frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 24 September 2024
DOI 10.3389/fphy.2024.1471077

Check for updates

OPEN ACCESS

EDITED BY
Xiaoming Duan,
Harbin Institute of Technology, China

REVIEWED BY
Yinghua Shen,
Chongqing University, China
Hengrong Ju,
Nantong University, China
Hongpeng Wu,
Shanxi University, China

*CORRESPONDENCE
Kaijie Xu,
kjxu@xidian.edu.cn

RECEIVED 26 July 2024
ACCEPTED 13 September 2024
PUBLISHED 24 September 2024

CITATION
Hu A, Xu K, Yin X and Wang D (2024) LiDAR point cloud simplification strategy utilizing probabilistic membership.
Front. Phys. 12:1471077.
doi: 10.3389/fphy.2024.1471077

LiDAR point cloud simplification strategy utilizing probabilistic membership

Ao Hu[1], Kaijie Xu[1,2]*, Xukun Yin[2] and Di Wang[3]

[1]School of Electronic Engineering, Xidian University, Xi'an, China, [2]Hangzhou Institute of Technology, Xidian University, Hangzhou, China, [3]School of Software Engineering, Xi'an Jiaotong University, Xi'an, China

With the continuous progress of information acquisition technology, the volume of LiDAR point cloud data is also expanding rapidly, which greatly hinders the subsequent point cloud processing and engineering applications. In this study, we propose a point cloud simplification strategy utilizing probabilistic membership to address this challenge. The methodology initially develops a feature extraction scheme based on curvature to identify the set of feature points. Subsequently, a combination of k-means clustering and Possibilistic C-Means is employed to partition the point cloud into subsets, and to simultaneously acquire the probabilistic membership information of the point cloud. This information is then utilized to establish a rational and efficient simplification scheme. Finally, the simplification results of the feature point set and the remaining point set are merged to obtain the ultimate simplification outcome. This simplification method not only effectively preserves the features of the point cloud while maintaining uniformity in the simplified results but also offers flexibility in balancing feature retention and the degree of simplification. Through comprehensive comparative analysis across multiple point cloud models and benchmarking against various simplification methods, the proposed approach demonstrates superior performance. Finally, the proposed algorithm was critically discussed in light of the experimental results.

KEYWORDS

point cloud simplification, possibilistic c-means (PCM), feature extraction, probabilistic membership, LiDAR point cloud

1 Introduction

In modern robotics applications, the utilization of 3-Dimensional (3D) maps of environments is widespread. However, the substantial storage demands inherent in dense 3D maps necessitate point cloud simplification for efficient storage and transmission [1]. This process involves condensing intricate LiDAR point clouds while retaining essential spatial information, enabling streamlined data handling and facilitating seamless integration into various robotic applications.

Moreover, beyond mere storage considerations, point cloud simplification offers significant advantages for subsequent data processing endeavors. By reducing the complexity of the point cloud representation, computational tasks such as object recognition, path planning, and navigation become more expedient and resource-efficient. This streamlined data structure not only enhances the operational efficiency of robots but also contributes to the overall robustness and reliability of autonomous systems in dynamic environments [2].

In essence, LiDAR point cloud simplification stands as a fundamental technique in modern robotics, bridging the gap between the rich spatial information captured by sensors and the practical constraints of computational resources and real-world application demands [3]. Its role in facilitating efficient storage, seamless data processing, and enhanced robotic performance underscores its indispensability in advancing the capabilities and applicability of robotic technologies across diverse domains. Therefore, there is an urgent need for various effective point cloud simplification solutions to address this issue.

From a topological perspective, solutions for point cloud simplification can be broadly classified into two categories: grid-based and point-based methods. Historically, grid-based methods [4] had been widely preferred; however, these approaches require grid reconstruction, resulting in significant computational resource utilization. As a result, point-based simplification techniques have gradually gained popularity, particularly for handling large-volume point cloud data in contemporary scenarios. In point-based simplification methods, preserving the geometric features of the point cloud is of utmost importance. Martin et al. [5] proposed a uniform grid down-sampling method; however, this approach does not consider the retention of feature points. Consequently, the simplified point cloud may lose the intricate details of the surface. To address this limitation, Lee et al. [6] introduced the use of normal deviation as feature information based on uniform meshing. Similarly, Alexa et al. [7] presented a simplified method that relies on moving least squares fitting, but it also falls short in retaining surface feature points. Song et al. [8] improved Alexa's method by introducing surface feature detection. In addition, there are several clustering-based simplification methods [9–12]. These methods mostly combine one or more feature information, such as normal, curvature, and information entropy, to effectively preserve the geometric features of point clouds.

