

Research Article

3D Point Cloud Simplification Based on k -Nearest Neighbor and Clustering

Abdelaaziz Mahdaoui ¹ and El Hassan Sbai ²

¹Department of Physics, Faculty of Sciences, Moulay Ismail University of Meknes, Meknes, Morocco

²High School of Technology, Faculty of Sciences, Moulay Ismail University of Meknes, Meknes, Morocco

Correspondence should be addressed to Abdelaaziz Mahdaoui; a.mahdaoui@edu.umi.ac.ma

Received 5 March 2020; Revised 8 June 2020; Accepted 13 June 2020; Published 15 July 2020

Academic Editor: Patrick Seeling

Copyright © 2020 Abdelaaziz Mahdaoui and El Hassan Sbai. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

While the reconstruction of 3D objects is increasingly used today, the simplification of 3D point cloud, however, becomes a substantial phase in this process of reconstruction. This is due to the huge amounts of dense 3D point cloud produced by 3D scanning devices. In this paper, a new approach is proposed to simplify 3D point cloud based on k -nearest neighbor (k -NN) and clustering algorithm. Initially, 3D point cloud is divided into clusters using k -means algorithm. Then, an entropy estimation is performed for each cluster to remove the ones that have minimal entropy. In this paper, MATLAB is used to carry out the simulation, and the performance of our method is testified by test dataset. Numerous experiments demonstrate the effectiveness of the proposed simplification method of 3D point cloud.

1. Introduction

The simplification of a 3D point cloud, obtained from the digitization of a real object, is a primordial and important step in the field of 3D reconstruction. This step ensures the optimization of the number of points that constitute the 3D point cloud [1]. The scanning of a real object is facilitated by a device called 3D scanner [2]. This device may be broken down into three primary sorts: contact, active noncontact, and passive noncontact.

Simplification of a 3D set of points can be defined as follows: being given an original surface S presented by a point cloud X such that $|X| = N$, simplification of X consists of calculating a point cloud X' such that $|X'| = M$, knowing that $|\cdot|$ is a cardinality. After simplification, we obtain a simplified point cloud such that $|X'| \leq |X|$. It should be noted that X' samples a surface S' close to the original surface S that is sampled by X .

Several scientific articles have studied and presented simplification methods. Pauly et al. [3] proposed a method based on hierarchical decomposition of the sample of points, calculated by binary partition of space. The cutting planes

are defined by the centre and the main direction of each region. The partitioning criterion depends both on a maximum number of points and on variations in local geometry in a region. Due to the spatial nature of this approach, it is difficult to control the quality of the distribution of points on the sampled surface. Wu and Kobbelt [4] computed an optimal set of splats to cover a sampled surface. The first step of the method consists in locally approximating the surface at each point of the sample by a circular or elliptical plane surface element called a splat. In the second step, the redundant splats are eliminated during a filtering process of the surface expansion type. To guarantee the recovery of the entire sampled surface, the algorithm proceeds as follows. For each splat processed, the points it covers are projected onto its plane, and then only the splats associated with the points projected inside the convex envelope of the projected points are eliminated. During this process, the regularity of the distribution is not checked. A relaxation phase can be applied to determine an optimal position for the remaining splats. This method makes it possible to generate high quality splat covers for smooth surfaces, by filtering noise. However, this method is penalized by the cost of its initialization and

that of the relaxation phase for large point samples. Linsen [5] presented a technique that associates a scalar value with each point locally measuring the average variation of certain information, such as the proximity of neighbors or the direction of normal. The points with the weakest measurement are removed iteratively. The algorithm has the disadvantage of not giving any guarantee on the density of the resulting set of points. Dey et al. [6] used an approximation of the LFS (local feature size) of the sampled area. This approximation is calculated from the Delaunay triangulation of the sample of input points, which has the drawback of very large samples. Alexa et al. [7] estimated the local geometrical properties of the sampled surface using a Moving Least Squares (MLS) model of the underlying surface, which requires having oriented normal in a consistent manner. They calculate the contribution of a point to this surface by projecting it onto an MLS surface estimated from neighboring points. The distance between the position of the point and its projection on the surface provides a measure of error. The points for which this distance is the smallest are removed. This method does not guarantee the density of the resulting sample points. To compensate, Alexa et al. [7] proposed to enrich the sample in the undersampled regions by considering the projection of these on a plan. They calculated the plane Voronoi diagram of the projected points so as to insert new points equidistant from the first. These new points are then raised to the surface using the projection operator. The process is repeated until the Euclidean distance between the next point to be added and the nearest existing point becomes less than a certain threshold. While this method achieves quality results, the intensive use of the MLS projection operator makes it expensive for very large samples. Pauly et al. [3] have directly extended the mesh simplification technique of Garland and Heckbert [8] for point samples by considering the relations of nearest neighbors as connectivity relations. Pairs of nearest neighbors are thus contracted, replacing two points with a new point calculated as a weighted average of the first. The cost of each contraction operation is measured by adapting the error measure proposed by Garland and Heckbert, whose idea is to approximate the surface locally by a set of tangent planes and to estimate the geometric deviation of a point, with respect to the surface represented by the sum of the distances squared to these planes. This method has the advantage of controlling the distribution of the simplified sample, which also has the property of preserving the details. However, its initialization cost is high, and it requires the maintenance of an overall priority queue, which is a disadvantage for large samples of points. Xuan et al. [9] proposed a progressive point cloud simplification technique, founded on the theory of the information entropy and normal angle. The fundamental of this technique is to find the importance of points using the information entropy of the normal angle. Calculation of the normal angle is based on the normal vectors. The simplification operation is carried out by removing the less relevant points.

