# 3D Model Simplification through Elementary Geometric Structures using Hough Transform

Nicolas Ebone Cendron
Institute of Informatics, UFRGS
necendron@inf.ufrgs.br

Dennis Giovani Balreira
Institute of Informatics, UFRGS
dgbalreira@inf.ufrgs.br

***Abstract*—Current methods for three-dimensional (3D) model simplification often involve intricate algorithms that may compromise visual fidelity or incur high computational costs. Traditional approaches using decimation algorithms may still fall short in terms of achieving storage efficiency comparable to basic 3D geometric primitives. This paper proposes an approach centered on the utilization of elementary geometric structures, such as spheres and cylinders, for efficient 3D model simplification. Our approach capitalizes on the inherent simplicity of such shapes, enabling representation with fewer parameters and minimal storage requirements. The proposed method replaces intricate geometric details with these fundamental shapes, identifying regions suited to substitution using a Hough Transform-based method. Preliminary findings replace a region of the mesh with a single sphere. We present these primary results visually with the use of spheres for three 3D meshes along with their corresponding percentage gain regarding fundamental characteristics such as vertices, edges, faces, and size. For future work, we intend on expanding our technique to map other parts of the model and exploring further elementary geometries with Hough Transform.**

## I. INTRODUCTION

In the area of computer graphics and computational geometry, the efficient representation and manipulation of three-dimensional (3D) models play a crucial role in various applications, ranging from virtual reality to computer-aided design. As the complexity of 3D models continues to increase, the need for techniques that simplify these models becomes necessary. These techniques of 3D model simplification aim to reduce the computational overheads associated with rendering, transmission, and storage, while upholding a level of detail that remains perceptually convincing. Traditional methods, often grounded in decimation algorithms [1]–[3], usually focus on specific regions or levels of detail, which undoubtedly helps in decreasing the meshes' size.

Even though decimation algorithms have been widely employed for 3D model simplification [4], they often fall short in achieving storage efficiency comparable to that of single basic 3D geometric primitives. While decimation methods aim to reduce complexity by removing unnecessary details, the resulting models can still exhibit significant storage requirements due to the persistence of intricate mesh structures, where the complex mesh topology remains a significant contributor to storage overhead.

In contrast, the utilization of elementary geometric primitives such as spheres and cylinders offers the advantage of inherently embodying simplicity, allowing them to be represented using fewer parameters and requiring minimal storage space [5]. By embracing the elegance of single basic geometric primitives, this approach not only streamlines storage but also retains the core characteristics of the original models, making it a compelling alternative in the pursuit of efficient 3D model representation. These structures can be seamlessly combined, stacked, and arranged to mimic the intricate contours and volumes present in complex 3D models. Moreover, the uniformity of these basic structures ensures a streamlined approach to simplification, leading to optimized rendering, reduced storage requirements, and efficient transmission, all while maintaining the essence of the original object.

Our approach seeks to replace intricate geometric details with these fundamental shapes by identifying regions within the 3D model amenable to such substitution using a Hough Transform-based approach. This strategy allows to detection of potential detailed mesh areas and substitutes them for a few equation parameters for performance enhancing purposes. By substituting detailed mesh areas with equation parameters that define fundamental shapes, the method optimizes storage space and computational demands, contributing to enhanced performance and responsiveness.

The remaining of this work contains four sections. Section II presents the most relevant work associated with 3D mesh simplification. Section III presents our approach to converting the models to elementary geometric structures. Next, in Section IV we discuss the partial results we obtained for some three-dimensional objects, along with a short discussion on our next planned steps. Finally, Section V presents our final remarks, limitations, and future work.

## II. RELATED WORK

Mesh simplification technology converts a given 3D model, which usually has high resolution and precision, into a less detailed and approximated mesh model [6]. In the present landscape, significant advancements have been made by numerous researchers in this domain in the last decades. The objective of shape approximation algorithms, particularly, is to derive uncomplicated geometric representations from intricate surface meshes. Numerous algorithms in this domain rely on mesh decimation methods that yield coarse triangulations while optimizing a specific metric designed to capture the proximity to the original shape [5]. Furthermore, due to the democratization of consumer depth cameras, captured 3D data is also continuously increasing, becoming a serious challenge at the frontier between computer graphics and computer vision [7]. One of the most classic works regarding mesh decimation can be seen in the book of Luebke [1].