In recent years, researchers have introduced methods utilizing Laplacian graphs [13–15] to streamline point clouds. While these approaches achieve effective simplification, they often come with high computational complexity. Additionally, several deep learning-based methods for point cloud simplification or sampling have been proposed [16–18] for specific tasks such as classification, registration, and recognition. For instance, a recent study introduced an innovative sampling method based on skeleton-aware learning [19], which employs the object's skeletal information as prior knowledge to better preserve its geometric shape and topological structure during the sampling process. However, these methods continue to face challenges in balancing uniformity with feature preservation.

For the optimal simplification solution, we believe it needs to meet several key requirements [20]: controllability of simplification ratio, preservation of geometric features, and uniform distribution of the results. Upon reviewing the aforementioned methods, we found that earlier approaches focused primarily on achieving uniform point distribution after simplification, whereas more recent methods tended to emphasize the preservation of point cloud geometric features. The controllability of the simplification ratio determines the method's versatility. Therefore, our goal is to develop a simplification technique that balances these three critical requirements.

To meet these requirements, this paper proposes a point cloud simplification strategy based on probabilistic membership. This approach employs the Z-score model to standardize the curvature of the point cloud, facilitating the assessment of the feature information. Following the scoring process, the point cloud is categorized into feature and non-feature subsets. Subsequently, the two point clouds are subclustered using k-means and Possibilistic C-Means (PCM) [21], and probabilistic membership is obtained. Based on probabilistic membership, we develop a hierarchical subcluster simplification scheme. The simplified results of all subclusters are combined to produce the final output. While maintaining the detailed features of the point cloud and the uniformity of simplification outcomes, the proposed algorithm allows users to adjust the level of point cloud reduction and geometric feature retention.

In summary, the main contributions of this study are as follows:

1. We developed a feature selection scheme based on curvature and Z-score models, effectively isolating feature points within the point cloud to facilitate subsequent processing.
2. We proposed a hierarchical subcluster simplification approach based on probabilistic membership. By integrating probabilistic membership into subcluster division, this method ensures reasonable and uniform simplification results while offering flexible control over the simplification ratio.

Additionally, we validated the feasibility and effectiveness of the proposed approach through theoretical analysis and experimental results.

The paper is organized as follows: Section 2 presents the proposed algorithm and related theory. Section 3 discusses the algorithm's parameter settings and presents the experimental comparison analysis. Lastly, Section 4 provides a summary of the paper.

2 Simplification strategy utilizing probabilistic membership

In this section, we focus on two key components of the proposed point cloud reduction strategy: the extraction of feature point clouds from the original point cloud, and the point cloud reduction algorithm based on probabilistic membership. After that, we will explain how the simplification scheme can achieve complete point cloud simplification around these two core parts. The detailed contents are as follows.

2.1 Feature extraction of the point cloud

To preserve the geometric characteristics of the point cloud, it is essential to extract the feature point cloud from the original point cloud. We opted to utilize the curvature metric as it is a commonly used method. Considering the tight computing resources during large-scale point cloud processing, we employed the K-Nearest Neighbors (KNN) [22] search to obtain neighborhood information. By integrating this neighborhood information with principal component analysis (PCA) [23], we were able to efficiently estimate the curvature of the point cloud. Subsequently, based on the curvature information, feature points were selected based on their deviation from the neighborhood's average curvature, which was quantified using the Z-score model [24]:

$$Z = \frac{C - \mu}{\delta}$$

where C represents the curvature of a 3D point, μ is the average curvature in the neighborhood of this point, and δ is the standard deviation of curvature in this neighborhood. The aim of applying z-score is to measure how many standard deviations the original data differs from the overall mean of the data. After calculating the Z-score for each point in the point cloud, a suitable deviation threshold Z_{th} is chosen. Points with scores exceeding this threshold and high curvature are then identified as feature points. We set a proportion of high curvature points to determine high curvature points.

2.2 A point cloud hierarchical simplification algorithm based on probabilistic membership

At this stage, we introduce the PCM clustering method for decomposing point clouds into subclusters and emphasize the acquisition of probabilistic membership for each subcluster. Then, a hierarchical subcluster simplification scheme is proposed. The ultimate result is the aggregate of the simplified outcomes of each subcluster.