Leal et al. [10] proposed a simplification technique comprised of three stages. First, to cluster point cloud, the expectation maximization algorithm is used. Second, the

point cloud to be removed using curvature is selected. Third, linear programming is used to simplify point cloud. Ji et al. [11] proposed a simplification technique named detail feature points simplified algorithm. In this technique, a rule of k neighborhood and an octree structure are used to reduce point cloud.

The first key interest of this paper is point cloud simplification. The extraordinary simplification point cloud strategies reviewed in the literature may be classified into three categories: subsampling algorithms, resampling algorithms, and a mixture of them [12]. A first strategy for simplifying a sample of points is to break it down into small regions, each of which is represented by a single point in the simplified sample, while the resampling algorithms rely on estimating the properties of the sampled surface to compute new relevant points. In the literature, these principles have been applied according to three main simplification schemes: simplification by selection or calculation of points representing subsets of the initial sample [3], iterative simplification [6], and simplification by incremental sampling [13].

The second key interest of this paper is the clustering notion. Clustering is a statistical analysis method used to organize raw data into homogeneous groups. Within each cluster, the data are grouped according to a common characteristic. The scheduling tool is an algorithm that measures the proximity between each element based on defined criteria. Clustering is an integrated concept in several areas such as pattern recognition [14], machine learning [15], and 3D point cloud simplification [12, 16]. In the literature, there are many clustering techniques [17]. The work in this article is based on clustering to optimize the number of points constituting an original 3D point cloud in order to obtain another simplified 3D point cloud close to the original.

The third key interest of this paper is generally information theory and particularly the concept of Shannon's entropy [18]. This work is based on this concept to select the set of points grouped into cluster in order to simplify the original point cloud. Information theory is presented in different areas such as data processing [19, 20], data clustering [21], and 3D point cloud simplification [1, 9].

In this work, we are inspired by the work of Wang et al. [22] in order to provide a robust method of simplifying the point cloud. This technique is based on the notion of entropy [18] and clustering algorithm [17].

This paper is organized as follows. In Section 2, we evoke the density function estimator and entropy definition. Then, in Section 3, we present clustering algorithm used in our method. In Section 4, we demonstrate how to evaluate simplified meshes. Afterwards, in Section 5, we lay out our 3D point cloud simplification algorithm based on the Shannon's entropy [18]. Section 6 lays out the experimental results and the validation of the proposed technique. Finally, we wrap up with a conclusion.

2. Clustering Algorithm

The k -means clustering [23] is a type of unsupervised learning and analysis. The goal of this algorithm is to find

groups in data, with the number of groups represented by the variable K , in which each goal belongs to the group with the closest average. The k -means clustering will be thought of as the foremost important unsupervised learning approach, which is widely used in pattern recognition and machine intelligence. The details of k -means clustering algorithm are presented in [17].

3. Density Estimation and Entropy Definition

In this 3D point cloud simplification work, we use the concept of entropy to simplify point clouds. The calculation of the entropy requires the estimation of the density function. Multitudes density estimation approaches exist in literature, such as parametric and nonparametric methods. The first category makes it possible to estimate a parameterized model of a density function such as the maximum likelihood estimator method [24]. The nonparametric category includes the kernel density estimator, also known as the Parzen-Rosenblatt method [25, 26], the k -nearest neighbor estimator (k -NN), and a combination of them [27]. Each type has its advantages and disadvantages. For Parzen estimator, the bandwidth choice has strong impact on the quality of the estimated density [28]. In other words, the main motivation stems from the fact that k -NN estimator represents a solution to adapt the amount of smoothing to the local density of the data [21, 27]. The parametric approach has the main disadvantage of requiring prior knowledge of the probability law of the random phenomenon under study. The nonparametric approach estimates the probability density directly from the available information on the set of observations. We are interested here rather in the nonparametric category, specifically the k -NN estimator.

3.1. Density Estimation Using k -NN Approach. In this work, an unstructured approach, so called nonparametric estimation, was used to estimate density function. There are two kinds of nonparametric estimation methods: one is the Parzen density estimator [25] the other is the k -nearest neighbor (k -NN) density estimator [27]. In this paper we use k -NN technique to estimate density function. In the literature, the k -NN concept is used in several fields related to classification as in articles [29–31].

The level of the estimator is defined by k , which is an integer number of the nearest neighbors, generally proportional to the size of the sample N . Definition of the density estimate is done for any point x . The distances between objects of the sample and points x are as follows:

$$R_1(x) < \dots < R_{k-1}(x) < R_k(x) < \dots < R_N(x), \quad (1)$$

where R_i with $i = 1, \dots, N$ are distances sorted in ascending order.

The k -nearest neighbor estimator in d dimension can be defined as follows:

$$p_{knn}(x) = N^{-1} R_k(x)^{-d} \sum_{i=1}^N K[R_k(x)^{-1}(x - X_i)], \quad (2)$$

where $R_k(x)$ is the distance from x to the k th nearest point and $K(u)$ is the Gaussian kernel:

$$K(u) = (2\pi)^{-(d/2)} \exp\left(-\frac{1}{2}u^T u\right). \quad (3)$$

Then, we obtain

$$p_{knn}(x) = N^{-1} R_k(x)^{-d} (2\pi)^{-(d/2)} \sum_{i=1}^N \exp$$

$$\cdot \left[-\frac{1}{2} (R_k(x)^{-1} (x - X_i))^2 \right],$$

$$p_{knn}(x) = \frac{k}{N \cdot V_k(x)} = \frac{k}{N \cdot C_d \cdot R_k(x)}, \quad (5)$$

where $V_k(x)$ is the volume of a sphere of radius $R_k(x)$ and C_d is the volume of the unit sphere in d dimension.

