UNIVERSITY OF WATERLOO

Faculty of Mechanical and Mechatronics Engineering

Leveraging Point Clouds for Architecture, Engineering, and Construction

Entuitive

200 University Ave

Kostubh Agarwal

20872383

2A Mechanical Engineering

September 19, 2022

Letter of Submittal

To MME Department

From Kostubh Agarwal

Date: Sept 19, 2022

Re: Work Report: Leveraging Point Clouds for AEC

I have prepared the enclosed report, "Leveraging Point Clouds for Architecture, Engineering, and Construction" for my 2A work report and for Entuitive.

Entuitive is a company offering professional services in the field of structural engineering, building science, transportation, and various sustainability applications.

As a member of the innovation department led by Mr. Blaine Jansen, my role is to help promote the culture of innovation by implementing new technologies and processes to improve the operation of the company.

This report was written entirely by me and has not received any previous academic credit at this or any other institution.

Sincerely,

Kostubh Agarwal

[20872383]

Table of Contents

List of Figures

Executive Summary

The intent of this report is to examine the technology, research, and workflows which can help the AEC industry leverage point clouds. Enabled by modern reality capture technologies such as LiDAR and Photogrammetry, point clouds are files whereby data is stored in a 3D space defined by geospatial coordinates. With point clouds, and reality capture technology in general, being so nascent, the AEC industry is struggling to make sense of these files. Currently there is limited infrastructure to distribute, manipulate, and decipher point clouds. More specifically, raw point clouds are not in format conducive to BIM work; They are often clunky, noisy, and ridden with outliers. While academia has done a lot of research in this space, technical complexity has yet to be abstracted away for the average user. Specifically, there are technical obstacles in the way of mass adoption within the AEC industry. This report attempts to offer a technical procedure to overcome these obstacles. Moreover, it will include references to well-regarded journals related to the post-processing pipelines, suggested distribution platforms, file formats, and more. Overall, this report serves as introductory material for enable the average member of AEC to begin leveraging point clouds.

1 Introduction

A point cloud is a set of data points plotted within a digital 3D space [1]. Point clouds are commonly used to represent physical objects, buildings, and or geography. LiDAR and photogrammetry equipment are the tools used to capture of such data. These capabilities are especially relevant for AEC[1]. Today, engineers, architects, and construction managers alike rely on manual measurement, video, and photographs to capture real-world conditions. In many cases, this information then must be interpreted and translated by humans to a higher-fidelity format: such as BIM models, or 2D drawings. Humans can be thought of as a translation layer, interfacing between the real-world and the digital world. Unfortunately, humans are prone to error, often missing or misinterpreting information. As such, the status quo for capturing and interpreting as-built conditions is costly. Many existing buildings have inaccurate, old, missing drawings; Taking manual measurements, photographs, and videos and then converting them to 2D drawings of BIM models is time-consuming; And most importantly, inaccurate capture of as-built conditions lead to an ever-growing set of problems downstream. Fortunately, the advent of LiDAR scanners and photogrammetry equipment enable the replacement of the human translation layer. Rather than relying on humans to piece together measurements, photos, videos, and old drawings point clouds offer an opportunity to automate the reality capture process. Unfortunately, point clouds are difficult to distribute, manipulate, and decipher - especially in the context of AEC. At best, the infrastructure to do so is limited. While there has been lots of academic research and development in the space, procedures, products, and services are not

[1] Acronym for Architecture, Engineering, and Construction

available or well-defined for non-technical consumers: File sizes are too large, riddled with uncertainties such as outliers, noise, and missing data, and there is simply lack of context and infrastructure to manipulate or translate this information. As such, engineers, construction managers and architects are hesitant to adopt point clouds and corresponding technology. LiDAR and reality capture technologies enable us to cut costs and deliver unparalleled accuracy and detail of as-built data for engineers and designers. Unfortunately, in their raw form, points clouds are hard to distribute, work-with, and or decipher. As such, the objective of this report is to provide a technical proposal to assist engineers, architects, and construction managers to easily leverage point clouds for use in AEC.

2 Background Information

2.1 Point Cloud

As shown in figure 1, a point cloud is a collection of data points plotted in 3D space [1]. Each data point is infinitesimally small and consists of a set of coordinates to define its location within a 3D space [1].

