Comparison of Cloud-to-Cloud Distance Calculation Methods - Is the Most Complex Always the Most Suitable?

Vitali Diaz, Peter van Oosterom, Martijn Meijers, Edward Verbree, Nauman Ahmed, and Thijs van Lankveld

Abstract Cloud-to-cloud (C2C) distance calculations are frequently performed as an initial stage in change detection and spatiotemporal analysis with point clouds. There are various methods for calculating C2C distance, also called inter-point distance, which refers to the distance between two corresponding point clouds captured at different epochs. These methods can be classified from simple to complex, with more steps and calculations required for the latter. Generally, it is assumed that a more complex method will result in a more precise calculation of inter-point distance, but this assumption is rarely evaluated. This paper compares eight commonly used methods for calculating the inter-point distance. The results indicate that the accuracy of distance calculations depends on the chosen method and a characteristic related to the point density, the intra-point distance, which refers to the distance between points within the same point cloud. The results are helpful for applications that analyze spatiotemporal point clouds for change detection. The findings will be helpful in future applications, including analyzing spatiotemporal point clouds for change detection.

Keywords Cloud-to-cloud distance calculation · Change detection · Spatiotemporal analysis

This article was selected based on the results of a double-blind review of an extended abstract

V. Diaz (✉) · P. van Oosterom · M. Meijers · E. Verbree
GIS Technology, Faculty of Architecture and the Built Environment, Delft University of Technology, Delft 2628 BL, The Netherlands
e-mail: v.diazmercado@tudelft.nl; vitalidime@gmail.com

P. van Oosterom
e-mail: P.J.M.vanOosterom@tudelft.nl

N. Ahmed · T. van Lankveld
Netherlands eScience Center, Amsterdam 1098 XH, The Netherlands

T. H. Kolbe et al. (eds.), *Recent Advances in 3D Geoinformation Science*, Lecture Notes in Geoinformation and Cartography, https://doi.org/10.1007/978-3-031-43699-4_20

1 Introduction

Accurate calculation of the inter-point distance, which refers to the distance between two pairs of point clouds each taken at distinct epochs, is important for different applications involving point clouds, such as change detection (Stilla and Xu 2023) and spatiotemporal analysis (Anders et al. 2021). The selection of the method for calculating inter-point distance depends on several factors, such as the nature and size of the clouds, presence of noise or outliers, distance definition (e.g., Euclidean, geodesic, or Hausdorff distance), method's computational efficiency, method's time cost and memory usage, and availability of the method's implementation.

There are currently various tools with diverse implemented methods for calculating inter-point distance (e.g., CloudCompare (GPL software. 2023)). These implementations often allow the user to specify the distance type, which is usually Euclidean, and how the data is navigated for the subsample of points (including the querying of a specific number of neighbor points). In practice, this enables the selection of any implemented method. In general, it is assumed that a more complex method will result in a more accurate inter-point distance calculation, but this assumption is not always verified.

In this research, we compare eight methods for calculating inter-point distance following a controlled displacement test to analyze the accuracy of distance computations. The association of the inter-point distance with the point density is also investigated through the analysis of the intra-point distance, i.e., the distance between individual points within the same point cloud.

2 Methods and Data

Two concepts related to point distance are discussed in this paper, which are defined for a better understanding of the methodology. On the one hand, intra-distance refers to the distance between points within the same point cloud. This calculation is carried out by determining the median of the Euclidean distance from a specific point to its k nearest neighbors, providing a local measure related to the point density. This process is applied individually to each single point within the same cloud, offering valuable insight into the internal relationship of the data points. On the other hand, inter-distance, often called C2C (Cloud-to-Cloud) distance, measures the spatial separation between two corresponding point clouds taken at different epochs (time steps). These clouds are commonly identified as the reference cloud and the compared cloud. Inter-distance calculation is performed individually for each point of the compared cloud, enabling an assessment of the relationship between these two clouds. Various methods exist for calculating inter-distance, each with distinct assumptions about distance calculation. Inter-distance calculation yields one value for each point in the compared cloud, providing a comprehensive understanding of the spatial dissimilarities/similarities between the two point clouds.