Equation (5) is the special case of (2) when K is the uniform kernel. The later function is defined as follows:

$$K(x) = \begin{cases} 1, & \text{if } |x| \leq 1 \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

3.2. Shannon's Entropy. Shannon's entropy [18] is a mathematical function, developed by Claude Shannon in 1948, that corresponds intuitively to the amount of information contained or delivered by an information source. This latter can be a text, an electrical signal, or any numerical file. For a source, which is a discrete random variable x with n symbols, each symbol X_i has a probability $p = \{p_1, \dots, p_N\}$ to appear. The entropy H of the source x is defined as

$$H(x) = -\mathbb{E}[\log_2(p)], \quad (7)$$

where \mathbb{E} is the expected value operator and \log_2 the logarithm in base 2.

Shannon's entropy can be found in the literature in various fields of research such as stock market [32], image segmentation [33], and cryptography [34].

The main reason for using Shannon's entropy is that it is a function that intuitively quantifies the amount of information in a variable. In order to remove irrelevant points, our simplification technique is based on the estimation of the amount of information.

4. Accuracy Evaluation

4.1. Simplification Error. In order to evaluate the accuracy of the novel simplification method, the geometric error between the original and simplified point cloud to be measured is used. To make a comparison between two surfaces, Cignoni et al. [35] developed a tool called Metro. Also, Pauly et al. [3] and Miao et al. [36] adopted a technique to measure simplification errors. In this paper, we evaluate the maximum geometric error and the average geometric error between the original model X and the simplified one X' .

The geometric max error is defined in paper [3] as

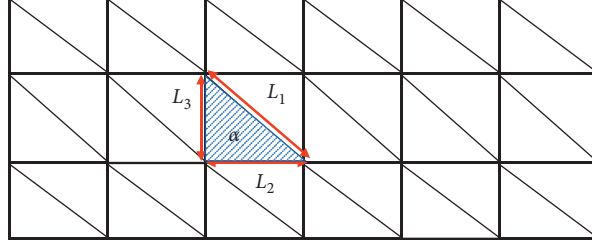


FIGURE 1: Measure of quality of the meshes.

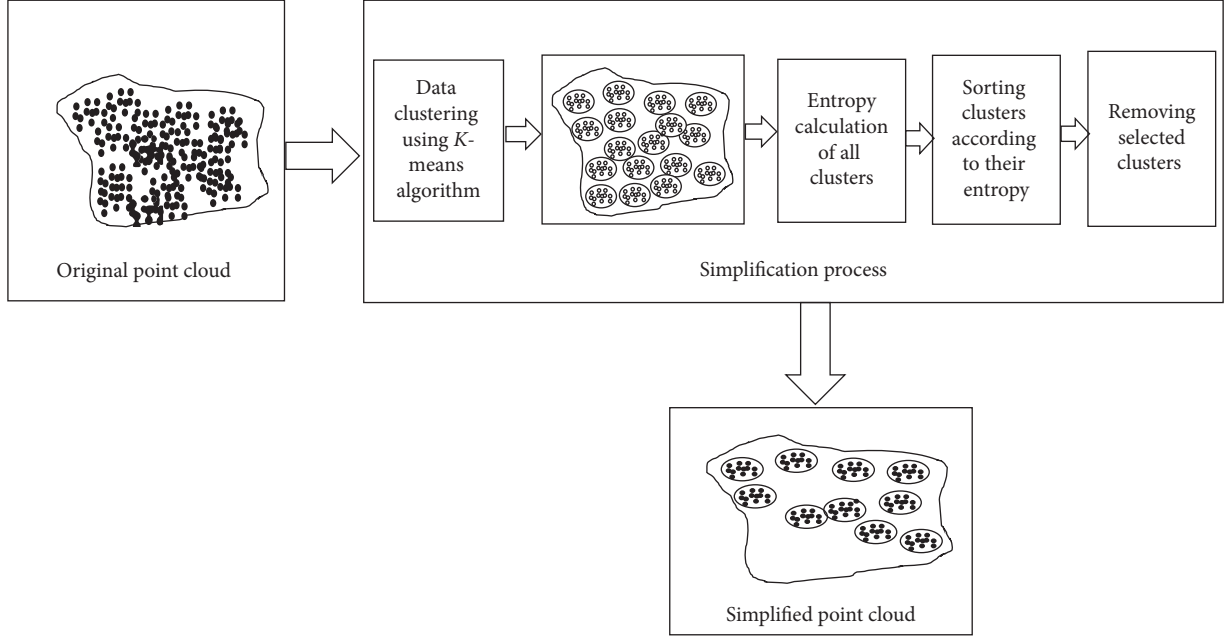


FIGURE 2: A diagram that shows how the new point cloud simplification method works.

$$\Delta_{\max}(X, X') = \max_{q \in X} d(q, X'). \quad (8)$$

The geometric average error is defined in paper [3] as

$$\Delta_{\text{avg}}(X, X') = \frac{1}{\|X\|} \sum_{q \in X} d(q, X'). \quad (9)$$

The corresponding normalized geometric errors can then be obtained by scaling the above error measures according to the model's diagonal of bounding box.

For each sample point $q \in X$, the geometric error $d(q, X')$ can be defined as the Hausdorff distance between the q on the original surface and its projection point q' on the simplified surface X' . The Hausdorff distance is defined as follows:

$$d = \max \left\{ \sup_{q \in X} \inf_{q' \in X'} d_e(q, q'), \sup_{q' \in X'} \inf_{q \in X} d_e(q, q') \right\}, \quad (10)$$

where $d(\cdot, \cdot)$ is an Euclidian distance. If N_q is the normal vector of point q and q' is the projection point on the simplified surface X' , the sign of d is the sign of $N_x \cdot (q' - q)$.