In this field, some authors classify the algorithms into two categories [8], [9]: geometry-driven simplification algorithms and appearance attribute-driven simplification algorithms. Among the prevalent geometry-driven simplification techniques we may cite vertex clustering-based algorithms [10], vertex extraction-based algorithms [11], and edge folding-based algorithms [12]. Another important classic technique proposed by Cohen and colleagues is the one called Simplification Envelopes [13], used for simplifying complex 3D models while preserving their essential features and overall appearance.

The Hough Transform [14], [15] is a powerful technique widely employed in image processing, pattern analysis, and computer vision. Initially developed for detecting straight lines in images, it has since evolved into a versatile tool for identifying complex shapes, curves, and patterns. This transformative method functions by converting the data space into a parameter space, facilitating the identification of patterns through peak detection in the parameter domain. Some works have worked with Hough Transform in Computer Graphics 3D meshes [16], [17].

The work proposed by Abuzaina et al. [16] introduces an algorithm founded on the Hough Transform methodology, aimed at efficiently and accurately detecting spherical structures within 3D point clouds. Khoshelham's work [17], works with automated object detection and 3D modeling within laser range data, detecting 3D objects characterized by arbitrary shapes within point cloud data. Further approaches have also been investigating mesh reconstruction from point clouds, which commonly uses Hough Transform [18]–[21].

In the context of our study, we leverage the Hough Transform as a foundation for detecting and characterizing potential substitution regions within 3D models, enabling an innovative approach to simplification. By harnessing the inherent strengths of the Hough Transform, our methodology seeks to enhance the efficiency and accuracy of our proposed technique for model simplification. Similar state-of-the-art methods, such as [5] involve producing several spheres inside the mesh and connecting them according to the importance parameter $\sigma$, others such as [22] use Lloyd clustering as a mean of minimizing the sphere sizes. We believe our technique can greatly enhance the performance of the meshes by automatically substituting the entire mesh regions for these structures, which can be represented by simple equation parameters without further information representing them.

## III. METHODOLOGY

In this section, we present the methodology we used to replace 3D meshes with elementary geometric structures. Given a three-dimensional mesh, our method consists of using the Hough Transform approach to find the best mesh's area for further substituting it to the corresponding geometry. Figure 1 presents a schema illustrating our method for a sphere.

A known way of detecting geometry in 2D is the Hough

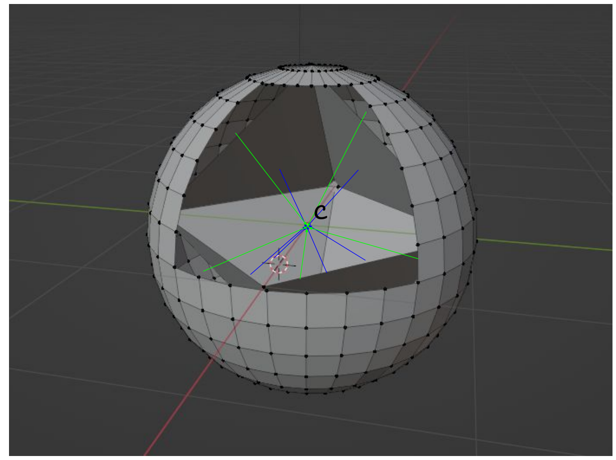


Fig. 1. Illustration of our technique, representing the normals of an octahedron figure (in blue) and of a sphere (in green) meeting at a center C.

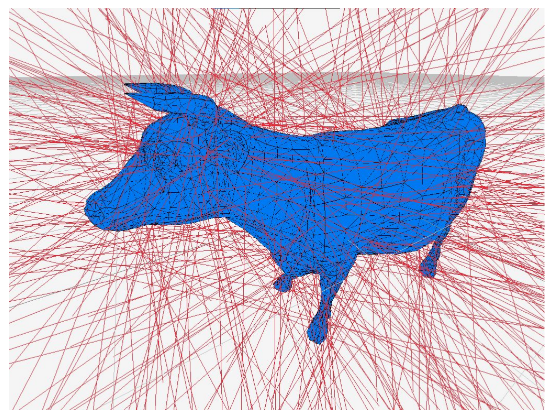

Fig. 2. Mesh example of the Cow (blue) along with some of its computed normals (red).

