 8. Among the selected algorithms, BPA reconstructs the minimum number of faces as it only creates a small collection of surfaces from point clouds. Alpha shape and PSR algorithms created more surfaces than BPA.

TABLE III. Number of 3D Reconstructed Faces of each Model based on Selected Algorithms

Model	PSR	Alpha Shape	BPA
1	138666	226709	36429
2	224968	208791	51778
3	240180	280285	78132
4	136416	91976	7756

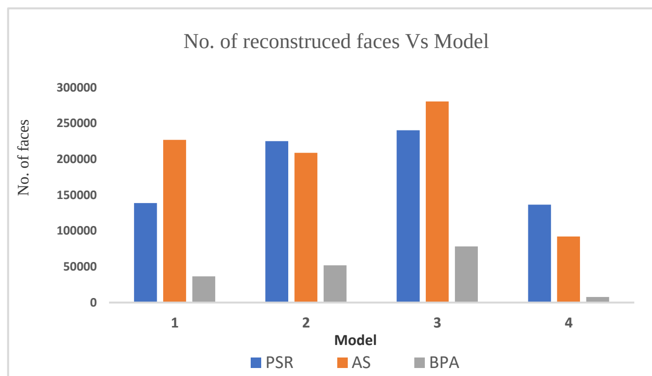

Fig. 8. Number of 3D Reconstructed Faces

V. Conclusion

A framework for the reconstruction of a 3D surface has been presented in the paper to reconstruct photorealistic 3D surfaces of irregularly formed objects from the set of their digital images to attain 3D point clouds that can be sparse, followed by dense clouds. The topology of the surface generated by these algorithms is based on the density of point clouds. Performance study of algorithms (PSA, Alpha shape, and BPA) has been carried out by observing their behavior with varying values of control parameters (octree depth, alpha value, and ball radius), on the amount of time taken and obtained quality of the reconstruction. Alpha shape and PSR algorithms performed better in creating more faces and hence a finer reconstructed 3D surface. To build a fine, feature-preserved, and accurate surface from the point cloud, several issues and constraints need to be considered. Noisy, sparse, and non-uniform points make the surface reconstruction process challenging. More research reconstructing finer surfaces using image features will be done in the future.

ACKNOWLEDGMENT

This research did not receive a specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

[1] S. P. Lim and H. Haron, "Surface reconstruction techniques: A review," *Artif. Intell. Rev.*, vol. 42, no. 1, pp. 59–78, 2014.

[2] A. Alqudah, "Survey of Surface Reconstruction Algorithms," *J. Signal Inf. Process.*, vol. 05, no. 03, pp. 63–79, 2014.

[3] H. Edelsbrunner and E. P. Mucke, "Three-dimensional Alpha Shapes 1 1 Introduction," *ACM Trans. Graph.*, vol. 13, no. 1, pp. 43–72, 1994.

[4] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin, "The ball-pivoting algorithm for surface reconstruction," *IEEE Trans. Vis. Comput. Graph.*, vol. 5, no. 4, pp. 349–359, 1999.

[5] N. Amenta, "The Crust Algorithm for 3D Surface Reconstruction from Scattered Points," *Proc. Conf. Comput. Geom. ({SCG} '99)*, pp. 423–424, 1999.

[6] N. Amenta, S. Choi, and R. K. Kolluri, "The power crust," *Proc. Symp. Solid Model. Appl.*, pp. 249–260, 2001.

[7] N. Amenta, S. Choi, T. K. Dey, and N. Leekha. A simple algorithm for homeomorphic surface reconstruction. Internat. J. Comput. Geom. & Appl., 12:125–141, 2002.

[8] T. K. Dey and S. Goswami, "Tight cocone: a water-tight surface reconstructor." *J. Comput. Inf. Sci. Eng.* 3, no. 4, pp. 302-307, 2003.

[9] T. K. Dey and S. Goswami, "Provable surface reconstruction from noisy samples." In *Proceedings of the twentieth annual symposium on Computational Geometry*, pp. 330-339. 2004.

[10] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson Surface Reconstruction," in *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, Cagliari, Sardinia, Italy, vol. 256, pp. 61–70, 2006.

[11] M. Kazhdan and H. Hoppe, "Screened poisson surface reconstruction," *ACM Trans. Graph.*, vol. 32, no. 3, pp. 1–13, 2013.

[12] Khatamian and H. R. Arabnia, "Survey on 3D surface reconstruction," *J. Inf. Process. Syst.*, vol. 12, no. 3, pp. 338–357, 2016.

[13] R. Tang, S. Halim, and M. Zulkepli, "Surface Reconstruction Algorithms: Review and Comparison," no. 1984, p. 22, 2013.

[14] V. Chavan, Jimit Shah, M. Vora, M. Vora, and S. Verma, "A Review on Feature Extraction Methods for Image Analysis," *SSRN Electron. J.*, pp. 7–10, 2020.

[15] Banerjee and D. Mistry, "Comparison of Feature Detection and Matching Approaches: SIFT and SURF," *GRD Journals-Global Res. Dev. J. Eng.*, vol. 2, pp. 7–13, 2017.

[16] R. Sharma and P. Abrol, "Impact of Distortions on the Performance of Feature Extraction and Matching Techniques," *Lect. Notes Electr. Eng.*, vol. 701, pp. 357–367, 2021.

[17] B. B. Swapnali and K. S. Vijay, "Feature Extraction Using Surf Algorithm for Object Recognition," *Int. J. Tech. Res. Appl.*, vol. 2, no. 4, pp. 197–199, 2014.

[18] Maiti and D. Chakravarty, "Performance analysis of different surface reconstruction algorithms for 3D reconstruction of outdoor objects from their digital images," *Springerplus*, vol. 5, no. 1, 2016.

[19] J. E. Nunes Masson and M. R. Petry, "Comparison of Algorithms for 3D Reconstruction," *19th IEEE Int. Conf. Auton. Robot Syst. Compet. ICARSC 2019*, pp. 1–6, 2019.

[20] Z. Huang, Y. Wen, Z. Wang, J. Ren, and K. Jia, "Surface Reconstruction from Point Clouds: A Survey and a Benchmark," pp. 1–27, 2022.