4.2. Surface Compactness. To measure the quality of the obtained meshes, Guezic [37] proposes a formula to compute the quality of the triangles. It is called compactness formula and is defined as follows:

$$\partial = \frac{4\sqrt{3}\alpha}{L_1^2 + L_2^2 + L_3^2}, \quad (11)$$

where L_i are the lengths of the edges of a triangle and α is the area of the triangles as shown in Figure 1. Note that this measure is equals to 1 for an equilateral triangle and 0 for a triangle whose vertices are collinear. According to [38], a triangle is of acceptable quality if $\partial \geq 0.6$.

5. The Simplification Method Proposed

The goal of 3D point cloud simplification is to choose the relevant and representative 3D points and remove redundant data points. In this work, the k -means clustering algorithm [23], which has been extensively used in the pattern recognition and machine learning literature, is extended to simplify dense points. As noted in Figure 2, the k -means algorithm is used to subdivide point cloud into c clusters.

```

Input
(i)  $X = \{x_1, x_2, \dots, x_N\}$ : the data sample (point cloud)
(ii)  $C[]$ : the array in which cluster indexes are stored
(iii)  $c$ : the number of clusters
(iv)  $n$ : the number of clusters to delete ( $n < c$ )
(v)  $E_{c \text{ min}} = E(R_1)$ : minimal entropy
(vi) Begin
(vii) Decomposing the initial set of points  $X$  into  $c$  small clusters denoting  $X = R_j (j = 1, 2, \dots, c)$ , using the  $k$ -means algorithm
(viii) For  $i = 1$  to  $n$ 
    For  $j = 2$  to  $c$ 
        Calculate global entropy of a cluster  $j$  by using all data samples in  $R_j = \{y_1, y_2, \dots, y_m\}$  according to equation (7), Note this entropy  $E(R_j)$ 
        If  $E(R_j) < E_{c \text{ min}}$  then
             $E_{c \text{ min}} \leftarrow E(R_j)$ 
             $\text{pos} \leftarrow j$ 
        End if
        For  $j = \text{pos}$  to  $c$ 
             $C[j] \leftarrow C[j + 1]$ 
        End for
    End for
(ix) End for,
     $X' = X\{C[i = 1, \dots, k - n]\}$ 
End.

```

ALGORITHM 1: Simplification of 3D point cloud based on the clustering algorithm and Shannon's entropy.

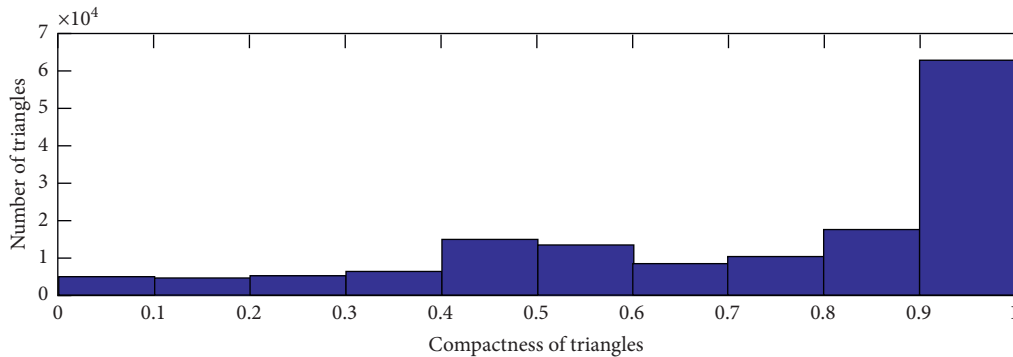


FIGURE 3: The compactness of the original surface of Bimba point set.

The size of the clusters is equal to 5% of the size of the original set of points. Subsequently, to select the clusters to be deleted, Shannon's entropy [18] will be used.

In this paper, we present a new robust approach based on clustering and Shannon's entropy. This approach allows keeping a uniform distribution of the points of the resulting cloud. In addition, it makes it easy to control the overall density of the coarse cloud by simply defining the size of the clusters. This approach, as shown in Figure 2, simplifies the 3D point cloud by saving the characteristics of the model presented by the original point cloud. Moreover, this simplification method preserves contours and sharp feature. Also, small features are maintained in the simplified point sets. This new method can be adapted to simplify nonuniformly distributed point sets.

Data clustering in small sets of points, using information theoretic clustering algorithm [21], makes it possible to

obtain groups containing points having a great similarity, which guarantees a good quality of simplification with an acceptable calculation time. To subdivide data sample into groups of 3D points, our technique of simplification is based on information theoretic clustering algorithm [21].

Next, the selection of relevant points in each cluster is done using Shannon's entropy [18]. The set of relevant points is the representative data samples that contain more information selected from the original dataset based on the proposed sample selection algorithm [1].

Compared to other simplification algorithms such as those of Shi et al. [16], Lee et al. [39], and Miao et al. [36], the advantages of the new algorithm are analyzed from many factors.

Firstly, our simplification method allows keeping the borders. This preservation of the integrity of original border is attributed to the nature of our method, as it uses Shannon entropy, which allows keeping clusters that have a high

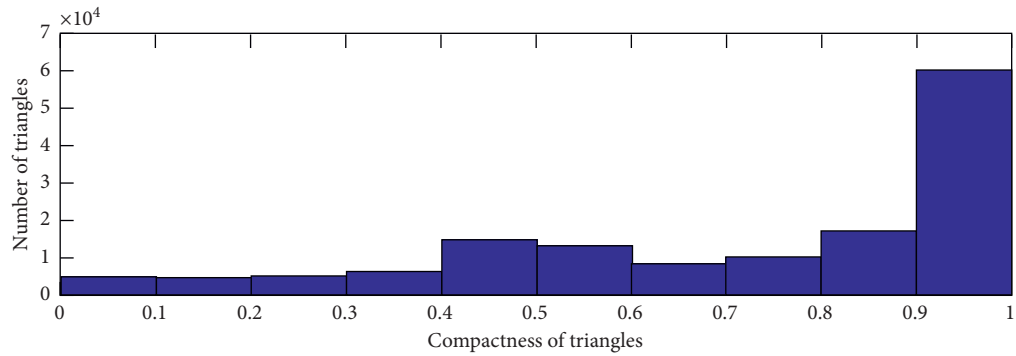


FIGURE 4: The compactness of the simplified surface of Bimba point set.