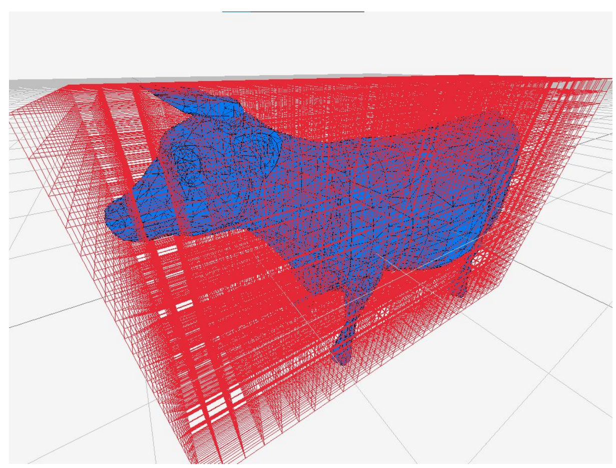

Fig. 3. Mesh example of the Cow (blue) and its corresponding accumulating cubes (red).

Transform, which uses mathematical equations that define shapes to detect said shapes. Circles, for instance, can be found using the Circle Hough Transform. We can use these tools in three dimensions to detect and simplify data in a 3D object.

Say we were to try and find and simplify spheres in a mesh $M$, composed of triangular polygons. We start by extracting all faces $F$ and all vertices $V$ corresponding to each face in $M$. Then we set some vertex as our origin $O$ and map out vectors $v$ that go from $O$ to the other two $V$ that belong to $F$.

By cross-multiplying both $v$, we will get the normal $n$ relative to $O$. By repeating this process replacing $O$ for each $V$ in $F$, we can find the normal $N$ of said $F$ by taking the average of the coordinates of each found $n$. Figure 2 shows the result of this normal extraction process in the Cow mesh.

We then create a 3D grid of small accumulator cubes of edge size $s$ (picking up more and more precise sphere centers as $s$ gets smaller). Each cube starts at 0 and adds 1 to its accumulator for each normal line that goes through it and keeps a reference of what face's normal hit it. This use of accumulator cubes can be seen in Figure 3 with the use of a Cow model and the 3D arrays surrounding it.

After that, for a small enough $s$, if there are sphere-like shapes in our mesh, we will find that the cubes with the highest accumulated value will be the center of said spheres. To find the radius of a sphere with the center in a cube $C$, we create a sphere $S$ of radius starting at $s$ and grow it until it covers all of $M$, keeping note of which faces are inside of it.

The point when $S$ covers the highest amount of faces $F_A$ with normal that hit $C$ and the lowest amount of faces $F_B$ with normal that did not hit $C$ will be the sphere with the approximate radius. This can also be thought of as the radius $r$, where $|F_A - F_B|$ is the smallest. The $F_A$ faces could then be removed and replaced by the equation of a sphere of center in $C$ and radius $r$, leaving us with the simplified mesh.

The same breakdown of the equations used in the usual Hough Transform can be applied to other shapes. We intend to continue investigating this issue in the following months, producing a subset of geometric elementary forms which can produce viable and memory-less alternatives to simplify the input mesh.

## IV. PRELIMINARY RESULTS AND DISCUSSION

This section presents our preliminary results, followed by a discussion. Table I presents the initial simplification results presented in this paper for three 3D objects using a sphere approximation centered at the accumulator with the highest normal counter. Figure 4 shows these results visually, with the sphere represented in green and the models in blue.

We can see that the number of vectors, edges, and faces decreases since they can be removed from the original file and substituted with the sphere equation. The Cow, Chueburashka, and Homer models presented 4.32%, 15.25%, and 6.08% decrease in size, respectively, with an average of 8.55%. Although marginal, these decreases are the result of a single sphere substitution, and would be significantly improved with multiple geometric shape replacements of the model. We highlight that Chueburashka had the most significant decrease among the figures since the sphere replaced the biggest percentage of the model compared to the others.