Controlled displacement test

In this methodology, the process unfolds as follows. (1) Firstly, a specific point cloud is designated as the “reference cloud.” (2) Subsequently, the intra-distance is calculated for every point within the reference cloud, as well as the average intra-distance. (3) To explore a range of scenarios, artificial displacements are proposed based on the average intra-distance (as detailed below). (4) These proposed displacements are systematically applied to all points within the reference cloud, creating a “compared cloud” for each displacement scenario. (5) Moving forward, the calculation of inter-point distance between the compared and the reference cloud takes place. Eight different methods were tested (as outlined below). (6) Finally, each method is evaluated to determine its accuracy in capturing the applied artificial displacement.

The following methods were applied for the inter-distance calculation: (1) The nearest neighbor, (2) Least squares plane, (3) Linear interpolation, (4) Quadratic (height function), (5) 2.5D triangulation, (6) Natural Neighbor Interpolation (NNI), (7) Inverse Distance Weight (IDW), and (8) Multiscale Model to Model Cloud Comparison (M3C2). Most of the methods are available in CloudCompare, but we implemented them in Matlab for batch processing and further investigation.

The batch processing of inter-point distance involves the following steps, which are applied to each point in the compared cloud. (1) For every point in the compared cloud, the k nearest neighbors are retrieved from the reference cloud, which are identified as “selected points”. (2) Subsequently, the inter-distance method is applied to these selected points, resulting in a distance vector (dx, dy, dz). It is important to note that for Natural Neighbor Interpolation and Inverse Distance Weight, only the dz component can be calculated.

The deviation between the applied and calculated displacements was calculated to assess the ability to capture the correct artificial displacement for each method. Deviation was determined as the absolute difference between the applied and calculated displacements divided by the applied displacement, and it is expressed as a percentage. Six intervals were considered for assessing the deviation: 0 to 10, 10 to 20, 20 to 30, 30 to 40, 40 to 50, and greater than 50%. We focused on the dz component to evaluate the inter-distance methods because it is the value that all methods can return.

The artificial displacements were carefully chosen to analyze three specific scenarios:

(a) When the displacements are smaller than the average intra-point distance.
(b) When the displacements are equal to the average intra-point distance.
(c) When the displacements are greater than the average intra-point distance.

For both the calculation of intra-distance and inter-distance, the eight nearest neighbors were used, i.e., $k = 8$. This number of neighbors allows the application of all the methods. For example, in the Quadratic (height function) case, at least six points are required. Using a higher number of points is not advisable due to the detrimental impact on points selection times. The methodology was applied in three databases: (1) bunny, (2) lake, and (3) coast scan. The first two are available with the LAStools (Isenbur 2023), and the last from Vos et al. (2022).

3 Results and Discussion

In the case of the object (bunny) (Fig. 1), the methods that best capture the applied vertical displacement (dz) are nearest neighbor, NNI, and IDW. However, when for horizontal displacements, the NNI method performs better.

Interestingly, the results exhibit sensitivity to the direction of displacement. When displacement is applied vertically (in the z-axis), the calculated distances align closely with the applied displacement, particularly for points near the top of the object. In contrast, for horizontal displacements, the points located on the sides show better accuracy in the calculated displacement.

In the case of the lake database (terrain with hills and trees) (Fig. 2), most methods effectively capture the vertical displacement on the terrain. Notably, the nearest neighbor, NNI, and IDW methods stand out for accurately representing displacement on trees. However, regarding horizontal displacement, most methods fail to provide accurate results, with only the nearest neighbor and NNI methods presented notable performance. Specifically, in horizontal offsets, the results indicate that the points close to the sides of objects (trees) capture the displacement more accurately. For terrains, only the nearest neighbor method appears to yield improved results.

On the beach, most methods exhibit similar performance in capturing vertical displacement (dz). This is primarily because, for most points in the compared cloud, the neighbors from the reference cloud effectively represent the same section of the beach. However, when a horizontal offset is applied, most methods fall short in accuracy, and only the nearest neighbor method performs relatively better.

4 Conclusion and Future Work

This paper has presented the results of comparing eight commonly used inter-point distance calculation methods in point cloud analysis. The findings show that the accuracy of the inter-point distance calculation depends on the intra-point distance. Specifically, it has been observed that better accuracy is achieved when the inter-point distance is smaller than the intra-point distance.