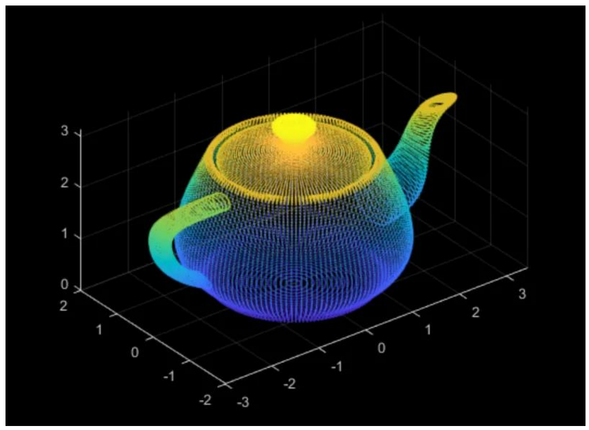

Figure 1: a point cloud of a teakettle [2]

2.2 LiDAR

LiDAR stands for light detection and ranging [3]. LiDAR scanners emit lasers to detect surfaces and their distance from the scanner [3]. This data is then encoded as a point cloud.

2.3 Mesh

As shown in figure 2, a mesh is a collection of vertices and polygons that define the shape of an object in 3D via a watertight surface [4]. A mesh can be generated from a point cloud.

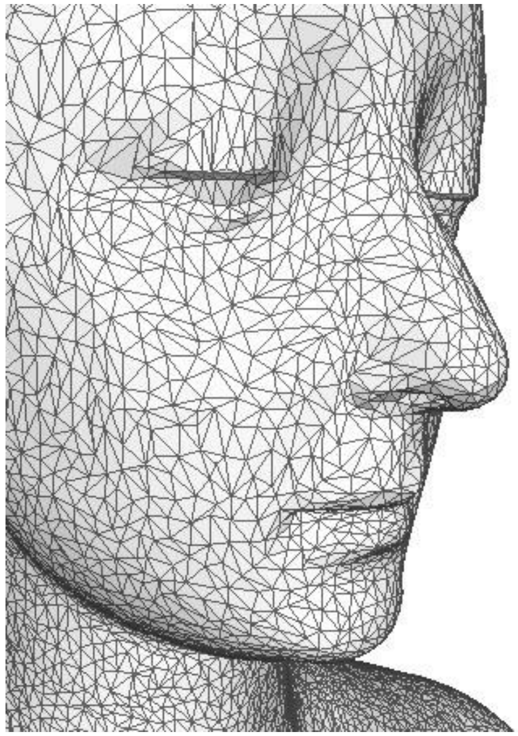

Figure 2: a mesh of a human-face [5]

2.4 BIM & Revit

The acronym BIM stands for building information modeling. BIM is technology for generating detailed digital models of buildings with information and data tied directly to geo-spatially within the model. Revit is a widely used BIM software.

2.5 Post-Processing

Point Clouds are often too large to distribute, too noisy to interpret, filled with outliers, or simply incompatible for BIM in its native form. As such, they require post-processing.

2.5.1 Sampling

LiDAR scanners capture large datasets. Correspondingly, point clouds are oftentimes cumbersome to process, distribute, and view. To manipulate their size and fidelity, point clouds can be down sampled or unsampled.

2.5.1.1 Decimation (Down sampling)

Decimation algorithms refer to algorithms which remove points on a numerical basis - independent of a points location within space. This may refer to the removal of 3 points for every 4 points - which would reduce the file size by approximately 75% as ¾ data points are discarded.

2.5.1.2 Voxel Sampling (Down sampling)

Voxel sampling is also known as grid sampling. Voxel sampling divides the 3D space into a cubic grid [6]. The user has control over the parameters and dimensions for grid itself. For example, each cell can have a volume of 8mm. Once a grid has been established, one point is selected to represent the entire cell. This can be a point at the center of the grid or a point that represents the barycenter of all the other points in the cell [6].

2.5.2 Removing Outliers and Denoising

In uncontrolled environments, LiDAR scanners are prone to capturing datasets with inaccuracies and outliers. As such, post-processing is often required to minimize these errors.

2.5.2.1 Edge aware resample (EAR)

Edge aware resampling is an algorithm which generates a "clean, uniform, and feature-preserving set of oriented points" [7] from a noisy, outlier-ridden, under sampled data set. Edge aware resampling well approximates the underlying surface generate a new dataset to better represent the original dataset [7]. This is done by resampling away from edges (identified by large differences in point orientations) and up sampling the gaps based on more reliable orientations [7]. Repeating this process multiple times results in a sharp, data-rich point cloud which is far more suitable for surface reconstruction procedures [7].

