https://doi.org/10.31449/inf.v48i22.6888 Informatica **48** (2024) 113–128 **113**

Natural Neighborhood Decomposable Point Algorithm for Feature Extraction in Intelligent Building Point Clouds

Zheng Gan[1*], Xiongya Shen[2], Xiang Liu[3], Ying Xie[4]
[1]Sichuan Zhongtai Jiangongye Construction Engineering Co., Ltd. Chengdu Sichuan 610000, China
[2]Sichuan Dingsheng Project Management Co., Ltd. Chengdu Sichuan 610000, China
[3]Chengdu Cangmang Construction Technology Co., Ltd. Chengdu Sichuan 610000, China
[4]Chongqing Information and Communication Research Institute Chongqing 400065, China
E-mail: 13890184644@163.com, sxyy667711@163.com, 13880476109@163.com, 18375715672@163.com
[*]Corresponding author

Keywords: intelligent buildings, natural neighborhood, feature extraction, 3d point cloud technology, decomposable graph filtering

Received: August 12, 2024

Intelligent buildings, emerging as a fusion of modern information technology and architectural structures, aim to provide intelligent, comfortable, and efficient building environments. However, the information processing and decision-making processes within intelligent buildings face challenges in dealing with large-scale, complex data for feature extraction. This research introduces a natural neighborhood decomposable point algorithm, effectively extracting crucial features within intelligent buildings. This extraction supports the decision-making and control processes for intelligence. The construction of the natural neighborhood of data points and the subsequent use of neighborhood information to extract key features enables the algorithm to maintain the interpretability of data while simultaneously enhancing the efficiency and accuracy of feature extraction. Experimental results demonstrated that employing this algorithm for registration reduced registration errors by 0.068 mm and 0.021 mm after the first iteration. The accuracy of this algorithm was 94.32%, the recall rate was 91.76%, the F1 value was 93.45%, and the processing time was only 24.37 seconds, which was obviously better than other comparison algorithms. This outcome validates the algorithm's efficiency in enhancing registration algorithms, maintaining both accuracy and interpretability while extracting data in intelligent building contexts. This algorithm can effectively analyze both local and global features in point cloud data, contributes to enhancing the energy efficiency, comfort, automation, and intelligence management levels within intelligent buildings. It propels innovation and development in intelligent buildings, improves security, and promotes sustainable development.

Povzetek: V prispevku je predstavljen algoritem za ekstrakcijo značilnosti v inteligentnih stavbah, ki temelji na naravni soseski in dekompoziciji točk. Algoritem učinkovito analizira lokalne in globalne značilnosti oblakov točk, kar prispeva k izboljšanju energetske učinkovitosti, udobja ter ravni avtomatizacije in inteligentnega upravljanja v inteligentnih stavbah.

1 Introduction

The rapid development of intelligent buildings enables structures to autonomously perceive and respond to environmental changes, delivering more comfortable and efficient user experiences [1]. However, data processing and decision-making within intelligent buildings encounter numerous challenges. Firstly, the generated data within intelligent buildings is massive, multi-source, and heterogeneous, necessitating effective feature extraction techniques to derive crucial insights. Secondly, data features in intelligent buildings are often nonlinear and high-dimensional, limiting the efficacy of traditional feature extraction methods [2-3]. Conventional methods rely on statistical and mathematical models such as principal component analysis and linear discriminant analysis. However, these methods often lack flexibility and precision when handling nonlinear and high-dimensional data. In recent years, with the advancement of deep learning and artificial intelligence, neural network-based feature extraction methods have achieved significant success. Nevertheless, due to the sparsity and heterogeneity typically found in intelligent building data, traditional neural network models face challenges in processing such data [4-5]. Hence, this research aims to propose an effective feature extraction algorithm tailored for processing and decision-making in intelligent building data. The study introduces a natural neighborhood decomposable point (NNDP) algorithm, leveraging natural neighborhood information among data points to extract crucial features. This algorithm preserves data interpretability while effectively extracting feature information from intelligent building data, supporting intelligent decision-making and control. The research contribution is that through the natural neighborhood method, the algorithm can more effectively identify the key feature points in the point cloud and extract them. Through accurate feature extraction, more accurate spatial information and data support can be provided for intelligent buildings. Through comprehensive scanning and extraction of pertinent features from the building surface, potential safety hazards and failure points can be identified in a timely manner, thereby enabling targeted repair and maintenance procedures to be undertaken. This not only improves the life and safety of the building, but also reduces maintenance costs and resource consumption, enabling more sustainable building management. The article comprises five part: the first part reviews literature, discussing the advanced in the NNDP based on natural neighborhood principles and international research on intelligent buildings. The second part presents the intelligent building feature extraction using the decomposable point feature extraction algorithm based on natural neighborhoods. The third part validates the effectiveness and performance of this algorithm through experiments. The fourth part is discussion, comparing and analyzing the research extraction algorithm with other methods. The fifth part provides a summary of the research outcomes.

2 Related works

In recent years, the feature extraction algorithm of decomposable points in natural neighborhood has been widely used. In order to find an accurate, stable and effective registration algorithm for augmented reality, Yang et al. combined the directional FAST and rotating BRIEF feature matching methods, and combined the random sample consistency (RANSAC) algorithm to obtain the homography matrix. They used the Kanade-Lucas-Tomasi (KLT) tracking algorithm to track the markers. The research results showed that the algorithm effectively reduced the accumulation of errors, thereby improving the stability and accuracy of registration [6]. To improve the accurate measurement level of pose of complex curved parts, Zhang et al. proposed a point cloud registration algorithm based on fast extraction of local feature points by combining the advantages of vector feature and fast point feature histogram. The experimental results showed that this improved algorithm significantly improved the efficiency and accuracy, and provides effective support for the precise positioning of products [7]. To achieve resource savings, cost reduction, and environmental impact reduction, Wang conducted an in-depth analysis of intelligent construction systems for prefabricated buildings using iot technology. The experiment results indicated that the system ensured operational efficiency without sacrificing safety. The quality-of-service value was 4.52, the synergy value was 5.26, and the objective function value was 4.81. The excellent performance of these indicators proved the potential and benefits of the system in practical application [8]. To solve the problem that density peak clustering ignores neighborhood information when calculating local density, Chen proposed a domain density peak clustering algorithm based on natural neighborhood (NDDC). Studies have shown that this method exhibits excellent accuracy and robustness [9].

In recent years, with the rapid development of advanced information technology such as artificial intelligence, building intelligence has gradually become the focus of attention. As the population continues to grow and the frequency of electrical appliances increases, the energy consumption of residential buildings is rising rapidly. To this end, Kaur et al. developed a deep learn-based bidirectional long short-term memory model designed to predict the power demand of individual appliances. Genetic algorithm was used to optimize the hyperparameters of the deep learning model. The research showed that this hybrid method significantly improved the accuracy of prediction by reducing the error rate of the data set [10]. At the same time, the intelligent technology of prefabricated buildings has also made remarkable progress. Yu et al. deeply analyzed the constraints between prefabricated building projects, and proposed the application of radial basis function fuzzy logic neural network algorithm to the optimization of building resource scheduling. The research results showed that this method could effectively solve the resource scheduling problem in prefabricated building projects and improve the resource utilization efficiency [11]. To promote the development of energy-efficient smart buildings, Wang et al. proposed an automatic intelligent LED light control system based on solar energy and applied it to actual buildings. The outcomes indicated that compared with the traditional AC power grid system, the system could achieve up to 78% energy saving [12]. These innovative technologies and methods are of great significance for promoting the development of intelligent buildings and energy conservation. The literature review is summarized in Table 1.