Furthermore, the results of this study challenge the assumption that a method with more steps always yields superior performance. The comparative analysis has revealed that a more complex method is not necessarily the most suitable for inter-point distance calculation. These insights are invaluable for future applications involving the analysis of spatiotemporal point clouds for change detection. Researchers and practitioners can leverage these findings to make informed decisions when selecting suitable methods tailored to their analysis requirements.

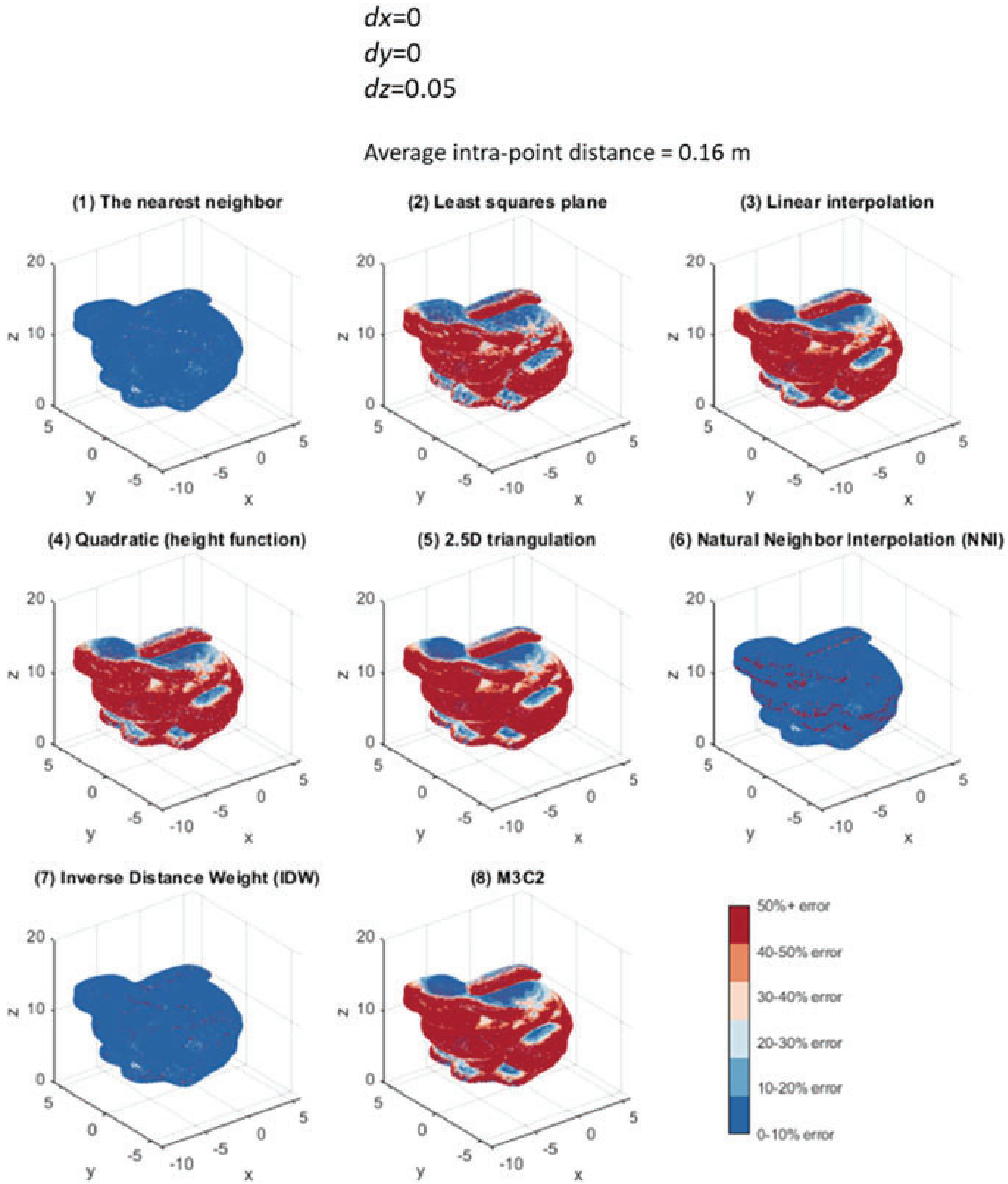

Fig. 1 Results for the bunny database

In our future work, we aim to explore the performance of these methods with larger and more complex point clouds while investigating the impact of various parameters, including point density, sampling techniques, and noise levels. Also, we will expand the scope of our study to include other transformations (e.g., rotation), beyond the artificial displacement (dx, dy, dz).