FIGURE 5: Various models simplified using the novel method. Left column: the original point clouds (triangulated). Right column: the simplified point sets (triangulated).

entropy value, and this is the case for borders. Secondly, the novel algorithm preserves compactness of the surface obtained from the simplified point cloud. This characteristic is measured by calculation of the percentage of compact

triangles using (11) proposed by Gueziec [37]. The construction of surfaces used in this article is realized using ball pivoting method [40].

The summary of contributions is as follows:

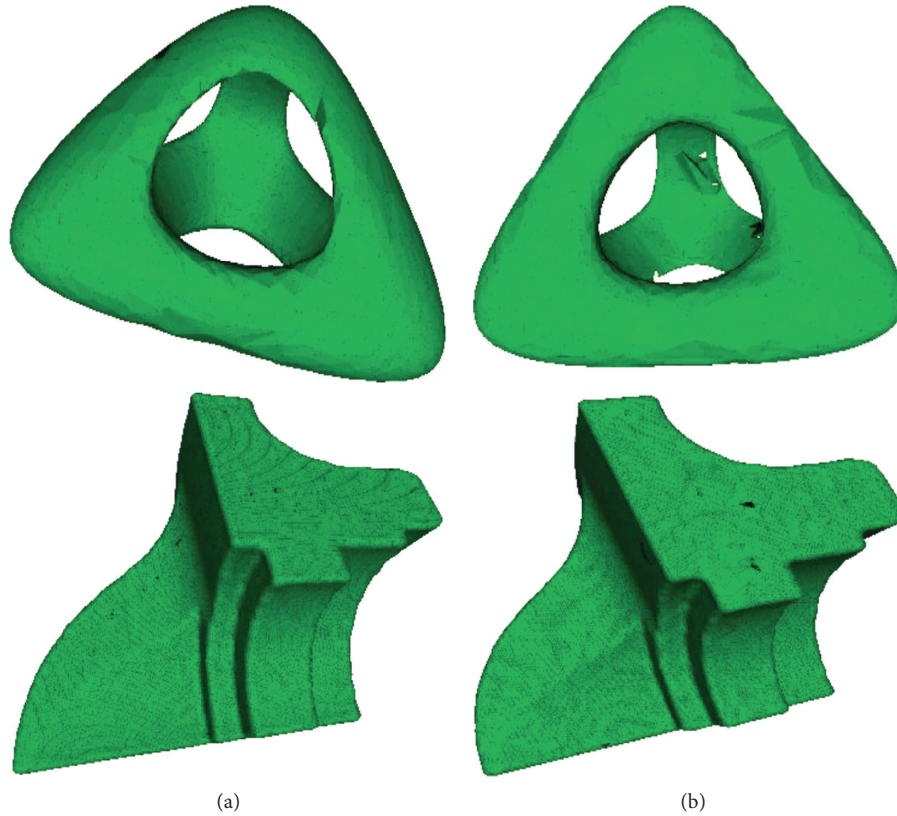


FIGURE 6: Fandisk and Genus model simplified using the novel method. (a) The original point sets (triangulated). (b) The simplified point cloud sets (triangulated).

- (i) Subdivide 3D dataset to clusters using k -mean clustering [23], which is widely applied in the pattern recognition and machine learning literature
- (ii) Shannon's entropy [18] is applied to select clusters of 3D point cloud, where it is applied to data classification
- (iii) The effectiveness and performance of the novel method are validated and illustrated through experimental results and comparison with other point sampling methods
- (iv) The new algorithm is validated and illustrated by the test of its efficiency and its performance through the realized experiments and the comparison with other simplification methods

The full description of the 3D point simplification algorithm, Algorithm 1, is as follows:

We note that the level of simplification of our approach is mainly determined by the user. This level is defined by the number (n) of clusters to be removed and the size of these clusters. In this work, the density of the clusters constituting the original point cloud is equal to 5% of the number of points of the original point cloud.

6. Results and Discussion

The new technique was implemented using MATLAB and MeshLab software. The algorithm for this new technique was

run on an Intel 64 core i5-2540M CPU 2.60 GHz PC. The David model and the Stanford Bunny model tested in this paper were developed at Stanford University [41]. The Fandisk, Max Planck, Genus, and Bimba models were obtained from the AIM@SHAPE database [42].

In order to approve the robustness of the proposed technique, we apply it using various 3D objects of different sizes and topologies. To ensure a better reconstruction, the surfaces of all the point clouds of the simplified objects were reconstructed using the MeshLab software [43].

6.1. Computing of Compactness. Computing of the compactness of the original and simplified surface of Bimba gives, respectively, 65.9498% and 66.7420%. The two values represent the percentages of the compact triangles of the two surfaces. The two previous results, Figures 3 and 4, show that this method ensures and increases the compactness of the simplified surface of Bimba. Calculation of the compactness is done using (11).

6.2. Results of the Novel Simplification Method. The novel strategy can deliver balanced point cloud. Figure 5 presents three cases of different models, where the David model shows that the original number of 3D points decreased from 182996 to 177454, the Max Planck model shows that the original points set diminished from 49089 to 48481, and the Bimba model was diminished from 74764 to 73458.