2.5.2.2 Weighted Locally Optimal Projections (WLOP)

Weighted locally optimal projections generate a new dataset to robustly approximate the original point cloud data and be well distributed [8]. To do so, L1 Spatial Medians are taken in well-defined local areas [8]. L1 Spatial Medians are insensitive to outliers [8]. Additionally, a repellant factor to ensure the projections are well distributed. Overall, WLOP can be described as a repeated local averaging algorithm [8].

2.5.2.3 Plane Fitting (MLS)

Moving Least Squares method relies on taking the surface of best fit, projecting all the points onto that surface, and repeating [9]. This is likely easier to visualize in a 2D space, Taking a line of best fit on a graph. This is a iterative process which if repeated enough times will render a flat surface. Overall, this is known as a smoothing procedure.

2.5.2.4 L0 minimization (L0)

L0 minimization serve to preserve and recover sharp features while smoothing everything else [10]. L0 minimization consists of three distinct stages. First using Principal Component Analysis, the point cloud is distinguished by smooth and sharp features [11]. If a point is identified to exist within a smooth region, the point itself and its surrounding points should form a plane perpendicular to the point norm [7]. Then points are unsampled along edges.

2.5.3 Meshing and Surface Reconstruction

Meshing refers to the generation of a watertight, intersection free, data-rich, and smooth surface from a set of points [12].

2.5.3.1 Poisson Surface Reconstruction

As shown in figure 3, Poisson Surface Reconstruction consists of a implicit function which generates an 'implied' point set from the original point set, with oriented normals [13]. This is defined by a function which defines the inside of a surface as a value >0 and the surface itself as 0 and the value outside the surface as <0.

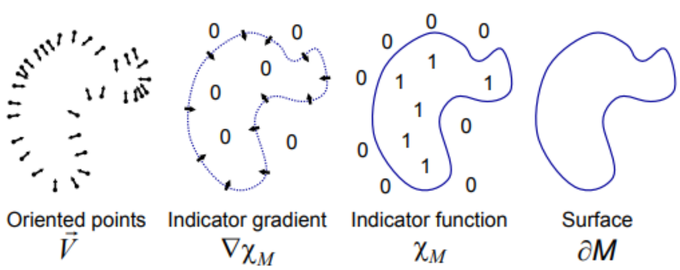

Figure 3: An intuitive illustration of Poisson reconstruction in 2D [13]

2.5.3.2 Ball Pivoting Algorithm

As shown in figure 4, the ball pivoting algorithm is an intuitive algorithm to visualize. It models a ball which rolls over the points in a point cloud, filling in the spaces where the ball does not fit. The radius of the ball is dynamic.

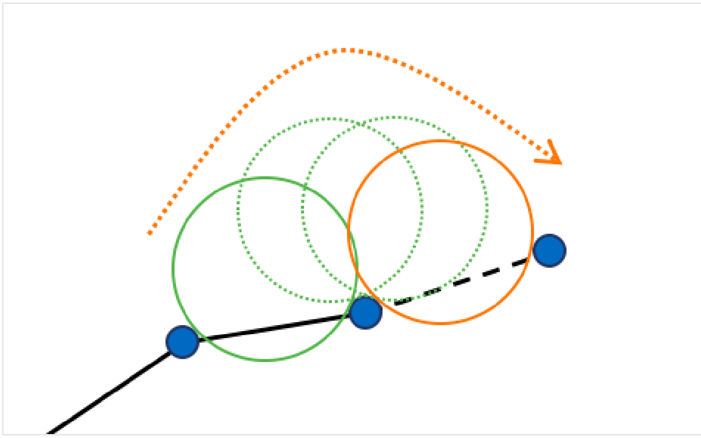

Figure 4: An intuitive illustration of ball-pivoting algorithm [14]

2.5.4 Optimizing for BIM

Meshes are difficult to extract information from and manipulate. As such, it needs to be compressed and reconfigured to become compatible for BIM.

2.6 Distribution

Point Clouds provide accurate as-built conditions. As such, collaborative teams, would collectively benefit from using a Point Cloud as a single source of truth.

2.6.1 File Type

2.6.1.1 3D File Formats

3D digital files are built to store polygons and their vertices. As such, they are also capable of storing points - point clouds.

eg. ply, obj, stl

2.6.1.2 LiDAR Format

LiDAR scanner manufactures have developed their own private file format to record LiDAR data to. LiDAR scanners collect geospatial data along with RGB data, and thermal data in unique cases.

eg. faro, las, laz

2.6.1.3 Point Cloud Formats

ASCII is a digital file format where points are simply stored in a text file as coordinates.

eg. ASCII, pcl, pcd

2.6.2 Visualization, Storage, and Distribution

2.6.2.1 Speckle.xyz

speckle.xyz is a platform to host, visualize, and distribute 3D information on the cloud. It is specifically designed for the AEC industry. It has added benefit of offering version control, interoperability, and automation.