Table 1: Summary of literature review

Reference	Research method	Accuracy (%)	Calculation time (s)	Application field	ContributionsImprove registration stability and accuracy
[6]	Directional FAST and rotating BRIEF feature matching + RANSAC + KLT tracking algorithm	85	22	Augmented reality registration	Improve efficiency and accuracy
[7]	Point cloud registration algorithm for normal vector feature and fast point feature histogram	88	35	Pose measurement of complex curved parts	Reduced run time and false alarm rate
[8]	Intelligent construction system of prefabricated building based on Internet of Things technology	75	29	Intelligent construction of prefabricated buildings	Save resources, reduce costs and reduce environmental impact
[9]	Peak density clustering algorithm based on natural neighborhood	80	38	Cluster analysis	Reduce data set error rates
[10]	Bidirectional long short-term memory model based on deep learning	82	30	Electricity demand forecast for home appliances	Improve the efficiency of resource utilization
[11]	Radial basis function fuzzy logic neural network algorithm	84	25	Building resource scheduling optimization	Achieve energy savings of up to 78%
[12]	Automatic intelligent LED light control system based on solar energy	77	83	Energy-saving intelligent building	Significantly improve accuracy and computational efficiency
NNDP	Feature extraction algorithm of decomposable points based on natural neighborhood	94	24	Intelligent building point cloud data processing	ContributionsImprove registration stability and accuracy

In summary, the NNDP can effectively extract valuable information from point cloud data, providing support for the design, construction, and operation of smart buildings. However, current research still faces certain limitations, such as the complexity and real-time issues of the algorithm that need further resolution. The study aims to optimize the algorithm to enhance its efficiency and accuracy in practical applications.

3 Optimization for natural neighborhood point cloud feature extraction algorithm for smart buildings

This study proposes a decomposable lightweight method based on natural neighborhood to optimize point cloud data in a large volume scenario. Firstly, combining the concept of natural neighborhood point set and the process of graph signal feature extraction, a decomposable feature extraction algorithm is created. Then, based on this, a decomposed filtering method is implemented, and the original data is effectively lightweight. Finally, based on the ADMM consensus principle, a distributed optimization method is proposed for the balance of corner and plane features to improve the adaptability of applications and the ability of processing large-scale point cloud data.

2.1 Decomposable point cloud feature extraction based on neighborhood information

Smart buildings refer to a form of architecture that utilizes advanced information technology and intelligent systems to achieve intelligent management, control, and optimization of buildings. Leveraging technologies such as sensors, network communication, automation control, and artificial intelligence, smart buildings monitor, analyze, and regulate information related to internal environments, energy usage, safety management, and more in real-time. This aims to enhance the comfort, energy efficiency, safety, and sustainability of the building. The rapid development of smart buildings relies on the extraction of key features to support intelligent decision-making and control processes [13]. In the design stage of intelligent buildings, 3D point cloud technology can be used to obtain accurate 3D models of buildings, provide designers with detailed spatial information, and help optimize building structure and system layout, as shown in Fig. 1.

Decomposable point cloud feature extraction based on neighborhood information is an effective method for processing point clouds. It can extract features from point clouds and decompose them into multiple components. Point cloud data is a collection of

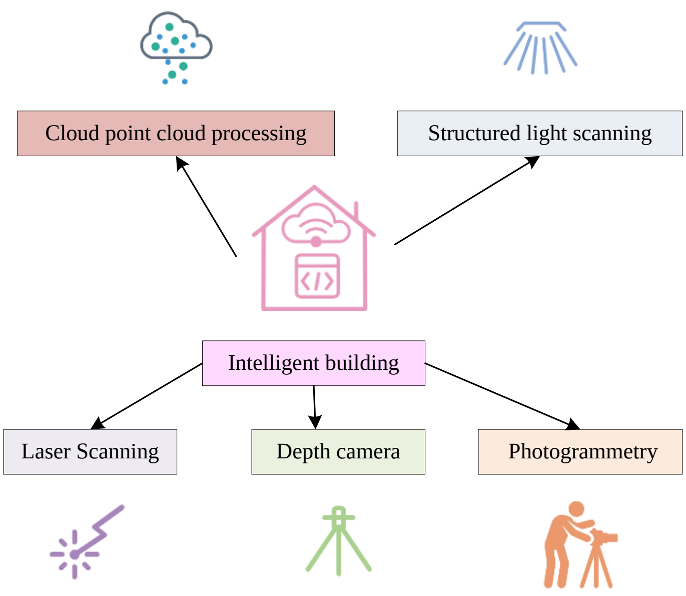


Figure 1: Three-dimensional point cloud technology for intelligent buildings

discrete spatial points obtained by laser scanners or other sensors, representing the three-dimensional geometry and structure of buildings or their parts [14-15]. In this method, neighborhood information refers to the nearest neighbors of each point, i.e., the set of points closest to the point in space. By establishing the neighborhood relationships for each point in the point cloud data, local structures and correlations between points can be obtained. The natural neighborhood refers to the closest points in the point cloud data, where the natural neighborhood of a point is the closest other points in space. In this method, the first step is to construct the natural neighborhood for each point. This can be done by calculating the distances between points and selecting the closest group of neighboring points for each point as its natural neighborhood. Subsequently, various feature extraction methods, such as geometric features (e.g., normals, curvature), local descriptors (e.g., spherical harmonics transform, rotation-invariant features), and statistical features (e.g., mean, variance), can be applied to extract separable features in each point and its natural neighborhood. These features are then aggregated, utilizing methods such as weighted averaging or maximum pooling, to obtain an overall separable feature representation. Finally, the separated features can be applied to tasks such as point cloud segmentation, object detection, scene classification, etc., considering point cloud data as a set, as shown in Equation (1).

$$x = (x_1, ..., x_i, ..., x_j, ..., x_N), 0 < i < j < N \quad (1)$$

In Equation (1), point x_j is the neighbor of point x_i in point set x. If any two points in the set satisfy this relationship, they can be mutually referred to as natural neighbors. Building upon the concept of natural neighbors, the following definitions can be established: fully natural neighborhood point set (FNNPS): For a given point set, if each point in the set is part of its natural neighborhood, the point set can be termed as an FNNPS. Natural neighborhood point set (NNPS): For a given point set, if each point in the set is at least adjacent to one other point, and these adjacent points are only in the natural neighborhood of that point, then the set can be referred to as an NNPS. Expanded natural neighborhood point set (ENNPS): For a given point set, if each point in the set is at least adjacent to one other point, and these adjacent points are in the natural neighborhood of that point or in the natural neighborhood of the natural neighbors of that point, then the set can be termed as ENNPS. For any large-scale point cloud data X, if it can be divided into P fully natural neighborhood point sets, the relationships between the point sets are represented in Equation (2).

$$\begin{aligned} AX &= diag(A_1, ..., A_p, ..., A_p)[X_1, ..., X_p, ..., X_p]^T \\ &= [A_1 X_1, ..., A_p X_p, ..., A_p X_p] \end{aligned} \quad (2)$$

In Equation (2), X_p represents the p th FNNPS obtained by partitioning the point cloud data X, A_p represents the transition matrix, and $diag(\cdot)$ represents the function for constructing a diagonal matrix. The collection of point clouds obtained by partitioning X using natural neighborhood point sets is represented in Equation (3).

$$S_e = (X_{e1}, ..., X_{ep}, ..., X_{ep}) \quad (3)$$

In Equation (3), X_{ep} represents the p th natural neighborhood point set. The relationship between fully natural neighborhood point sets and natural neighborhood point sets is illustrated in Fig. 2.