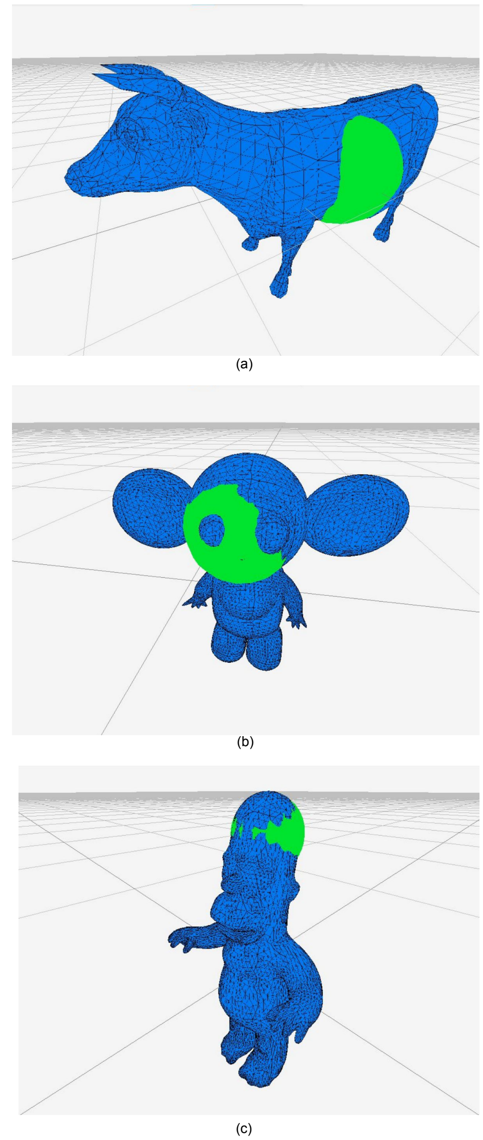


Fig. 4. Results of our technique presenting a sphere (green) for the three models (blue): (a) Cow, (b) Chueburashka, and (c) Homer.

The results present the potential of such simplifications, showing that a simplification with more spheres or other shapes like cones and cylinders could further reduce the sizes while better representing several mesh subsets.

## V. CONCLUSIONS

Our approach involves substituting intricate geometric complexities with fundamental shapes such as spheres and cylinders, achieved through a Hough Transform-based methodology, suiting approaches that rely on low-resolution meshes. We use elementary geometric structures to simplify 3D models by identifying model regions available for such substitution. Our results present, on average, a decrease of 8.5% in size. Despite

TABLE I
ORIGINAL AND NEW MESHES' SIZE INFORMATION BEFORE AND AFTER OUR APPROACH.

| Model | Orig. #V | Orig. #E | Orig. #F | Orig. size (OBJ) | New #V | New #E | New #F | New size (OBJ) |
|---|---|---|---|---|---|---|---|---|
| Cow | 2,903 | 8,706 | 5,804 | 185kB | 2,762 | 8,177 | 5,415 | 177kB |
| Chueburashka | 6,669 | 20,001 | 13,334 | 413kB | 5,746 | 16,857 | 11,114 | 350kB |
| Homer | 6,002 | 18,000 | 12,000 | 362kB | 5,899 | 17,587 | 11,690 | 340kB |
| **Average** | 5.191 | 15.569 | 10.379 | 320kB | 4.802 | 14.207 | 9.406 | 289kB |
| **Std. Dev.** | 2,009.62 | 6,027.15 | 4,018.10 | 119.66 | 1,768.64 | 5,234.87 | 3,468.57 | 97.12 |

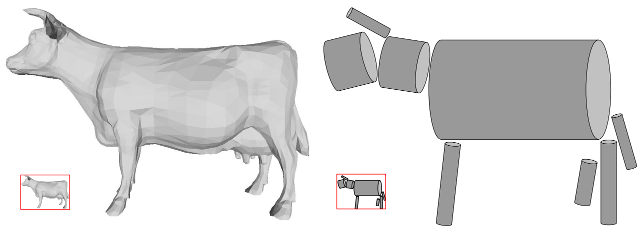

Fig. 5. An example of the potential of our approach is to be further explored for the Cow model, which can be roughly approximated by using eight cylinders. The same images can also be seen as low-sized in the red contoured images.

still being preliminary at this stage, our approach provides an attractive method that can be further expanded by capturing and adjusting the most important set of triangles to elementary geometric structures in a space-efficient way.

We intend to expand our technique to map elementary geometric structures to other parts of the model. For instance, by only using cylinders, we may produce a sketch-based similar Cow as the one presented in Figure 5. As we present in the short red images, a model in such low resolution could simplify the rendering of far objects in a scene without losing much semantic information and occupying much less space than decimate approaches. Furthermore, we also intend to keep exploring other limitations, such as dealing with textured models and improving our results by presenting a geometric-based error calculus on the input and output meshes, such as quadratic error metric [6], [23].