2.2.1 Acquisition of probabilistic membership

Probabilistic membership is a key concept in the context of the PCM clustering algorithm, which represents an advancement of the classic Fuzzy C-Means (FCM) [25] clustering algorithm. The PCM algorithm aims to cluster a given point cloud $\boldsymbol{P}$ ($\boldsymbol{P} \in R^{N\times 3}$) by optimizing the following objective function:

$$\begin{aligned} J(\mu, \boldsymbol{v}) = &\sum_{i=1}^{N}\sum_{j=1}^{C}\mu_{ij}\left\|\boldsymbol{p}_i - \boldsymbol{v}_j\right\|^2 \\ &+\sum_{j=1}^{C}\eta_j\sum_{i=1}^{N}\left(\mu_{ij}\log\mu_{ij} - \mu_{ij}\right)^2 \\ &\text{s.t.} \\ &0 \le \mu_{ij} \le 1 \end{aligned}$$

where $\boldsymbol{v}_j$ denotes the prototype for the jth cluster, μ_{ij} is the probabilistic membership degree of the point $\boldsymbol{p}_i$ ($\boldsymbol{p}_i \in \boldsymbol{P}$) belonging to the jth cluster, $\|\bullet\|$ denotes the Euclidean distance. Furthermore, η_j is a customizable constant, which can generally be expressed as:

$$\eta_j = \frac{\sum_{i=1}^{N}\mu_{ij}\left\|\boldsymbol{p}_i - \boldsymbol{v}_j\right\|^2}{\sum_{i=1}^{N}\mu_{ij}}$$

In the iterative process of objective function minimization, the update formulas for the prototype and probabilistic membership are as follows:

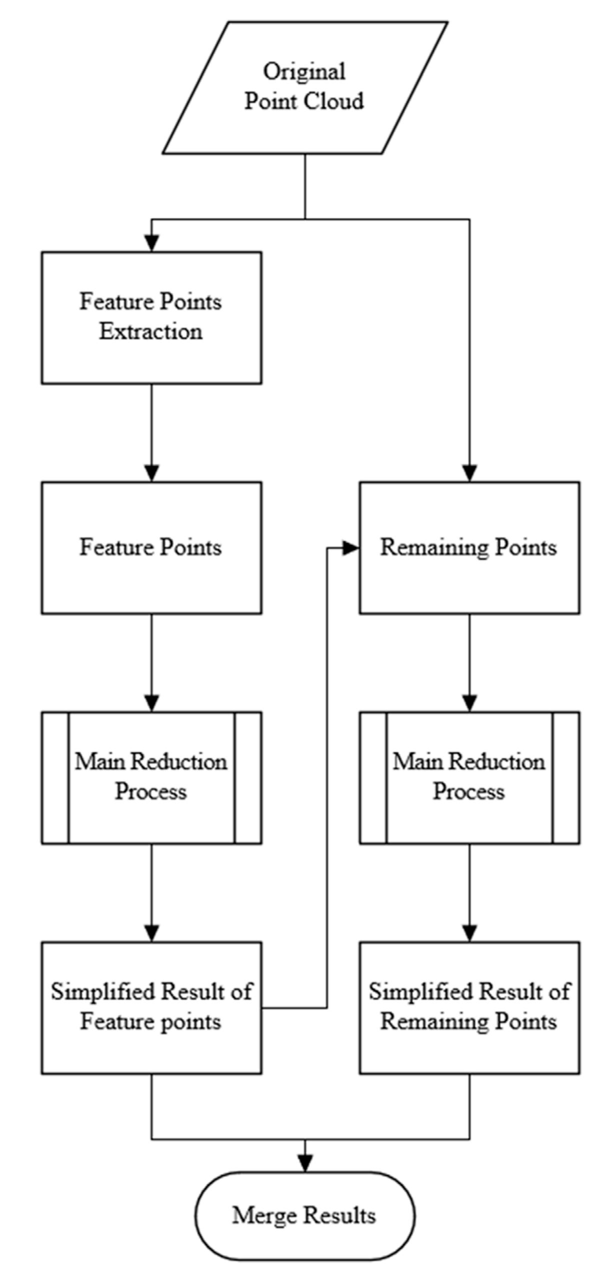

FIGURE 1
An overall model: main processing phases.

$$\boldsymbol{v}_j = \frac{\sum_{i=1}^{N}\mu_{ij}\boldsymbol{p}_i}{\sum_{i=1}^{N}\mu_{ij}}$$

$$\mu_{ij} = \exp\left\{-\frac{\left\|\boldsymbol{p}_i - \boldsymbol{v}_j\right\|^2}{\eta_j}\right\}$$

In order to minimize computational expense, we employ minibatch k-means [26] clustering to derive the ultimate prototype matrix. Subsequently, we utilize the formula in PCM to compute the probabilistic membership. Concurrently, the point cloud is partitioned into numerous subclusters, with the maximum probabilistic membership degree of each point serving as an index. To simplify the subcluster simplification task, parallelization with multiple threads can be implemented to enhance the algorithm efficiency.