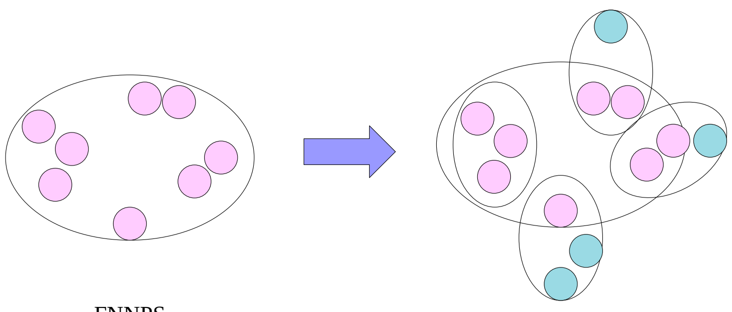


Figure 2: Completely natural neighborhood point set and natural neighborhood point set

Defining ENNPS as X_{up}, the graph filter is represented as shown in Equation (4).

$$f(X_{up}) = A_{up} X_{up} \quad (4)$$

In Equation (4), A_{up} represents the transition matrix on X_{up}. For the feature extraction algorithm for separated point clouds, the feature representation is given in Equation (5).

$$f(X) = \sum_{l=0}^{L-1} h_1 A^l X = \sum_{l=0}^{L-1} F_l, L > 1 \quad (5)$$

In Equation (5), $f(X)$ represents the feature, L represents any coefficient, A represents the transition matrix, h_1 represents the coefficient of the feature term, $F_l, L > 1$ represents the feature term of $f(X)$ with any coefficient greater than 1. Research focuses on using the graph transfer operator matrix as the transition matrix, as shown in Equation (6).

$$A = D^{-1} W \quad (6)$$

In Equation (6), A represents the transition matrix, D represents the degree matrix of the graph and W represents the adjacency matrix of the graph. Signal transfer operations are performed on the feature terms, as shown in Equation (7).

$$F_l = A \cdot F_{l-1}^{'} \tag{7}$$

In Equation (7), F_l represents the result of performing signal transfer operations on the feature term $F_{l-1}^{'}$. The algorithm flow for feature extension terms F_l is illustrated in Fig. 3.

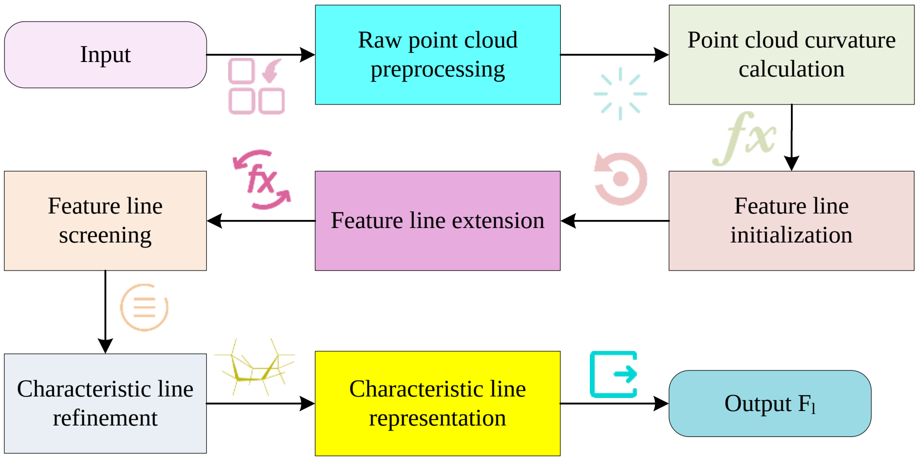


Figure 3: Feature extension item F_l algorithm flow

Due to environmental noise and other scanning errors, raw data often contains a large number of noise and anomalies. Therefore, before applying the NNDP algorithm, it is necessary to preprocess the data to improve the accuracy and efficiency of subsequent feature extraction. Use a statistical filter to remove outliers. To reduce computation, the point cloud is downsampled by voxel grid filtering or random sampling, preserving as much of the building's structural information as possible. Coordinate transformation of point cloud data to ensure that all data is in a unified coordinate system for subsequent processing. Once the pre-processing is complete, the subsequent crucial phase is the calculation of the distance between each point and its neighbors, in accordance with the spatial position of each point within the point cloud. This is followed by the determination of the natural neighborhood of each point through the application of a threshold value. Using the natural neighborhood information, a graph filter is applied to each point to extract key geometric features. Through decomposition and aggregation operations, local features are integrated into global feature descriptors to support subsequent point cloud segmentation, classification, and recognition tasks. In this way, NNDP algorithms can effectively extract useful information from complex 3D point cloud data to support intelligent management and maintenance of buildings.

2.2 Decomposable graph filtering based on decomposable point cloud feature extraction algorithm

Decomposable graph filtering is a method used for point cloud data processing. It is based on the decomposable point cloud feature extraction algorithm, which identifies and extracts decomposable parts in point cloud data to achieve filtering and noise removal. The fundamental idea of decomposable graph filtering is to represent point cloud data as a graph, where each point is a node, and the relationships between points are represented as edges of the graph. Utilizing the decomposable point cloud feature extraction algorithm, the point cloud data is decomposed into a set of feature vectors. These feature vectors can represent the local structure and geometric features of the point cloud data. In decomposable graph filtering, a decomposable graph of the point cloud data is first constructed. Then, using the feature extraction algorithm, the feature vectors for each node are calculated. Subsequently, by filtering and processing these feature vectors, noise reduction and smoothing operations on the point cloud data can be achieved. Finally, based on the processed feature vectors, the original point cloud data can be reconstructed, resulting in the filtered outcome [16-17]. Taking the example of a graph high-pass filter, the algorithmic workflow of decomposable graph filtering based on the decomposable point cloud feature extraction algorithm is illustrated in Fig. 4.