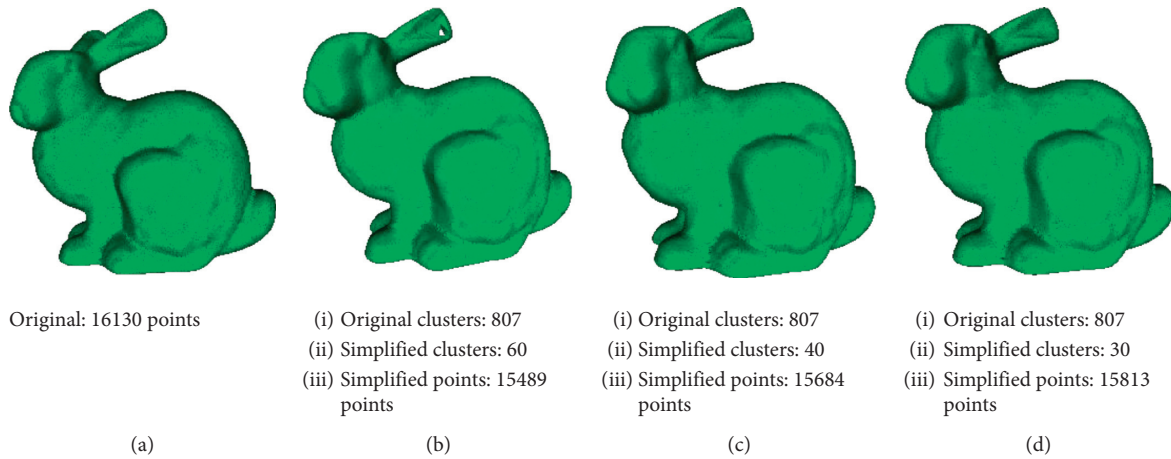


FIGURE 7: Simplification of the bunny model at different levels of detail (triangulated).

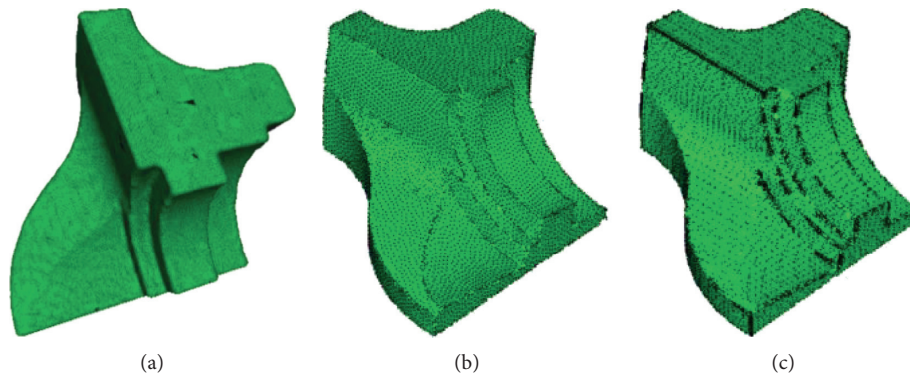


FIGURE 8: Simplification results of the Fandisk model (triangulated); the original number of points is 103,570. (a) Simplified using the novel method with parameters of cluster size and deleted cluster number (2590, 310) and (9759). (b) Simplified using Shi et al. method [16] with parameters of space interval and normal vector deviation threshold as (0.076, 0.13) and (9682). (c) Simplified using the 3D Grid method [39] with parameters of space interval and standard normal vector deviation as (0.0696, 0.41) and (9694).

Among the models tested in this paper, we used non-uniform objects such as the models of David, Bimba, and Max Planck. After simplification of these point clouds using the new method, we obtained satisfying results with the preservation of small details. Therefore, we can use the new technique for the simplification of nonuniform point clouds.

Figure 6 shows two models simplified using the new technique. These point sets have boundaries. The Genus model was simplified from 1234 to 1134, and the Fandisk model was reduced from 103568 to 93809. The experimental results obtained in Figure 6(b) indicate that the new technique can preserve the boundaries. Furthermore, the original sharp edges were well maintained, which again illustrates the superiority of our technique.

The novel method can produce some sparser level-of-detail point sets while preserving the small features and the sharp edges. In Figure 7, the sharp edges of the bunny model can be clearly seen when the point set is reduced from 16130 to 15813. This example demonstrates the good performance of the proposed method.

6.3. Comparison with Other Simplification Methods. The adaptive simplification of point cloud using k -means

clustering of Shi et al. [16] and 3D Grid method [39] was employed for a comparative study. The simplification results were triangulated with the software MeshLab [43]. In Figure 8, the famous Fandisk model was simplified. Since there was no redundant data in the original model (vertices 2502, faces 5000), we increased the vertices with the Geomagic Studio [44]. Finally, the number of vertices was 103 570. As shown in Figures 8 and 9, the new simplification technique gives better results either in terms of the number of points deleted or in terms of the error which presents the difference between original and simplified surfaces. We obtain uniformly distributed sparse sampling points in the flat areas and necessary dense points in the high curvature regions. The sharp edges of the Fandisk model are well maintained. The adaptive simplification of point cloud using k -means clustering of Shi et al. [16] and 3D Grid method [39] can also preserve sharp edges, but too many sampling points are assigned to the sharp edges. 3D Grid method [39] preserves fewer points in the flat areas, which leads to unbalance, unlike the proposed technique, as shown in Figure 4, which produces balanced simplified surfaces. On the other hand, as shown in Figure 4, the novel technique produces balanced simplified surfaces. Figures 8 and 9 and Table 1 show that the