2.2.2 Subcluster simplification based on maximum probabilistic membership

For established subclusters, we have developed a hierarchical simplification scheme. This scheme applies the maximum probabilistic membership of subcluster members to the prototype to systematically obtain the outermost member points of the subcluster. The specific operations are as follows:

1) The data points are arranged based on their maximum probabilistic membership.
2) A threshold percentage r_{out} is set, and data points within this threshold are identified as outer points.
3) The k-means clustering method is applied to the outer points, and the nearest point to the cluster center is selected as the output.
4) The same operations are repeated for non-outer points within the subcluster until a specified number of subcluster members (N_{th}) is reached.
5) The results from each iteration are consolidated to obtain the simplified output of the subcluster.

In the above process, we opted to persist with k-means clustering due to its high efficiency and reliable performance. Moreover, prior to initiating each iteration, it is imperative to pre-calculate the quantity of output points in order to effectively regulate the simplification ratio.

2.3 The overall structure of the proposed simplified solution

The overall structure of the proposed simplification scheme is shown in Figure 1. The feature point extraction and main simplification processes correspond to the parts 2.1 and 2.2 mentioned earlier in Section 2. The simplification scheme divides the final output into two parts: the simplified results of the feature point cloud and the simplified results of the remaining point cloud. The remaining points refer to the points in the original point cloud that are left after the feature point cloud is simplified.

The input parameters of the algorithm mainly consist of the original point cloud, the simplification ratio, and the feature preservation ratio r_f. By adjusting r_f, users can effectively control the degree of feature preservation. Additionally, the simplification ratio of the remaining point cloud can be calculated based on the simplification ratio and the feature preservation ratio.

3 Experimental studies

In this study, we applied the proposed methodology, conducted data visualization, and estimated errors, all of which were programmed using MATLAB. All experimental datasets are sourced from the point cloud dataset built by Stanford University [27]. In the following content, the parameter settings and result images of each stage of the proposed algorithm will be displayed, as well as the final comparative experiment.

TABLE 1 Simulation parameters.

Parameter	Value
Number of neighbors in KNN search (K)	15
Z-score threshold in feature extraction (Z_{th})	0.6
Proportion of high curvature points	30%
Number of clusters (C)	5–150
Proportion of outer subclusters (r_{out})	60%
Number of subcluster members before iteration terminates (N_{th})	80
Default feature point reduction ratio (r_f)	40%

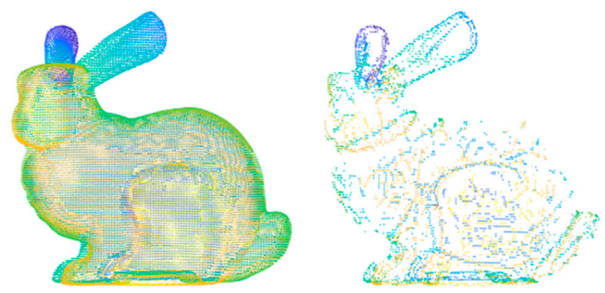

FIGURE 2
Original point cloud and the extracted feature point cloud of the bunny model.

3.1 Experimental parameter settings

In our experiments, the proposed algorithm involves multiple stages, each involving specific parameter values. These parameters and their experimental values are listed in Table 1.

3.2 Comparison of the experimental results

In this section, we present a simplified example that illustrates the proposed method. We also conducted a comparative analysis of various down-sampling methods to assess the effectiveness of the proposed algorithm. The compared methods included random down-sampling (RD), uniform grid down-sampling (UG) [5], Laplacian graphs (LG) [2], and a simplification algorithm based on a partitioning strategy (PS) [28]. Next, we analyzed the performance of the proposed method from two aspects of simplified results: the surface reconstruction model [29] and average geometric error [30].

In Figure 2, we show the original point cloud and the extracted feature point cloud of the rabbit model. When the simplification

10.3389/fphy.2024.1471077

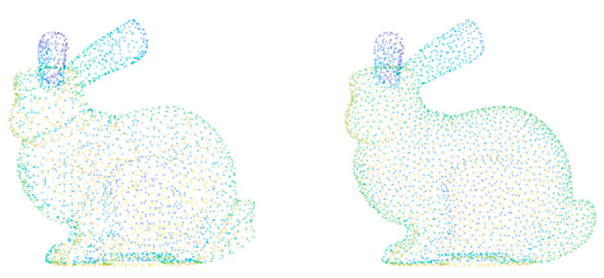

FIGURE 3
Simplification results of the bunny model.