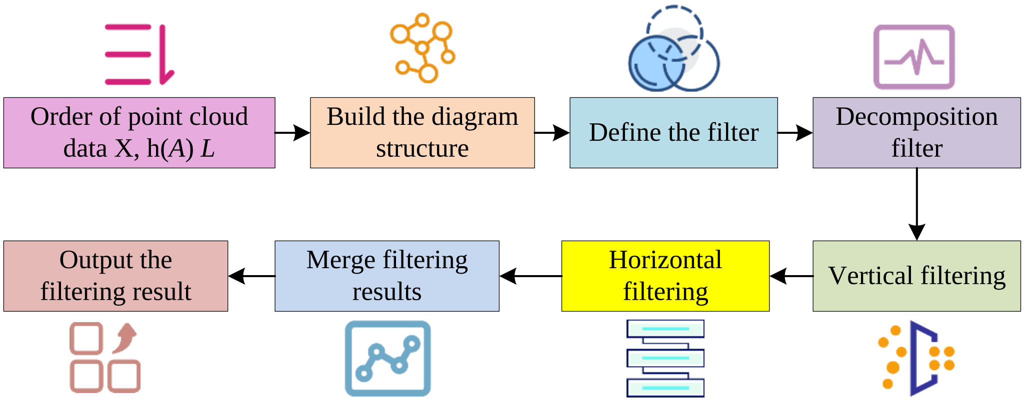
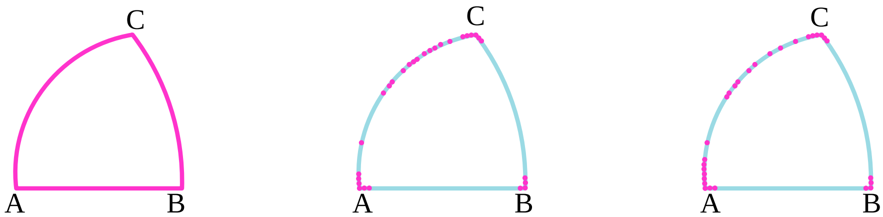


Figure 4: The algorithm flow of separable graph filtering

A second-order high-pass filter is a commonly used filter in image processing to enhance details and edge information. By applying high-pass filtering to an image, it emphasizes the high-frequency components such as edges and texture details. However, in certain situations, a second-order high-pass filter may respond strongly to noise and low-frequency components, leading to false detections and filtering out important details. On the other hand, Haar features define a specific set of local image patterns, offering a more precise description and capture of edge and texture features in the image. Haar features are an image-based feature description method that, compared to traditional second-order high-pass filters, exhibits higher sensitivity and expressiveness [18-19]. In contrast to second-order high-pass filters, Haar features have increased sensitivity and local expressiveness, allowing for more accurate representation and detection of features in the image. Therefore, research on the use of Haar features as a replacement for traditional second-order high-pass filters aims to improve the performance and accuracy of the algorithm. The patterns used in Haar features are typically composed of square and rectangular regions, where the pixel values in the rectangular region contribute to the intensity of the image features within that region. By sliding these patterns across the image, a set of feature values can be computed, representing features such as texture, edges, and color in different regions of the image. The Haar feature-sensitive second-order high-pass filter is depicted in Equation (8).

$$h_{H_2}(A) = 1 - \frac{1}{2}A - \frac{1}{2}A^2 \tag{8}$$

The lightweight results of the Haar feature-sensitive second-order high-pass filter and the Haar feature-sensitive first-order high-pass filter are shown in Fig. 5.



(a) Primitive point set (b) First-order filter (c) Second-order filter

Figure 5: Difference between a first-order filter and a second-order filter

The lightweight algorithm based on graph filters may face limitations in certain applications, making it difficult to meet the evolving demands in practical scenarios. To address this challenge, research has explored the use of distributed computing for algorithmic solutions. Distributed computing involves parallel computation by assigning computing tasks to multiple computing nodes. Leveraging the characteristics of distributed computing can enhance algorithmic computational efficiency and scalability to adapt to the changing demands of applications. The alternating direction method of multipliers (ADMM) is a commonly used algorithm for optimizing NNDP. It is employed to solve convex optimization problems with specific structures. The main idea behind ADMM is to transform the original problem into an equivalent optimization problem with additional penalty terms. By introducing multiplier variables to constrain the relationships between variables, distributed solving is achieved [20-23]. ADMM typically addresses problems of the form shown in Equation (9).

$$\begin{cases} \min_{x} f(w)+g(z) \\ s.t. Mw+Nz=C \end{cases} \tag{9}$$

In Equation (9), $f(w)$, $g(z)$ represent appropriate closed convex functions. It is observed that the above problems are separable, while the constraints are mutually coupled. ADMM employs alternating optimization to update the solution to the problem. The research introduces dual variables $\lambda > 0$ and penalty parameter $\rho > 0$, deriving the augmented Lagrangian equation as shown in Equation (10).

$$l_p(w,u,\lambda) = f(w)+g(z)+\lambda^T(Mx+Nz-C) + \frac{\rho}{2}\|Mx+Nz-C\|_2^2 \tag{10}$$

By combining linear and quadratic terms, the equivalent minimization form of the above equation is expressed in Equation (11).

$$l_p(w,u,\lambda) = f(w)+g(z)+\frac{\rho}{2}\left\|Mx+Nz-C+\frac{\lambda}{\rho}\right\|_2^2 \tag{11}$$

In each iteration, the specific update formulas are given by Equation (12).

$$\begin{cases} w^{k+1} = \arg\min_w f(w)+\frac{\rho}{2}\left\|Mw+Nz^k-C+\frac{\lambda^k}{\rho}\right\|_2^2 \\ z^{k+1} = \arg\min_z g(z)+\frac{\rho}{2}\left\|Mw^{k+1}+Nz-C+\frac{\lambda^k}{\rho}\right\|_2^2 \\ \lambda^{k+1} = \arg\min_\lambda \lambda^k+\rho(Mw^{k+1}+Nz^{k+1}-C) \end{cases} \tag{12}$$

In the ADMM algorithm, dynamic adjustment of the coefficients of the quadratic penalty terms is critical. This method uses the original primal and dual feasible residuals as criteria for determining whether the algorithm satisfies termination conditions. Improper adjustment of the quadratic penalty coefficients can lead to slow convergence or failure to achieve the desired results. The research dynamically adjusts the penalty coefficients to balance the convergence rates of the original primal and dual feasible solutions, ensuring consistent average reduction rates during runtime, as shown in Equation (13).

$$\rho^{k+1} = \begin{cases} \gamma_p \rho^k, \|r^k\|_2 > u\|s^k\|_2 \\ \frac{\rho^k}{\gamma_d}, \|s^k\|_2 > u\|r^k\|_2 \\ \rho^k, \text{other} \end{cases} \tag{13}$$

In equation (13), parameters γ_p , γ_d and u are all greater than 1. The mechanism for adjusting the quadratic penalty term parameters aims to maintain a consistent relative multiplier between original feasibility and original feasibility coefficients throughout the adjustment process. When the original feasibility exceeds a certain factor μ , the quadratic penalty factor is amplified, rapidly reducing the error in the original feasible solution. Conversely, by reducing the quadratic penalty coefficient, the dual feasible residual rapidly decreases. The main optimization function of ADMM is the augmented Lagrange function after the transformation of the original problem. In the framework of ADMM, the original optimization problem (such as equation (9)) is usually a convex optimization problem with special structure, which is separable, but the constraints are coupled with each other. To solve this problem, ADMM algorithm introduces multiplicator variables to constrain the relationship between variables, and transforms the original problem into an equivalent optimization problem with additional penalty terms. This equivalence problem is expressed by augmenting the Lagrange function (such as equation (10) and (11)), which contains the objective function of the original problem, the multipliers of the constraints, and an additional penalty term. In each iteration, the ADMM algorithm updates the solution of the problem by alternating optimization. The lightweight algorithm based on ADMM consensus is outlined in Fig. 6.