3 Analysis & Results

To optimize a point cloud for distribution, visualization and information extraction, especially in the context of AEC, technical post-processing is required and distribution/storage/visualization platforms have to be defined. Post processing ensures enough fidelity to automate and replace the human translation layer between the physical and digital world. LiDAR scanners and photogrammetry equipment of all types are susceptible to noise, outliers, and missing information. Ideally, a raw point cloud should be converted to cloud hosted, BIM optimized, interoperable file type.

3.1 Denoising and Outlier Removal

With the end goal in mind, creating a tight mesh which can be converted into a BIM model, de-noising and outlier removal is critical. Four such filtration algorithms are said to perform well: L0 Minimization, Plane Fitting with Moving Least Squares, Weighted Locally Optimal Projections, and Edge Aware Resampling. A paper Xian-Feng Han et al. includes detailed tests comparing the filtering algorithms listed above [15]. The four filtration algorithms are tested against a benchmark: a ground truth point cloud with artificial gaussian noise [15]. As shown in figure 5, the quality of the output is defined by two error metrics: standard deviation and Cohen's mean [15]. The standard deviation measures averaged angle over all angles between the ground truth point normal and the resulting point normals [15]. Cohen's mean is a measurement of the average distance between the output points and their corresponding ground truths [15].

Model	Error	VG	NBF	MLS	RMLS	WLOP	EAR	L0
Chair	δ	32.125	20.336	8.962	8.932	8.904	8.785	8.752
	D_{mean}	0.427	0.307	0.133	0.132	0.134	0.132	0.130
i-H bunny	δ	10.223	6.021	5.862	6.105	5.710	5.553	5.559
	D_{mean}	0.185	0.092	0.089	0.099	0.089	0.085	0.086
Sofa	δ	12.850	10.126	10.388	7.211	5.334	5.186	5.032
	D_{mean}	0.174	0.169	0.167	0.150	0.087	0.087	0.080
Julius	δ	13.139	10.647	3.409	3.372	2.166	3.378	3.364
	D_{mean}	0.181	0.177	0.058	0.058	0.038	0.058	0.049
Coffee-mug	δ	12.962	12.729	7.655	7.375	6.447	6.498	6.487
	D_{mean}	0.210	0.201	0.120	0.116	0.100	0.101	0.101
Bowl	δ	25.494	21.676	6.149	6.003	7.667	7.311	7.386
	D_{mean}	0.501	0.312	0.086	0.074	0.107	0.102	0.104
Iron	δ	11.504	7.513	5.844	5.915	6.182	6.180	5.837
	D_{mean}	0.193	0.116	0.091	0.093	0.097	0.096	0.090
Armadillo	δ	9.071	3.616	3.599	4.406	5.124	5.139	5.005
	D_{mean}	0.117	0.052	0.052	0.061	0.097	0.075	0.066
Dragon	δ	26.375	23.036	20.816	21.090	20.816	17.646	16.494
	D_{mean}	0.421	0.359	0.299	0.303	0.310	0.262	0.202
Table	δ	32.481	29.313	17.010	17.449	28.540	21.918	15.998
	D_{mean}	0.553	0.502	0.277	0.285	0.463	0.358	0.254

figure 5 - point cloud filtering algorithm benchmark testing (numerical) [15]

Visually WLOP, EAR, and MLS seem to perform well, as seen in figure 6. Conversely, EAR, WLOP, and L0 Minimization preserve shape [15].