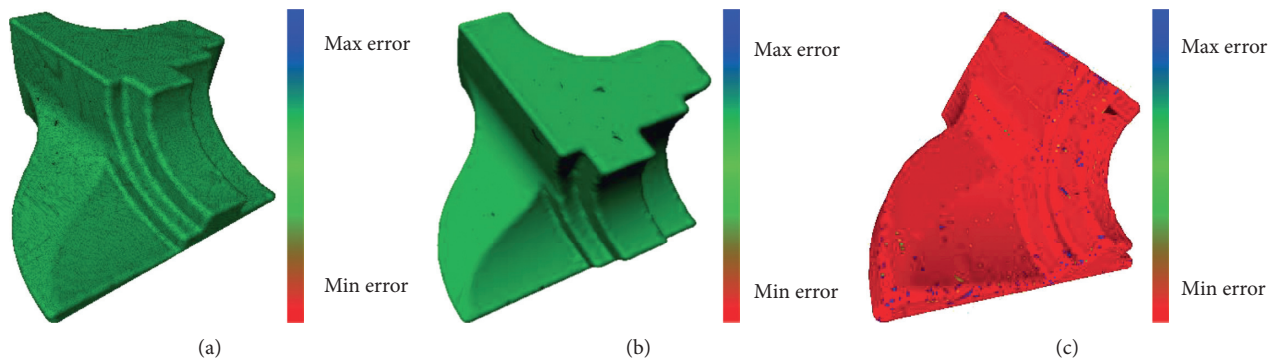


FIGURE 9: Simplification error of the Fandisk point cloud: (a) original point set, (b) simplified point set, (c) difference between original and simplified point cloud.

TABLE 1: Statistics of the new simplification technique applied to different 3D point clouds.

Object	Number of original points	Number of simplified points	Number of original clusters	Number of deleted clusters	Average error	Max error
David	182996	177454	458	20	0.000690	0.009997
Fandisk	103568	93809	2590	310	0.000434	0.009989
Bimba	74764	73458	3739	100	0.000002	0.005166
Max Planck	49089	48481	2455	50	0.000024	0.009972
Bunny	16130	15489	807	60	0.0001	0.009988
Genus	2496	2436	832	30	0.000113	0.009878

error of the original surface and the simplified surface obtained from the application of the new method is small compared to the error obtained from the method of Shi et al. [16] and 3D Grid method, which shows that our technique allows giving simplified point cloud close to that of the original one.

7. Conclusion

In this work, Shannon’s entropy, which has been largely used in data processing, and k -means clustering algorithm, which has been extensively used in pattern recognition and machine learning literature, have been extended to reduce 3D point cloud. This simplification procedure is achieved through the removal of redundant and less attractive 3D groups of points that have a minimum entropy value. Clusters are obtained using the k -means clustering algorithm. The new method is mainly impacted by two factors: number of original clusters and number of deleted clusters. The studies and illustrations made above show that, since both factors are regulated, this new method can be applied to different levels of detail and different forms of 3D point clouds and produce well-balanced surfaces, which makes it robust, as the results show.