Regarding the selection of sampling points, the research employs a probabilistic approach based on the quantity of point cloud block samples. The point cloud is initially sampled into blocks with probabilities and then further sampled with certain probabilities. This method uses the sum of midpoint probabilities of point cloud data blocks as weights for sampling the current point cloud data block. Additionally, each midpoint of the point cloud data block is sampled probabilistically, controlling the sampling rate of point cloud data blocks while ensuring sampling quality. To sum up, 3D feature extraction is an important link in pattern recognition, which can extract key feature information from 3D data and provide a basis for subsequent recognition, classification, and other tasks. In the field of intelligent buildings, 3D feature extraction can be applied to the 3D modeling and structural analysis of buildings to help designers and engineers understand the structure and characteristics of buildings more accurately.

In the design and operation process of intelligent buildings, a lot of decision support is needed, including the optimization of building layout, the selection of equipment systems, and the strategy of energy management. These decisions need to be based on a large amount of 3D spatial information and building characteristics data, which can be obtained through 3D feature extraction. Point cloud registration is an important step in 3D space reconstruction, which can register the point cloud data obtained from different sources or at different times to a unified coordinate system so as to achieve a complete reconstruction of three-dimensional scenes or objects. In smart buildings, point cloud registration can be used for 3D scanning and reconstruction of buildings, helping to obtain an accurate 3D model of the building. This is important for structural analysis, maintenance management, and virtual reality applications of buildings. In the context of intelligent buildings, decision making involves how to optimize the design of the building and improve the performance of the building according to the results of three-dimensional feature extraction and the data registered by the point cloud. These decisions must consider a variety of factors, including technical feasibility, economic cost, user needs, etc., and make the best choice through scientific analysis and evaluation.

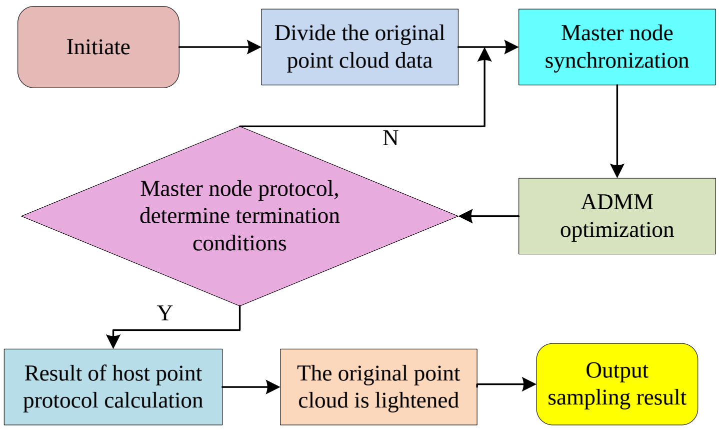


Figure 6: Lightweight algorithm flow based on ADMM consensus

4 Performance study of NNDP

The study involved conducting decomposable lightweight experiments on different-sized point clouds to validate the proposed algorithm. For the sake of experimental analysis, the research used regular-sized point cloud data as a control test and compared it to the NNDP algorithm.

4.1 Effect analysis of ADMM-optimized NNDP Algorithm

For each point or local region, a two- or three-dimensional surface is fitted and the Haar function is applied to this local surface. In this way, each point or region can obtain a Haar eigenvalue based on its local geometric properties. Three-dimensional point cloud data is projected into low-dimensional spaces by a dimensionality reduction technique, and then Haar features are applied to these low-dimensional spaces. For point cloud data, Haar features can be applied at different scales or resolutions. This can be achieved by building a pyramid representation of the point cloud, where each level represents a different scale of the point cloud data. Haar features can then be applied separately at each level. Using the concept of natural neighborhoods, each point considers other points in its neighborhood and computes some form of local statistic or feature. These features can be based on the distribution, density, orientation, or other geometric properties of the neighboring points. These local features can then be combined with Haar features to capture the local and global properties of multidimensional point clouds. To analyze the decomposable graph filtering effect of the feature extraction algorithm based on decomposable point cloud, the study set the total number of points in the polygonal point set to 400 and the sampling rate to 20%. In the original point set, 400 points are evenly distributed along each edge of the polygon. The sampling probabilities of points in the polygonal point set under different filters are shown in Fig. 7.

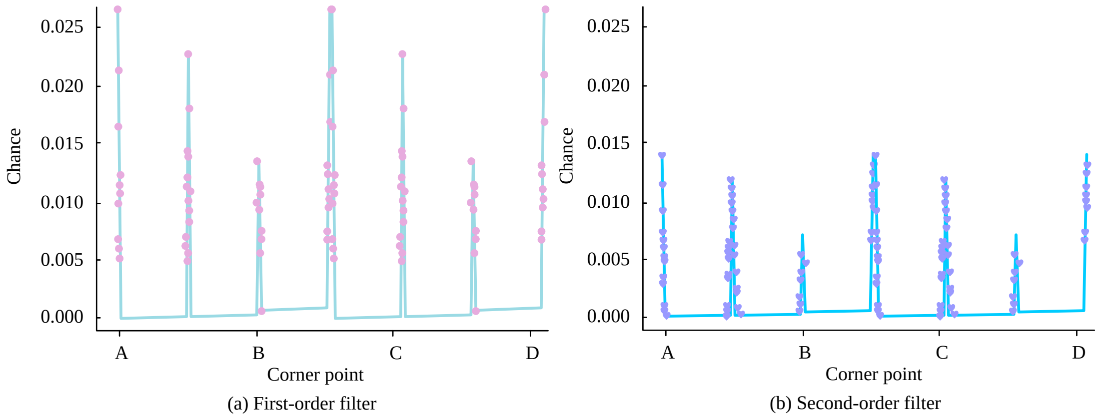


Figure 7: The sampling probability of polygonal points under different filters

Fig. 7 (a) represents a first-order high-pass filter sensitive to Haar features, while (b) represents a second-order high-pass filter sensitive to Haar features. The first-order high-pass filter sensitive to Haar features is less sensitive to local detail changes compared to the second-order high-pass filter sensitive to Haar features, resulting in lower sampling probabilities. Due to the lightweight nature of the Haar feature-sensitive filter, points in the corner regions are more likely to be considered as sampling points and retained, as they exhibit greater curvature changes relative to other points. In the decomposable point cloud feature extraction algorithm, point cloud data is decomposed into multiple triangular grids, and feature extraction is performed on each triangular grid. This algorithm effectively reduces computational complexity and improves feature extraction accuracy. To investigate the acceleration performance of the ADMM algorithm in optimizing the NNDP algorithm, the study conducted experiments under the same experimental parameter conditions. In this experiment, the study used dynamic coefficient adjustment and over-relaxation to further enhance performance. The over-relaxation parameter is set to 1.5. The study compared the performance of the accelerated method with the non-accelerated method on the PCEWP and Room datasets, and the convergence curves of these two methods are shown in Fig. 8.