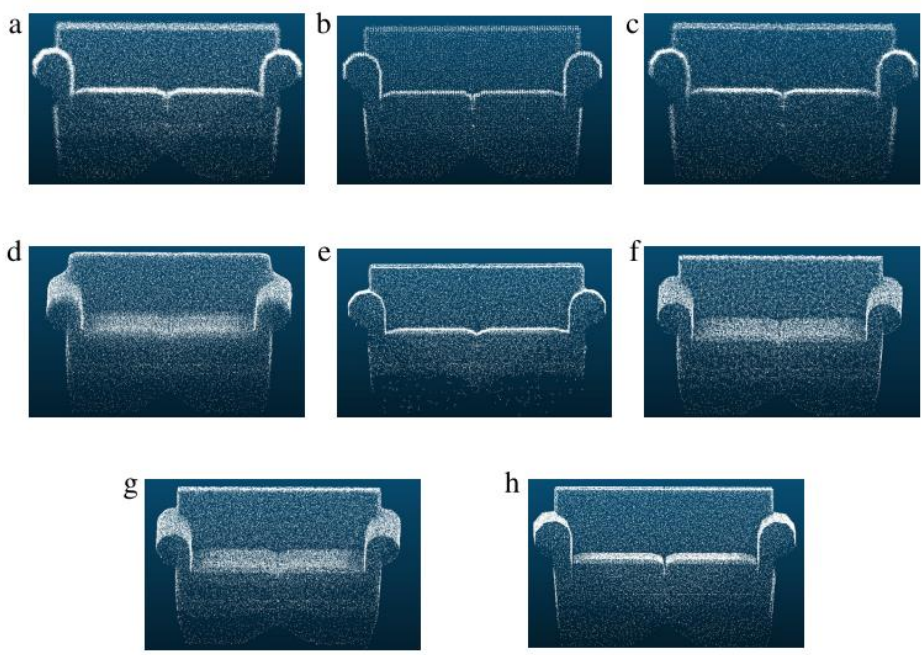

Fig. 8. (a) Noisy sofa model; (b) filtering result with VG; (c) NBF; (d) MLS; (e) RMLS; (f) WLOP; (g) EAR; (h) L0.

figure 6- point cloud filtering algorithm benchmark testing (visuals) [15]

Overall L0 minimization and EAR was able to retain shape, while producing the best visual results.

3.2 Sampling

Sampling is often important, especially for the sake of reducing the computational and storage loads downstream in the post-processing workflow. Down sampling a point cloud before filtration or before surface reconstruction makes it more difficult retain shape and produce quality results. As such, it is a matter of finding a 'sweet spot'. Preferably a point cloud is not down sampled before filtration, as this leads to the greatest error. In terms of choosing a down sampling algorithm, voxel sampling is the preferred method, as it down sample's point clouds in a predictable and uniform manner [6]. Additionally, it is customizable, with the parameters for defining the grid and the orientation of corresponding points being configurable. Whereas rank decimation is fairly unpredictable and results in a non-uniform removal of points. Overall, the degree of down sampling and when it is applied, should be context dependent, and is a matter of trading-off between performance and quality.

3.3 Surface Reconstruction

Surface reconstruction is arguably one of the most critical aspects of post-processing, but also the most complex. While sampling, or the lack thereof, and filtration are all precursors to the quality of mesh created, the choice of surface reconstruction algorithm is even more important. Whilst there are many credible papers on surface reconstruction from the likes of Boissonnat, Hoppe et al, Tight Cocone, Ball-pivoting, Poisson Surface Reconstruction, very few are robust [13]. For instance, Boissonat, Tight

Cocone, and the reconstruction algorithm by Hoppe et al. are very susceptible to noise and sparsity [13]. The Ball Pivoting Algorithm, especially with advent of a dynamic ball radius[2] is more [14]. However, the best performing algorithm in the context, of robustness and quality of output is the Poisson Surface Reconstruction. This algorithm insensitive to noise and outliers and octree depth (level of detail) is the greatest determinant for quality of output [13].

3.4 BIM Optimization

While a high-fidelity mesh is extremely performant in the context of 3D Graphics, it fails in the BIM setting. For example, if an engineer wants the dimensions of a wall, rather than getting the distance from once corner of the wall to the other, his measurement will be hindered by the tens of hundreds of vertices that exist in between - even if the wall is flat. Furthermore, meshes are computationally intensive and cumbersome to manipulate. Because of this inherent constraint with a default mesh, the status quo for converting a point cloud to BIM model, simple requires engineers to create a BIM model from scratch with the point cloud acting as points of reference: essentially enabling the engineer to 'color within the lines'. However, the intention of this report is to replace and automate engineers. As such algorithms must be applied to the mesh to automatically render it into a BIM friendly format. As such, the mesh needs to be defeatured. Or in simpler terms, the vertices of the mesh need to be removed on perceived single plane surfaces. This consists of a light-weight mesh, without any vertices intercepting a single flat plane.. In a manual

[2] The radius is inversely related to the susceptibility to noise [14].

context, a mesh can be thrown into blender where the edges can manually be locked, and the rest of the mesh can be defeatured. Overall, this is a large barrier to the automating the conversion of point clouds to BIM models.