## ACKNOWLEDGEMENTS

This study was partially financed by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

## REFERENCES

[1] D. Luebke, *Level of detail for 3D graphics*. Morgan Kaufmann, 2003.
[2] M. Corsini, M.-C. Larabi, G. Lavoué, O. Petřík, L. Váša, and K. Wang, "Perceptual metrics for static and dynamic triangle meshes," in *Computer graphics forum*, vol. 32, no. 1. Wiley Online Library, 2013, pp. 101–125.
[3] R. L. Cook, J. Halstead, M. Planck, and D. Ryu, "Stochastic simplification of aggregate detail," *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3, pp. 79–es, 2007.
[4] J. Hasselgren, J. Munkberg, J. Lehtinen, M. Aittala, and S. Laine, "Appearance-driven automatic 3d model simplification." in *EGSR (DL)*, 2021, pp. 85–97.
[5] J.-M. Thiery, É. Guy, and T. Boubekeur, "Sphere-meshes: Shape approximation using spherical quadric error metrics," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 6, pp. 1–12, 2013.
[6] M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," in *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, 1997, pp. 209–216.
[7] A. Kaiser, J. A. Ybanez Zepeda, and T. Boubekeur, "A survey of simple geometric primitives detection methods for captured 3d data," in *Computer Graphics Forum*, vol. 38, no. 1. Wiley Online Library, 2019, pp. 167–196.
[8] Z. Liu, C. Zhang, H. Cai, W. Qv, and S. Zhang, "A model simplification algorithm for 3d reconstruction," *Remote Sensing*, vol. 14, no. 17, p. 4216, 2022.
[9] Y. Xian, Y. Fan, Y. Huang, G. Wang, C. Tu, X. Meng, and J. Peng, "Mesh simplification with appearance-driven optimizations," *IEEE Access*, vol. 8, pp. 165 769–165 778, 2020.
[10] J. Rossignac and P. Borrel, "Multi-resolution 3d approximations for rendering complex scenes," in *Modeling in computer graphics: methods and applications*. Springer, 1993, pp. 455–465.
[11] W. J. Schroeder, J. A. Zarge, and W. E. Lorensen, "Decimation of triangle meshes," in *Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, 1992, pp. 65–70.
[12] M. Garland and Y. Zhou, "Quadric-based simplification in any dimension," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 2, pp. 209–239, 2005.
[13] J. Cohen, A. Varshney, D. Manocha, G. Turk, H. Weber, P. Agarwal, F. Brooks, and W. Wright, "Simplification envelopes," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996, pp. 119–128.
[14] J. Illingworth and J. Kittler, "A survey of the hough transform," *Computer vision, graphics, and image processing*, vol. 44, no. 1, pp. 87–116, 1988.
[15] P. Mukhopadhyay and B. B. Chaudhuri, "A survey of hough transform," *Pattern Recognition*, vol. 48, no. 3, pp. 993–1010, 2015.
[16] A. Abuzaina, M. S. Nixon, and J. N. Carter, "Sphere detection in kinect point clouds via the 3d hough transform," in *Computer Analysis of Images and Patterns: 15th International Conference, CAIP 2013, York, UK, August 27-29, 2013, Proceedings, Part II 15*. Springer, 2013, pp. 290–297.
[17] K. Khoshelham, "Extending generalized hough transform to detect 3d objects in laser range data," in *ISPRS Workshop on Laser Scanning*, vol. 36, 2007, p. 206.
[18] A. K. Patil, P. Holi, S. K. Lee, and Y. H. Chai, "An adaptive approach for the reconstruction and modeling of as-built 3d pipelines from point clouds," *Automation in construction*, vol. 75, pp. 65–78, 2017.
[19] T. Rabbani and F. Van Den Heuvel, "Efficient hough transform for automatic detection of cylinders in point clouds," *Isprs Wg Iii/3, Iii/4*, vol. 3, pp. 60–65, 2005.
[20] Y.-H. Jin and W.-H. Lee, "Fast cylinder shape matching using random sample consensus in large scale point cloud," *Applied Sciences*, vol. 9, no. 5, p. 974, 2019.
[21] D. Borrmann, J. Elseberg, K. Lingemann, and A. Nüchter, "The 3d hough transform for plane detection in point clouds: A review and a new accumulator design," *3D Research*, vol. 2, no. 2, pp. 1–13, 2011.
[22] R. Wang, K. Zhou, J. Snyder, X. Liu, H. Bao, Q. Peng, and B. Guo, "Variational sphere set approximation for solid objects," *The Visual Computer*, vol. 22, pp. 612–621, 2006.
[23] M. Garland and P. S. Heckbert, "Simplifying surfaces with color and texture using quadric error metrics," in *Proceedings Visualization'98 (Cat. No. 98CB36276)*. IEEE, 1998, pp. 263–269.