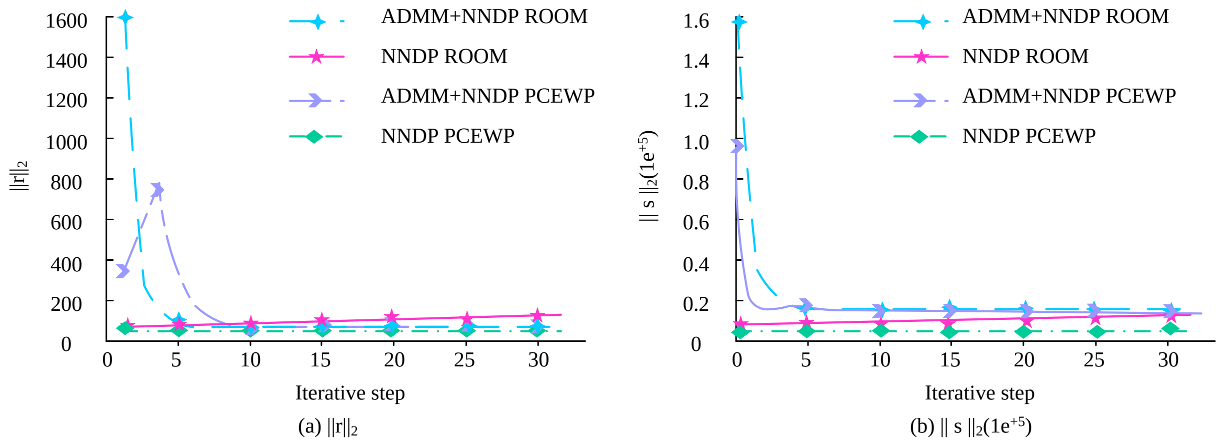


Figure 8: Convergence curves for both methods

In Fig. 8 (a), that the NNDP algorithm optimized with the ADMM algorithm significantly outperforms the unoptimized version in terms of convergence speed. Especially when dynamic coefficient adjustment and over-relaxation are used, the convergence speed is further improved. Fig. (b) shows that the accelerated method excels in reducing the average original error and average dual error. This demonstrates the effectiveness of the ADMM algorithm in optimizing the NNDP algorithm, and the introduction of dynamic coefficient adjustment and over-relaxation further enhances the algorithm's performance. To quantify the lightweight point cloud results, a quantitative evaluation is conducted using point cloud registration. Point clouds to be registered are sampled by 20% using uniform sampling, high-pass graph filtering, and distributed optimization methods, as shown in Fig. 9.

In Fig. 9, the uniformity of the point cloud is more pronounced in subfigure (a). This indicates that the algorithm based on graph filters achieves optimal performance as it focuses more on point cloud features. The algorithm proposed in the study, compared to uniform sampling, results in a smaller number of corresponding points but with a more favorable mean square error. Therefore, utilizing distributed optimization methods for lightweight point cloud registration leads to improved performance and results. To further demonstrate the effectiveness of NNDP, the study compares the registration errors between optimized and non-optimized NNDP, plotting histograms of bad point quantities for BPN models with noise-free and Gaussian noise point clouds of varying standard deviations, as shown in Fig. 10

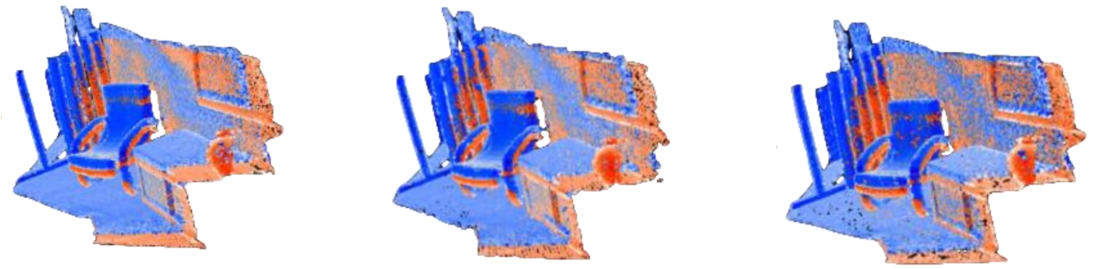
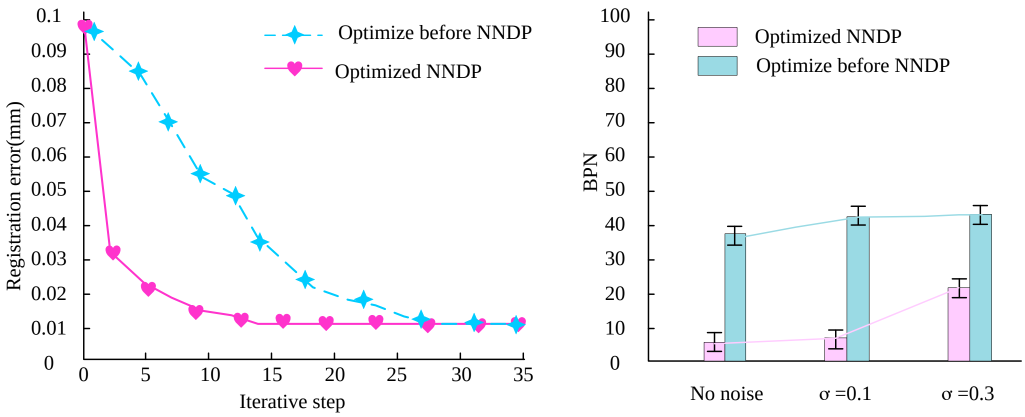

(a) The proposed algorithm (b) Uniform sampling (c) High-pass filtering

Figure 9: Lightweight point cloud registration results



(a) Registration error (b) Bad point distribution histogram

Figure 10: Histogram of registration errors and the number of bad points generated

In Fig. 10 (a), after registration using the NNDP algorithm, registration errors decrease by 0.068 mm and 0.021 mm after the first iteration, demonstrating the efficiency improvement of the NNDP algorithm for registration. In Fig. (b), with a noise standard deviation of 0.1, a noticeable increase in bad point quantity may be attributed to the beginning interference of this level of noise with the point cloud data. As the noise standard deviation further increases to 0.3, both methods exhibit a clear trend of increased bad point numbers. It indicates severe noise interference in the point cloud data, posing challenges for the methods in data processing.

4.2 Comparison of different algorithms and simulation analysis in smart building

To validate the performance of the NNDP algorithm, the study uses a single-machine simulation cluster to compare the proposed algorithm with nearest-point algorithm, Douglas-Rachford splitting algorithm, and Bregman projection algorithm, as shown in Table 2.

In Table 2, the NNDP algorithm achieves precision of 94.32%, recall of 91.76%, and an average F1 value of 93.45%, with a runtime of 24.37 seconds. In comparison, the nearest-point algorithm, Douglas-Rachford splitting algorithm, and Bregman projection algorithm exhibit lower precision, recall, and average F1 values along with longer runtimes, highlighting the superior performance of the NNDP algorithm. To further assess the NNDP algorithm against the nearest-point algorithm, precision, recall, and fitness are compared, as illustrated in Fig. 11.