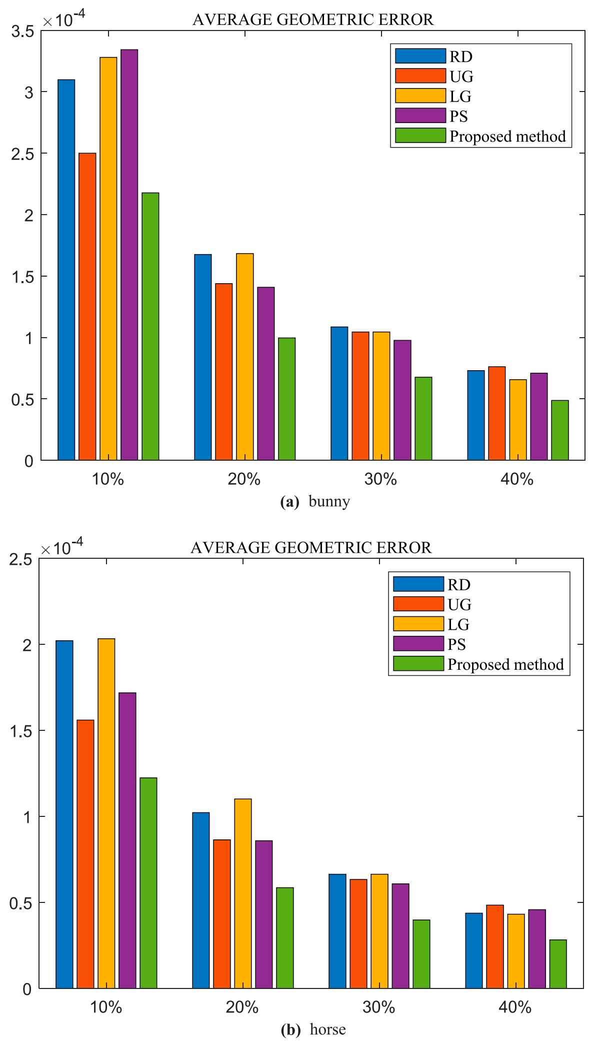

FIGURE 4
Average geometric error for each method **(A)** bunny **(B)** horse.

rate is 10%, we compared the simplification results of different r_f. In Figure 3, the value of r_f on the left is 0.4, and the value of r_f on the right is 0.1. Upon analysis of Figure 3, it is evident that the image on the left, with a higher r_f value, exhibits a greater number of details, particularly in the depiction of ears. The image on the right describes more of the overall distribution. Users can freely set the r_f value according to their needs.

We evaluated the proposed method and the aforementioned methods using both bunny and horse models. Four reduction rates (10%, 20%, 30%, and 40%) were employed for the comparative analysis. Due to the limitations of UG and PS in controlling the reduction rate, we had to consistently adjust the parameters to approximate the desired reduction rate. The average geometric error for each method is illustrated in Figure 4. After analyzing the figure, we discovered that the average geometric error of the proposed method is significantly lower than that of other methods under different reduction rates. Specifically, when the reduction rate is low, methods like PS and LG, which are able to retain features, demonstrate poor performance. This can be attributed to their inability to adequately control the proportion of feature points, leading to an excess retention of feature points which subsequently impacts the uniformity of the results. This observation indirectly emphasizes the substantial advantages of our algorithm in achieving uniform results. Similarly, the UG method focuses more on ensuring the uniform distribution of the simplified point cloud. However, when the simplification rate is high and the resulting point cloud contains a sufficient number of points, the description of fine details becomes more critical, and the advantages of UG become less pronounced.

To assess the extent of feature retention, we employed the aforementioned method to simplify the Armadillo model at a 20% reduction rate. Subsequently, the simplified point clouds were used to reconstruct mesh models, with the results depicted in Figure 5. Through comparative analysis of the reconstruction models of the original point cloud, we discovered that the reconstruction performance of RD was inadequate. This was attributed to its emphasis on speed, which led to a lack of detail and uniformity in the simplified results. In the case of UG, its focus on ensuring uniformity in the simplified results produced a relatively smooth reconstruction model, but the details of the armadillo's leg muscles appeared blurry. PS utilized curvature as feature information, resulting in a reconstruction model better at retaining sharp features, such as the curve of the armadillo's forehead. However, the details on the thighs remained insufficiently clear. LG excelled in retaining detail, but exhibited noticeable roughness in flat areas. The proposed method's reconstruction model outperforms other methods in terms of feature retention. Moreover, it exhibits good smoothness in flat areas.