Finally, we also plan to analyze the AHN (*Actueel Hoogtebestand Nederland*, in Dutch) database, which covers lidar data for the entire Netherlands. We will select the most suitable method and implement it efficiently within a Database Management System (DBMS) using Space Filling Curve (SFC) techniques, particularly for the

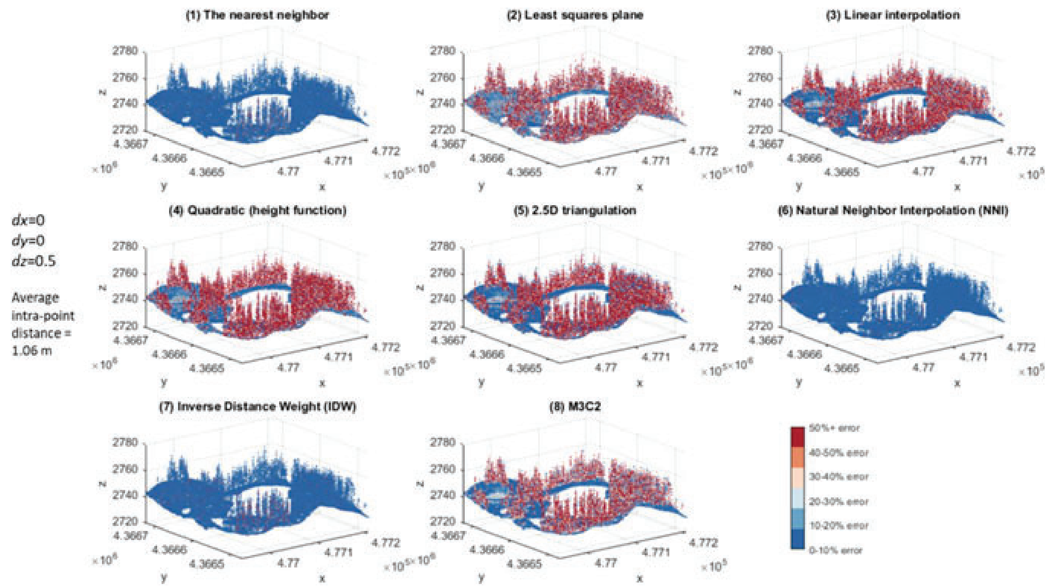

Fig. 2 Results for the lake database

comprehensive change detection analysis of the AHN-2, -3, and -4 datasets. Outputs project can be consulted at nd-pc.org and www.gdmc.nl/publications.

References

Anders K, Winiwarter L, Mara H, Lindenbergh R, Vos SE, Höfle B (2021) Fully automatic spatiotemporal segmentation of 3D LiDAR time series for the extraction of natural surface changes. ISPRS J Photogramm Remote Sens 173:297–308. https://doi.org/10.1016/j.isprsjprs.2021.01.015

GPL software (2023) CloudCompare (version 2.13.alpha). Retrieved from http://www.cloudcompare.org/. Accessed on 22 May 2023

Isenbur M (2023) LAStools–efficient LiDAR processing software. Retrieved from http://rapidlasso.com/LAStools. Accessed on 22 May 2023

Stilla U, Xu Y (2023) Change detection of urban objects using 3D point clouds: a review. ISPRS J Photogramm Remote Sens 197:228–255. https://doi.org/10.1016/j.isprsjprs.2023.01.010

Vos S, Anders K, Kuschnerus M, Lindenbergh R, Höfle B, Aarninkhof S, de Vries S (2022) A high-resolution 4D terrestrial laser scan dataset of the Kijkduin beach-dune system, The Netherlands. Sci Data 9(191):1–11. https://doi.org/10.1038/s41597-022-01291-9