Table 2: Parameter comparison of different algorithms

Algorithm	Precision(%)	Recall(%)	F1 value(%)	Total algorithm time (s)
ADMM	94.32	91.76	93.45	24.37
Proximal point algorithm	73.68	82.38	76.30	35.84
Douglas-Rachford splitting algorithm	86.11	77.25	76.25	25.16
Bregman projection algorithm	69.34	69.58	70.38	19.28

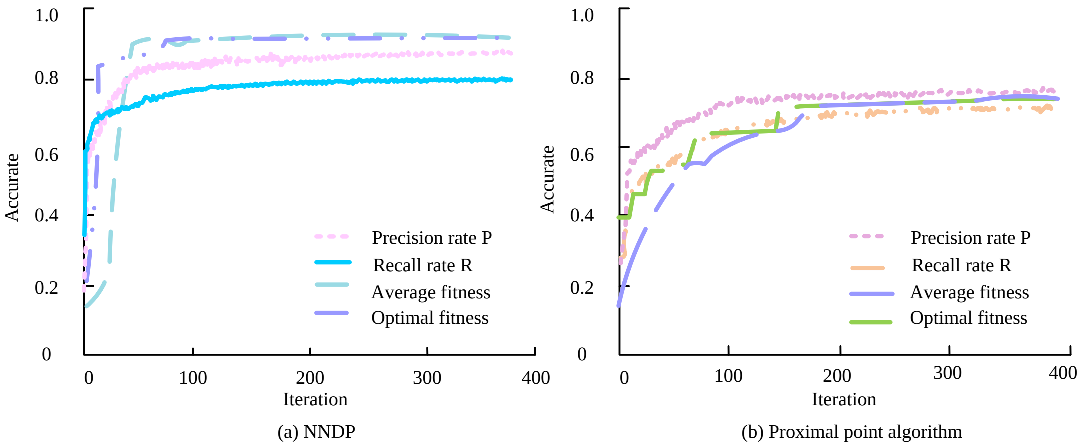


Figure 11: Comparison of different indexes of the two algorithms

Table 3: Statistical results of t test

Performance index	NNDP algorithm	Traditional algorithm	Mean difference	t-value	p-value
Accuracy (%)	94.32	88.56	5.76	7.85	<0.001
Calculation time (s)	24.37	55.43	-31.06	-9.23	<0.001

Fig. 11 (a) depicts the precision, recall, and fitness of the NNDP algorithm, demonstrating high precision and stable values above 0.9 after 100 iterations, with recall consistently above 0.8. In contrast, Fig. 11 (b) shows the precision and recall of the nearest-point algorithm hovering around 0.6 to 0.7, indicating that the NNDP algorithm outperforms the nearest-point algorithm in terms of accuracy. In order to prove the significant improvement of NNDP algorithm, the research conducted t test, as shown in Table 3.

In Table 3, the mean accuracy of the NNDP algorithm is 94.32%, whereas the mean accuracy of the conventional algorithm is 88.56%. The mean difference is 5.76%, thereby indicating that the accuracy of the NNDP algorithm has been markedly enhanced. The mean processing time for the NNDP algorithm is 24.37 seconds, whereas the mean processing time for the traditional algorithm is 55.43 seconds. The mean difference is -31.06 seconds, indicating that the NNDP algorithm has a significant improvement in calculation efficiency. The T-value of the T-test indicates the degree of difference between the sample mean and the population mean. A positive value indicates that the NNDP algorithm is superior to the traditional algorithm, whereas a negative value indicates that the NNDP algorithm improves performance. The P-values are all less than 0.001, which is considerably less than the significance level of 0.05. This indicates that the improvement of the NNDP algorithm is both significant and effective.

4.3 Noise sensitivity analysis

When processing points cloud data for intelligent buildings, typical noise levels vary depending on factors such as scanning equipment, environmental conditions, and scanning distance. In general, noise levels can manifest themselves as errors in the point cloud data, such as deviations from point positions, outliers, or anomalies. To evaluate the performance of NNDP algorithms under real-world noise conditions, experiments are being conducted that compare NNDP algorithms with other algorithms and run-on datasets with varying noise levels. The comparison algorithms include iterative closest point (ICP) algorithm and random sample consensus (RANSAC) algorithm. Data sets include small office building point cloud data (DataSet1), urban park environment point cloud data (DataSet2), industrial plant point cloud data (DataSet3). The final experimental results are shown in Table 4.

Table 4: Noise sensitivity analysis results

Noise level	Data set	NNDP	ICP	RANSAC	IndicatorsAccuracy (%),
Low	DataSet1	92.5	89.0	88.5	Calculation time (s)
		23.5	30.0	29.0	Accuracy (%),
Medium	DataSet2	91.0	86.5	85.0	Calculation time (s)
		24.0	32.5	31.0	Accuracy (%),
High	DataSet3	89.5	83.0	81.5	Calculation time (s)
		25.0	35.0	34.0s	IndicatorsAccuracy (%),

In Table 4, NNDP algorithm shows superior performance on data sets with different noise levels. In the small office building data set with low noise levels, the NNDP algorithm achieves an accuracy of 92.5% with a computation time of 23.5 seconds. In contrast, the ICP algorithm has an accuracy of 89.0% and takes 30.0 seconds. The RANSAC algorithm is 88.5% accurate and takes 29.0 seconds. The NNDP algorithm maintained high accuracy and low computation time at all noise levels tested, demonstrating its robustness and efficiency in processing intelligent building point cloud data. The results show that the NNDP algorithm can effectively deal with the noise in the real world, provides better performance than the existing technology, and is suitable for the feature extraction of intelligent building point cloud data.

5 Discussion

The accuracy of NNDP algorithm reached 94.32%, compared with 85.00% of directional FAST and rotating BRIEF feature matching combined with RANSAC algorithm mentioned in literature [6]. The normal vector feature point cloud alignment algorithm mentioned in literature [7] combined with fast point feature histogram has an accuracy of 88.00%. NNDP algorithm showed significant advantages. In terms of computing time, the average processing time of the NNDP algorithm was 24.37 seconds, which was much lower than the 300 seconds of the bidirectional long short-term memory model based on deep learning mentioned in literature [10] and the 250 seconds of the radial basis function fuzzy logic neural network algorithm mentioned in literature [11]. These results demonstrated a notable enhancement in the efficiency of the NNDP algorithm. The NNDP algorithm showed excellent performance on data sets with different noise levels. For example, at high noise levels, the accuracy of the NNDP algorithm could still reach 89.5%, while the accuracy of the ICP and RANSAC algorithms dropped to 83.0% and 81.5%, respectively. This showed that NNDP algorithm had better robustness when processing noisy data. The natural neighborhood method significantly reduced the registration error by identifying the natural neighborhood of each point and extracting the local structure information. After the first iteration, the registration errors decreased by 0.068mm and 0.021mm, respectively. NNDP algorithm solved the limitation of traditional algorithm in dealing with nonlinear and high dimensional data. This point was also reflected in the clustering algorithm of peak domain density based on natural neighborhood mentioned in literature [9]. However, the NNDP algorithm extends this concept and improves the applicability and accuracy of the algorithm through decomposable graph filtering and feature extraction. The NNDP algorithm takes into account the characteristics of smart building data, such as data sparsity and heterogeneity. This allows algorithms to more efficiently process these complex datasets where traditional algorithms may have limitations.