In summary, the method described in this study offers a flexible and controllable reduction rate and effectively ensures the uniform distribution and retention of features in the reduction results. Additionally, our approach allows for parameter adjustment to modify the level of feature retention in the point cloud, catering to varying requirements. These advantages are particularly evident in the comparative experiments.

However, the superior performance comes at the cost of increased computational complexity. Specifically, when processing large-scale point clouds, the process of obtaining probabilistic membership after clustering can result in substantial computational costs. To mitigate this issue, specialized data structures or parallel processing strategies can be employed to reduce runtime and enhance the algorithm's efficiency.

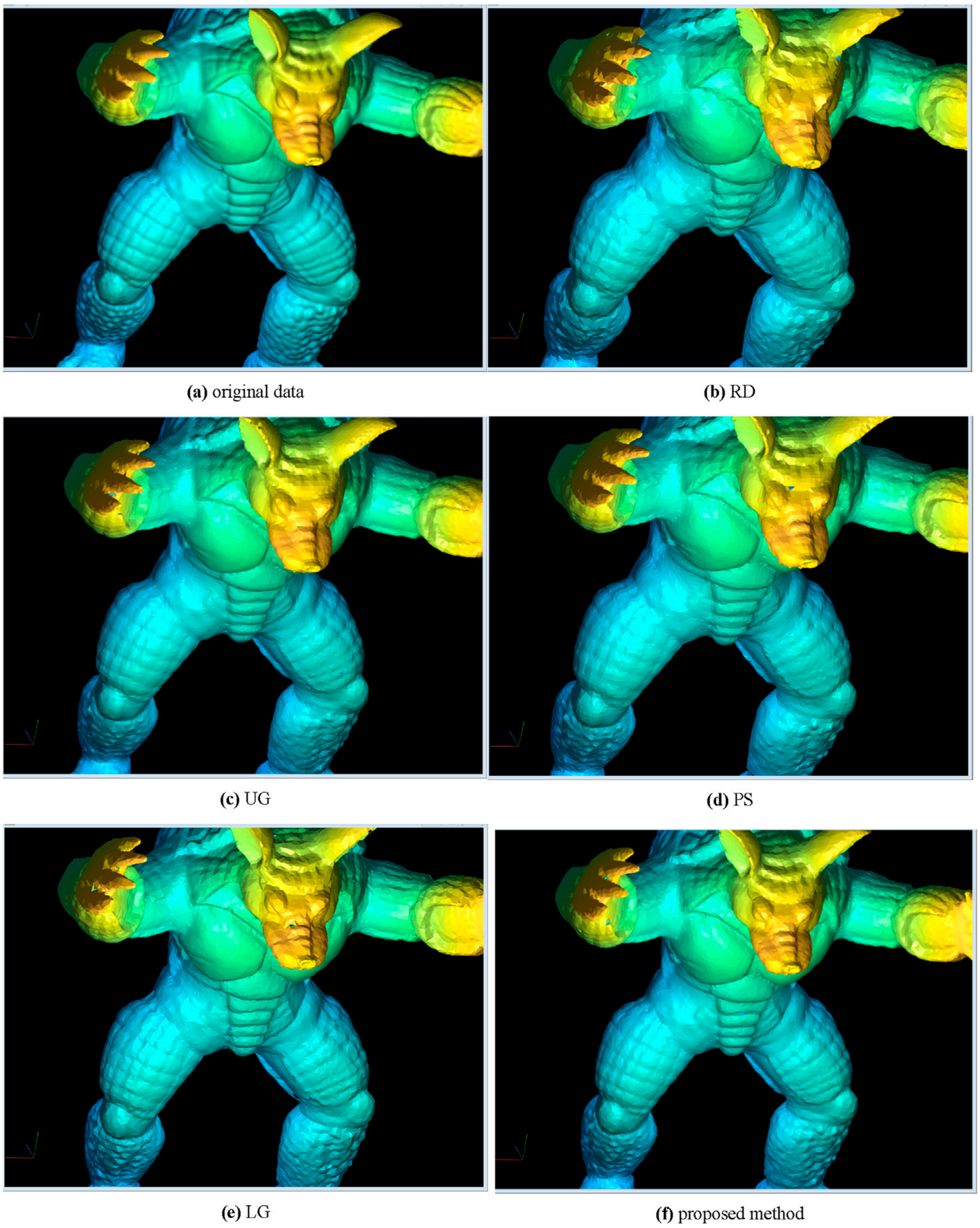

FIGURE 5
Mesh models with the simplified point clouds **(A)** original data **(B)** RD **(C)** UG **(D)** PS **(E)** LG **(F)** proposed method.