3.5 Distribution / Storage / Visualization

Due to the general scale of AEC projects, collaboration is critical: between companies, progression, departments, and people. As such working from a single point of truth has extreme benefit. As projects become more and more abstract and specific, having a common starting point will minimize the ever-growing set of downstream errors. As such a cloud-based service such as speckle.xyz would serve best. It enables widespread collaboration, visualization on many mediums, and stores information in a cloud-based format. Additionally, the ability to convert between file-types, software's, and to maintain control is critical in an environment like this. Overall, the interoperability, friendliness to 3D file formats, and collaborative features make speckle.xyz the best platform for the storage, distribution and visualization purposes.

4 Conclusion and Recommendations

Point clouds have the capacity to revolutionize the AEC industry. However, it will be decades before Point Clouds become the native 3D file format. The current barrier to adoption is technical. As such, to ease the transition from BIM models to point clouds, the technical challenges must be abstracted away from the user. In the time being, this may mean delivering point clouds in a format that the AEC industry is already used to and comfortable with: BIM models. Point clouds are a wasted technology if limited to academia. This report proposed a workflow to automate the and best leverage point clouds in an AEC context. However, this report completely disregards the compute demands of each of these platforms, processing pipelines, and file types. Overall, a lot more research and development, in the consumer market, has to be done to increase for adoption and realize the of benefits of this technology.

5 References

[1] “Point Clouds for Beginners: Your Questions Answered.” *GeoSLAM*, 30 Aug. 2022, https://geoslam.com/point-clouds/.

[2] Bennett, Alison. “Point Cloud and 3D Image.” *Revopoint 3D Technologies Inc.*, 5 July 2021, https://www.revopoint3d.com/point-cloud-and-3d-image/.

[3] NOAA. Historical Maps and Charts audio podcast. National Ocean Service website, https://oceanservice.noaa.gov/podcast/july17/nop08-historical-maps-charts.html, accessed on 8/13/17.

[4] Maiti, Abhik, and Debashish Chakravarty. “Performance Analysis of Different Surface Reconstruction Algorithms for 3D Reconstruction of Outdoor Objects from Their Digital Images.” *SpringerPlus*, vol. 5, no. 1, 2016, https://doi.org/10.1186/s40064-016-2425-9.

[5] “3D Object Representations.” *Polygon Meshes*, https://glasnost.itcarlow.ie/~powerk/GeneralGraphicsNotes/meshes/polygon_meshes.html.

[6] Florent Poux, Ph.D. “How to Automate Lidar Point Cloud Processing with Python.” *Medium*, Towards Data Science, 11 Apr. 2021, https://towardsdatascience.com/how-to-automate-lidar-point-cloud-processing-with-python-a027454a536c.

[7] *VCC*, https://vcc.tech/research/2013/EAR.

[8] Lipman, Yaron, et al. “Parameterization-Free Projection for Geometry Reconstruction.” *ACM Transactions on Graphics*, vol. 26, no. 3, 2007, p. 22., https://doi.org/10.1145/1276377.1276405.

[9] Alexa, M., et al. “Computing and Rendering Point Set Surfaces.” *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, no. 1, 2003, pp. 3–15., https://doi.org/10.1109/tvcg.2003.1175093.

[10] Sun, Yujing, et al. “Denoising Point Sets via L0 Minimization.” *Computer Aided Geometric Design*, vol. 35-36, 2015, pp. 2–15., https://doi.org/10.1016/j.cagd.2015.03.011.

[11] *Principal Components Analysis*, http://ordination.okstate.edu/PCA.htm.

[12] MicrosoftResearch, director. *YouTube*, YouTube, 16 Aug. 2016, https://www.youtube.com/watch?v=bMhR-UYlKqU. Accessed 16 Sept. 2022.

[13] Hoppe, Hugues. “Poisson Surface Reconstruction and Its Applications.” *Proceedings of the 2008 ACM Symposium on Solid and Physical Modeling - SPM '08*, 2008, https://doi.org/10.1145/1364901.1364904.

[14] “Point Cloud to Mesh, Ball-Pivoting Algorithm Brett Rapponotti, Michael Snowden, and Allen Zeng.” *Point Cloud to Mesh: BPA*, https://cs184team.github.io/cs184-final/writeup.html.

[15] Han, Xian-Feng, et al. “A Review of Algorithms for Filtering the 3D Point Cloud.” *Signal Processing: Image Communication*, vol. 57, 2017, pp. 103–112., https://doi.org/10.1016/j.image.2017.05.009.