6 Conclusion

In smart buildings, the application of sensors, intelligent control systems, and big data analysis technologies enables buildings to achieve automated, intelligent management, and operation. The study proposed an NNDP method for processing and analyzing point cloud data in smart buildings. The research findings indicated that the accuracy of NNDP algorithm was 94.32%, the recall rate was 91.76%, the average F1 value was 93.45%, and the time was 24.37s. After using NNDP algorithm for registration, the registration error decreased by 0.068mm and 0.021mm respectively after the first iteration, which proved that NNDP algorithm could effectively improve the efficiency of registration algorithm. When the noise standard deviation was 0.1, the number of bad points generated by NNDP noticeably increased. Moreover, as the noise standard deviation further increased to 0.3, the number of bad points produced by NNDP showed a clear increasing trend. NNDP could extract multiple features from point cloud data, facilitating the design, management, and maintenance of smart buildings. This algorithm can effectively analyze both local and global features in point cloud data and demonstrates good performance in tasks such as modeling, classification, segmentation, among others. However, NNDP requires computations for each point's neighborhood, involving significant operations on point cloud data and resulting in higher computational complexity, thus leading to longer processing times. Future research could optimize the algorithm through GPU-accelerated and distributed computing methods to improve its real-time performance and large data set processing capabilities in smart building systems.

References

[1] A. D'Amico, and G. Ciulla, “An intelligent way to predict the building thermal needs: ANNs and optimization,” Expert Systems with Applications, vol. 191, no. Apr., pp. 116293.1-116293.18, 2022. https://doi.org/10.1016/j.eswa.2021.116293

[2] M. Abedi, and M. Z. Naser, “RAI: Rapid, Autonomous and Intelligent machine learning approach to identify fire-vulnerable bridges,” Applied Soft Computing, vol. 113, no. Pt. A, pp. 107896-1-107896-12, 2021. https://doi.org/10.1016/j.asoc.2021.107896

[3] B. Fan, and X. Xing, “Intelligent prediction method of building energy consumption based on deep learning,” Scientific programming, vol. 2021, no. Pt.14, pp. 3323316.1-3323316.9, 2021. https://doi.org/10.1155/2021/3323316

[4] A. Laracaballero, M. P. Hu, “A Population-Based Local Search Algorithm for the Identifying Code Problem,” Mathematics, vol. 11, no. 20, pp. 1-17. 2023. https://doi.org/10.3390/math11204361

[5] Z. S. Li, Y. K. Yang, and J. C. Zhang, “Filtering feature selection algorithm based on entropy weight method,” Journal of Northeastern University (Natural Science), vol. 43, no. 7, pp. 921-929, 2022. https://doi.org/10.12068/j.issn.1005-3026.2022.07.002

[6] T. Yang, S. Jia, B. Yang, and C. Kan, “Research on tracking and registration algorithm based on natural feature point,” Intelligent Automation and Soft Computing, vol. 28, no. 3, pp. 683-692, 2021. https://doi.org/10.32604/IASC.2021.017235

[7] J. Zhang, Z. Qiao, and S. Wang, “An accurate pose measurement method of workpiece based on rapid extraction of local feature points,” Optoelectronics Letters, vol. 18, no. 6, pp. 372-377, 2022. https://doi.org/10.1007/s11801-022-1152-4

[8] J. Wang, “Design of intelligent construction system for assembly building based on improved IoT,” Informatica, vol. 48, no. 10, 2024. https://doi.org/10.31449/inf.v48i10.5889

[9] D. Chen, T. Du, J. Zhou, and T. Shen, “A domain density peak clustering algorithm based on natural neighbor,” Intelligent data analysis, vol. 27, no. 2, pp. 443-462, 2023. https://doi.org/10.3233/IDA-216541

[10] S. Kaur, A. Bala, and A. Parashar, “GA-BiLSTM: an intelligent energy prediction and optimization approach for individual home appliances,” Evolving Systems, vol. 15, no. 2, pp. 413-427, 2024. https://doi.org/10.1007/s12530-023-09529-6

[11] X. Yu, “An RBF fuzzy logic neural network algorithm for construction resource scheduling,” Journal of Intelligent and Fuzzy Systems, vol. 41, no. 4, pp. 4937-4945, 2021. https://doi.org/10.3233/JIFS-189980

[12] X. Wang, Q. Chen, and J. Wang, “Fuzzy rough set based sustainable methods for energy efficient smart city development,” Journal of Intelligent and Fuzzy Systems, vol. 40, no. 4, pp. 8173-8183, 2021. https://doi.org/10.3233/JIFS-189640

[13] C. Li, Y. Peng, P. Peng, and L. Cao, “A study of fuzzy modeling analysis of factors influencing socially regulation of learning performance in an online environment,” Journal of Intelligent and Fuzzy Systems, vol. 41, no. 3, pp. 4639-4649, 2021. https://doi.org/10.3233/JIFS-189724

[14] P. S. Varma, and V. Anand, “Intelligent scanning period dilation-based Wi-Fi fingerprinting for energy efficient indoor positioning in IoT applications,” Journal of Supercomputing, vol. 79, no. 7, pp. 7736-7761, 2023. https://doi.org/10.1007/s11227-022-04980-9

[15] J. Bai, Y. Ren, and J. Zhang, “Adaptive momentum with discriminative weight for neural network stochastic optimization,” International Journal of Intelligent Systems, vol. 37, no. 9, pp. 6531-6554, 2022. https://doi.org/10.1002/int.22854

[16] Y. Jiao, “Digital music waveform analysis and retrieval based on feature extraction algorithm,” Advances in Multimedia, vol. 2021, no. Pt.1, pp. 7131992.1-7131992.10, 2021. https://doi.org/10.1155/2021/7131992

[17] N. R. Aravind, S. Kalyanasundaram, and A. S. Kare, “Vertex partitioning problems on graphs with bounded tree width,” Discrete Applied Mathematics, vol. 319, no. 1, pp. 254-270, 2022. https://doi.org/10.1016/j.dam.2021.05.016

[18] R. S. Dornelas, and D. A. Lima, “Correlation filters in machine learning algorithms to select demographic and individual features for autism spectrum disorder diagnosis,” Journal of Data Science and Intelligent Systems, vol. 3, no. 1, pp. 7-9, 2023. https://doi.org/10.47852/bonviewjdsis32021027

[19] Z. Li, and J. Shan, “RANSAC-based multi primitive building reconstruction from 3D point clouds,” ISPRS Journal of Photogrammetry and Remote Sensing, vol. 185, no. 1, pp. 247-260, 2022. https://doi.org/10.1016/j.isprsjprs.2021.12.012

[20] J. Li, Q. Hu, and M. Ai, “Point cloud registration based on one-point RANSAC and scale-annealing biweight estimation,” IEEE Transactions on Geoscience and Remote Sensing, vol. 59, no. 11, pp. 9716-9729, 2021. https://doi.org/10.1109/TGRS.2020.3045456

[21] R. A. Kuçak, “The feature extraction from point

clouds using geometric features and ransac algorithm," Advanced LiDAR, vol. 2, no. 1, pp. 15-20, 2022.
[22] H. Chen, W. Chen, R. Wu, and Y. Huang, "Plane segmentation for a building roof combining deep learning and the RANSAC method from a 3D point cloud," Journal of Electronic Imaging, vol. 30, no. 5, pp. 053022-053022, 2021. https://doi.org/10.1117/1.JEI.30.5.053022
[23] L. Sun, "RANSIC: Fast and highly robust estimation for rotation search and point cloud registration using invariant compatibility," IEEE Robotics and Automation Letters, vol. 7, no. 1, pp. 143-150, 2021. https://doi.org/10.48550/arXiv.2104.09133