Data Availability

The experimental data, which are in the form of 3D objects, used to support the results of this study are downloadable from the AIM@SHAPE database included in references.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] A. Mahdaoui, A. Bouazi, A. Marhraoui Hsaini, and E. H. Sbai, “Entropic method for 3D point cloud simplification,” in *Innovations in Smart Cities and Applications*, pp. 613–621, Springer, Cham, Switzerland, 2018.
- [2] W. Boehler and A. Marbs, “3D scanning instruments,” in *Proceedings of the CIPA WG, 2002*, pp. 9–18, Corfu, Greece, September 2002.
- [3] M. Pauly, M. Gross, and L. P. Kobbelt, “Efficient simplification of point-sampled surfaces,” in *Proceedings of the IEEE Visualization, 2002*, pp. 163–170, IEEE, Boston, MA, USA, October 2002.
- [4] J. Wu and L. Kobbelt, “Optimized sub-sampling of point sets for surface splatting,” *Computer Graphics Forum*, vol. 23, no. 3, pp. 643–652, 2004.
- [5] L. Linsen, “Point cloud representation,” Technical report, Karlsruhe Institute of Technology, Karlsruhe, Germany, 2001.
- [6] T. K. Dey, J. Giesen, and J. Hudson, “Decimating samples for mesh simplification,” in *Proceedings of the 13th Canadian Conference on Computational Geometry, 2001*, pp. 85–88, Ontario, Canada, August 2001.
- [7] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva, “Point set surfaces,” in *Proceedings of the Visualization, 2001*, pp. 21–28, IEEE, San Diego, CA, USA, October 2001.
- [8] 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—SIGGRAPH'97, pp. 209–216, Los Angeles, CA, USA, August 1997.
- [9] W. Xuan, X. Hua, X. Chen, J. Zou, and X. He, “A new progressive simplification method for point cloud using local entropy of normal angle,” *Journal of the Indian Society of Remote Sensing*, vol. 46, no. 4, pp. 581–589, 2018.
 - [10] N. Leal, E. Leal, and S. T. German, “A linear programming approach for 3D point cloud simplification,” *IAENG International Journal of Computer Science*, vol. 44, no. 1, pp. 60–67, 2017.
 - [11] C. Ji, Y. Li, J. Fan, and S. Lan, “A novel simplification method for 3D geometric point cloud based on the importance of point,” *IEEE Access*, vol. 7, pp. 129029–129042, 2019.
 - [12] A. Mahdaoui, A. Bouazi, A. M. Hsaini, and E. H. Sbai, “Comparison of K -means and fuzzy C -means algorithms on simplification of 3D point cloud based on entropy estimation,” *Advances in Science, Technology and Engineering Systems Journal*, vol. 2, no. 5, pp. 38–44, 2017.
 - [13] C. Moening, C. Moening, and N. A. Dodgson, “A new point cloud simplification algorithm,” in *Proceedings of 3rd IASTED Conference on Visualization, Imaging and Image Processing*, pp. 1027–1033, Benalmádena, Spain, September 2003.
 - [14] E. Diday, G. Govaert, Y. Lechevallier, and J. Sidi, “Clustering in pattern recognition,” in *Digital Image Processing*, pp. 19–58, Springer, Dordrecht, Netherlands, 1981.
 - [15] C. E. Brodley and A. P. Danyluk, “Machine learning,” in *Proceedings of the 18th International Conference (ICML 2001)*, Williamstown, MA, USA, June 2001.
 - [16] B.-Q. Shi, J. Liang, and Q. Liu, “Adaptive simplification of point cloud using k -means clustering,” *Computer-Aided Design*, vol. 43, no. 8, pp. 910–922, 2011.
 - [17] R. Xu and D. C. Wunsch, *Clustering*, IEEE Press, Piscataway, NJ, USA, 2008.
 - [18] C. E. Shannon, “A mathematical theory of communication,” *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.
 - [19] D. Guo, S. Shamai, and S. Verdu, “Mutual information and minimum mean-square error in Gaussian channels,” *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1261–1282, 2005.
 - [20] D. Erdogmus, R. Agrawal, and J. C. Principe, “A mutual information extension to the matched filter,” *Signal Processing*, vol. 85, no. 5, pp. 927–935, 2005.
 - [21] V. V. Vikjord and R. Jenssen, “Information theoretic clustering using a k -nearest neighbors approach,” *Pattern Recognition*, vol. 47, no. 9, pp. 3070–3081, 2014.
 - [22] J. Wang, X. Li, and J. Ni, “Probability density function estimation based on representative data samples,” in *Proceedings of the 2011 IET International Conference on Communication Technology and Application (ICCTA 2011)*, pp. 694–698, IEEE, Beijing, China, October 2011.
 - [23] J. B. MacQueen, “Some methods for classification and analysis of multivariate observations,” in *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, Berkeley, CA, USA, 1967.
 - [24] J. Jiao, K. Venkat, Y. Han, and T. Weissman, “Maximum likelihood estimation of functionals of discrete distributions,” *IEEE Transactions on Information Theory*, vol. 63, no. 10, pp. 6774–6798, 2017.
 - [25] E. Parzen, “On estimation of a probability density function and mode,” *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.
 - [26] M. Rosenblatt, “Remarks on some nonparametric estimates of a density function,” *The Annals of Mathematical Statistics*, vol. 27, no. 3, pp. 832–837, 1956.
 - [27] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman & Hall, London, UK, 1986.
 - [28] H.-G. Müller and A. Petersen, “Density estimation including examples,” in *Wiley StatsRef: Statistics Reference Online*, pp. 1–12, John Wiley & Sons, Chichester, UK, 2016.
 - [29] C. Li, S. Zhang, H. Zhang et al., “Using the K -nearest neighbor algorithm for the classification of lymph node metastasis in gastric cancer,” *Computational and Mathematical Methods in Medicine*, vol. 2012, Article ID 876545, 11 pages, 2012.
 - [30] Y. Lou, “Storage and allocation of English teaching resources based on k -nearest neighbor algorithm,” *International Journal of Emerging Technologies in Learning (ijET)*, vol. 14, no. 17, pp. 102–113, 2019.
 - [31] J. E. Vidueira Ferreira, C. H. S. da Costa, R. M. de Miranda, and A. F. de Figueiredo, “The use of the k nearest neighbor method to classify the representative elements,” *Educación Química*, vol. 26, no. 3, pp. 195–201, 2015.
 - [32] R. Gu, “Multiscale Shannon entropy and its application in the stock market,” *Physica A: Statistical Mechanics and Its Applications*, vol. 484, pp. 215–224, 2017.
 - [33] M. S. R. Naidu, P. Rajesh Kumar, and K. Chiranjeevi, “Shannon and Fuzzy entropy based evolutionary image thresholding for image segmentation,” *Alexandria Engineering Journal*, vol. 57, no. 3, pp. 1643–1655, 2018.
 - [34] C. E. Shannon, “Communication theory of secrecy systems,” *Bell System Technical Journal*, vol. 28, no. 4, pp. 656–715, 1949.
 - [35] P. Cignoni, C. Rocchini, and R. Scopigno, “Metro: measuring error on simplified surfaces,” *Computer Graphics Forum*, vol. 17, no. 2, pp. 167–174, 1998.
 - [36] Y. Miao, R. Pajarola, and J. Feng, “Curvature-aware adaptive re-sampling for point-sampled geometry,” *Computer-Aided Design*, vol. 41, no. 6, pp. 395–403, 2009.
 - [37] A. Guezic, “Locally toleranced surface simplification,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, no. 2, pp. 168–189, 1999.
 - [38] R. E. Bank, *PLTMG: A Software Package for Solving Elliptic Partial Differential Equations*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998.
 - [39] K. H. Lee, H. Woo, and W. Suk, “Point data reduction using 3D grids,” *The International Journal of Advanced Manufacturing Technology*, vol. 18, no. 3, pp. 201–210, 2001.
 - [40] A. Mahdaoui, A. Marhraoui Hsaini, A. Bouazi, A. M. Hsaini, and E. H. Sbai, “Comparative study of combinatorial 3D reconstruction algorithms,” *International Journal of Engineering Trends and Technology*, vol. 48, no. 5, pp. 247–251, 2017.
 - [41] M. Levoy, “The digital michelangelo project: 3D scanning of large statues,” in *Proceedings of ACM SIGGRAPH 2000*, pp. 131–144, New Orleans, LA, USA, July 2000.
 - [42] “AIM@SHAPE Database,” 2019.
 - [43] G. R. P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, and F. Ganovelli, “MeshLab: an open-source mesh processing tool,” in *Proceedings of the 6th Eurographics Italian Chapter Conference*, pp. 129–136, Fisciano, Italy, July 2008.
 - [44] *Geomatic User Guide*. 2013.