4 Conclusion

This study introduces a hierarchical point cloud simplification strategy based on probabilistic membership. This scheme first simplifies the feature point cloud independently to preserve geometric features, and then simplifies the remaining points. In this process, a neighborhood curvature deviation model was designed to identify feature points, and probabilistic membership was introduced in subsequent simplifications as the basis to divide the point cloud into subclusters. For subclusters, we propose a hierarchical simplification algorithm based on probabilistic membership characteristics, aiming to control the number of output points while achieving uniform distribution.

In the control experiments, the proposed method effectively preserves geometric features while maintaining uniform distribution of output points. Additionally, it offers flexibility in adjusting the reduction rate and feature retention rate to cater to user preferences. In future work, we plan to further investigate the adaptive optimization of parameter selection, such as the number of clusters, and explore strategies to reduce computational overhead. We also welcome constructive feedback to help continuously refine and improve our research.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

AH: Writing–original draft, Visualization, Software, Data curation. KX: Writing–review and editing, Supervision, Methodology. XY: Writing–review and editing, Validation, Funding acquisition, Conceptualization. DW: Writing–review and editing, Supervision.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was supported the National Key Research and Development Program of China (No. 2021YFF0704600), in part by the Hangzhou Science and Technology Development Project (202204T04), in part by the National Natural Science Foundation of China under Grant Nos. 62101400, 92367206, 72171069, 42101330, 62475204, and 62105252, Natural Science Foundation of Shaanxi Province (Nos. 2023-YBSF-452 and 2023-YBGY-099), in part by the China Postdoctoral Science Foundation under Grant 2023M732743, in part by the Shaanxi Fundamental Science Research Project for Mathematics and Physics under Grant 22JSQ032.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Wiesmann L, Milioto A, Chen X, Stachniss C, Behley J. Deep compression for dense point cloud maps. *IEEE Robotics Automation Lett* (2021) 6(2):2060–7. doi:10.1109/lra.2021.3059633

2. Qi J, Hu W, Guo Z. Feature preserving and uniformity-controllable point cloud simplification on graph. In: 2019 IEEE International Conference on Multimedia and Expo (ICME); 2019 July 08–12; Shanghai, China (2019). p. 284–9.

3. Wang D, Xu K, Quan Y. Structure-aware subsampling of tree point clouds. *IEEE Geosci Remote Sensing Lett* (2022) 19:1–5. doi:10.1109/lgrs.2021.3124139

4. Maglo A, Courbet C, Alliez P, Hudelot C. Progressive compression of manifold polygon meshes. *Comput and Graphics* (2012) 36(5):349–59. doi:10.1016/j.cag.2012.03.023

5. Suchde P, Jacquemin T, Davydov O. Point cloud generation for meshfree methods: an overview. *Arch Comput Methods Eng* (2022) 30(2):889–915. doi:10.1007/s11831-022-09820-w

6. Lee KH, Woo H, Suk T. Point data reduction using 3D grids. *The Int J Adv Manufacturing Technology* (2001) 18:201–10. doi:10.1007/s001700170075

7. Alexa M, Behr J, Cohen-Or D, Fleishman S, Levin D, Silva CT. Computing and rendering point set surfaces. *IEEE Trans Vis Comput Graphics* (2003) 9(1):3–15. doi:10.1109/tvcg.2003.1175093

8. Song H, Feng H-Y. A progressive point cloud simplification algorithm with preserved sharp edge data. *The Int J Adv Manufacturing Technology* (2009) 45:583–92. doi:10.1007/s00170-009-1980-4

9. Yuan XC, Wu LS, Chen HW. Feature preserving point cloud simplification. *Opt Precision Eng* (2015) 23(9):2666–76. doi:10.3788/ope.20152309.2666

10. Shi B, Liang J, Liu Q. Adaptive simplification of point cloud using k-means clustering. *Computer-Aided Des* (2011) 43(8):910–22. doi:10.1016/j.cad.2011.04.001

11. Yang Y, Li M, Ma X. A point cloud simplification method based on modified fuzzy C-means clustering algorithm with feature information reserved. *Math Probl Eng* (2020) 2020:1–13. doi:10.1155/2020/5713137

12. Nezhadarya E, Taghavi E, Razani R, Liu B, Luo J. Adaptive hierarchical down-sampling for point cloud classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2020 June 13–19; Seattle, WA, USA (2020). p. 12956–64.

13. Qi J, Hu W, Guo Z. Feature preserving and uniformity-controllable point cloud simplification on graph. In: Proc. IEEE Int. Conf. Multimedia Expo (ICME); Shanghai; 2019 July 8–12 (2019). p. 284–9.

14. Luo C, Ge X, Wang Y. Uniformization and density adaptation for point cloud data via graph Laplacian. *Comput Graph Forum* (2018) 37(1):325–37. doi:10.1111/cgf.13293

15. Zeng J, Cheung G, Ng M, Pang J, Cheng Y. 3D point cloud denoising using graph Laplacian regularization of a low dimensional manifold model. *IEEE Trans Image Process* (2019) 29:3474–89. doi:10.1109/tip.2019.2961429

16. Yang J, Zhang Q, Ni B, Li L, Liu J, Zhou M, et al. Modeling point clouds with self-attention and Gumbel subset sampling. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR); Long Beach, CA; 2019 June 15–20 (2019). p. 3323–32.

17. Nezhadarya E, Taghavi E, Razani R, Liu B, Luo J. Adaptive hierarchical down-sampling for point cloud classification. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR); Seattle, WA; 2020 June 13–19 (2020). p. 12956–64.

18. Yan X, Zheng C, Li Z, Wang S, Cui S. PointASNL: robust point clouds processing using nonlocal neural networks with adaptive sampling. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR); Seattle, WA; 2020 June 13–19 (2020). p. 5589–98.

19. Wen C, Yu B, Tao D. Learnable skeleton-aware 3D point cloud sampling. In: Proc. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); Vancouver, BC; 2023 June 17–24 (2023). p. 17671–81.

20. Lv C, Lin W, Zhao B. Approximate intrinsic voxel structure for point cloud simplification. *IEEE Trans Image Process* (2021) 30:7241–55. doi:10.1109/tip.2021.3104174

21. Yang M-S, Benjamin JBM. Feature-weighted possibilistic c-means clustering with a feature-reduction framework. *IEEE Trans Fuzzy Syst* (2020) 29(5):1093–106. doi:10.1109/tfuzz.2020.2968879

22. Zhang S, Li X, Zong M, Zhu X, Wang R. Efficient kNN classification with different numbers of nearest neighbors. *IEEE Trans Neural networks Learn Syst* (2017) 29(5):1774–85. doi:10.1109/tnnls.2017.2673241

23. Ringnér M. What is principal component analysis?. *Nat Biotechnol* (2008) 26(3):303–4. doi:10.1038/nbt0308-303

24. Fei N, Gao Y, Lu Z, Xiang T. Z-score normalization, hubness, and few-shot learning. In: Proceedings of the IEEE/CVF International Conference

on Computer Vision; 2021 October 10–17; Montreal, QC, Canada (2021). p. 142–51.

25. Xu K, Pedrycz W, Li Z. Augmentation of the reconstruction performance of Fuzzy C-Means with an optimized fuzzification factor vector. *Knowledge-Based Syst* (2021) 222:106951–61. doi:10.1016/j.knosys.2021.106951

26. Zhu X, Sun J, He Z, Jiang J, Wang Z. Staleness-reduction mini-batch K-Means. *IEEE Trans Neural Networks Learn Syst* (2023) 1:1–13. doi:10.1109/tnnls.2023.3279122

27. Guan S, Li G, Xie X, Wang Z. Bi-direction ICP: fast registration method of point clouds. In: 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA); 2017 May 08–12; Nagoya, Japan (2017). p. 129–32.

28. Wang S, Hu Q, Xiao D, He L, Liu R, Xiang B, et al. A new point cloud simplification method with feature and integrity preservation by partition strategy. *Measurement* (2022) 197:111173–89. doi:10.1016/j.measurement.2022.111173

29. Di Angelo L, Di Stefano P, Giaccari L. A new mesh-growing algorithm for fast surface reconstruction. *Computer-Aided Des* (2011) 43(6):639–50. doi:10.1016/j.cad.2011.02.012

30. Song H, Feng H-Y. A global clustering approach to point cloud simplification with a specified data reduction ratio. *Computer-Aided Des* (2008) 40(3):281–92. doi:10.1016/j.cad.2007.10.013