RANDOMISED COMPRESSION RATIOS FOR EFFECTIVE LARGE POINT CLOUD PROCESSING USING COMPRESSIVE SENSING

Z. Qiu[1,2,*], S. Nagesh[3], J. Martínez-Sánchez[1], P. Arias[1]

[1] CINTECX, Universidade de Vigo, Applied Geotechnology Group, Vigo, Spain - (zhouyan.qiu, joaquin.martinez, parias)@uvigo.es
[2] ICT & Innovation Department, Ingeniería Insitu, Vigo, Spain - zhouyan.qiu@ingenieriainsitu.com
[3] Fraunhofer-Institut für Hochfrequenzphysik und Radartechnik FHR, Wachtberg, Germany - saravanan.nagesh@fhr.fraunhofer.de

KEY WORDS: Compressive sensing, Point Cloud, Point Cloud Compression, Sparse Representation, Lossy Compression, 3D Modeling.

ABSTRACT:

Effectively navigating the intricacies of extensive 3D point cloud data in urban environments poses a series of formidable computational challenges. These challenges are primarily attributed to the substantial data volume and density inherent in urban settings, the presence of noise and inconsistencies within the collected data, and the constraints imposed by limited transmission bandwidth, which consequently impact storage requirements. This paper introduces an innovative methodology for handling large point cloud datasets, based on concepts from Sparse Signal Processing (SSP), also known as compressive sensing. The proposed approach integrates well known geometric data manipulation such as the Octree to work hand in hand with SSP, as unified method. Through experimental validation using the Santiago Urban Dataset (SUD), we demonstrate the effectiveness of our method in achieving high data fidelity, as measured by Peak Signal-to-Noise Ratio (PSNR) values reaching approximately 60 dB even at substantial compression ratios. Comparative analysis against traditional methods, including those implemented in the widely used Point Cloud Library (PCL), reveals the superior performance of our proposed methodology. The results underscore the robustness and efficiency of our approach, positioning it as a compelling alternative for compressing extensive 3D point cloud data. This has crucial implications for diverse applications, ranging from city planning to rapid and effective disaster response.

1. INTRODUCTION

Advanced 3D data acquisition methods, such as Light Detection and Ranging (LiDAR), have become increasingly important in various fields (Qiu et al., 2023), including but not limited to, 3D modeling, city planning, and geospatial analysis (Otepka et al., 2013). These technologies generate point cloud data, a specialized form of spatial data structure that has increasingly become a foundational element in decision-making across a wide array of applications. As the demand for higher spatial coverage and resolution grows, the resulting datasets are becoming increasingly large and complex (Biljecki et al., 2014). This expansion in data size presents significant challenges in data storage, transmission, and real-time processing (Biljecki et al., 2015). These challenges are particularly significant in urban settings where high-resolution point cloud data is essential for activities like infrastructure monitoring and city planning (Pu and Vosselman, 2009).

The traditional methods of data compression for point clouds, such as Run-Length Encoding (RLE) (Golomb, 1966) and Huffman coding (Knuth, 1985), although pioneering in the field, have shown significant limitations when applied to large-scale 3D datasets (Cao et al., 2019). These early methods, primarily designed for simpler data formats, often struggle to maintain the integrity of complex point cloud data, leading to the loss of essential details critical for accurate scene interpretation and analysis (Peng and Kuo, 2005). This loss is particularly detrimental in applications where precision is paramount, such as in geospatial analysis, autonomous vehicle navigation, and 3D modelling.

To overcome these challenges, researchers have been motivated to explore more advanced techniques. Octree encoding, a method that spatially organises data for hierarchical compression, emerged as a significant advancement. This technique, by partitioning the space into octants, allows for an efficient representation and compression of 3D point clouds, which is particularly beneficial in handling large datasets with high spatial complexity (Schnabel and Klein, 2006).

The introduction of geometry-based compression techniques explored another leap forward. These methods focus on efficiently encoding the geometric structure of point clouds, often employing sophisticated mathematical models to achieve higher compression ratios without substantial loss of quality (Graziosi et al., 2020).

Deep learning has been made significant advancements to address a range of technological challenges (Hooda et al., 2022). In point cloud compression, Convolutional Neural Networks (CNNs) have emerged as effective tools, which exploit higher-dimensional data correlations to improve compression performance (Guarda et al., 2020). Additionally, other neural network architectures like Recurrent Neural Networks (RNNs) (Tu et al., 2019) and Fully Connected Neural Networks (Yan et al., 2019) are also being utilised in point cloud compression applications.

Diverging from the deep learning methods, our approach integrates principles of Sparse Signal Processing, with a focus on Compressive Sensing (CS) (Candès et al., 2006). CS operates on the concept that a sparse signal in a certain domain can be effectively reconstructed using a limited number of linear, non-adaptive measurements. This technique is aptly suited for point cloud data, which typically displays sparsity in either spatial or frequency domains. The proposed method aims to integrate

---
* Corresponding author

the core principles of CS with the foundational concepts of established data compression methodologies. It leverages the benefits of randomised projections, a technique that has shown promise in efficiently encoding high-dimensional data into a lower-dimensional space without significant loss of information (Nagesh et al., 2022). This integration aims to preserve the precision in scene interpretation, a critical aspect often compromised in traditional methods, while optimising computational efficiency. This efficiency is crucial for real-time applications like autonomous driving and augmented reality, where rapid data processing and transmission are essential. Furthermore, this approach opens up new possibilities in handling vast datasets with limited storage and bandwidth resources, a common challenge in the era of big data.

In summary, the evolution from basic to advanced point cloud data compression techniques reflects a growing need to handle more complex datasets efficiently without sacrificing data fidelity. The proposed method stands at the forefront of this evolution, promising a balance between computational efficiency and the preservation of crucial data details for accurate 3D scene interpretation.

2. METHOD

Our strategy for compressing extensive point cloud datasets combines geometric data manipulation with the principles of compressive sensing. As can be seen from Figure 1, the entire process can be divided into the four essential phases: Data Pre-processing, Compression Framework, Reconstruction and Assessment, and Quality Check.

2.1 Data Pre-processing

The Data Pre-processing phase is the initial step in our pipeline and is crucial for ensuring the quality and reliability of the subsequent compression and reconstruction processes. This phase consists of two main steps: Data Cleaning and Coordinates Conversion.

2.1.1 Data Cleaning This step focuses on cleaning the raw point cloud data by removing inconsistent and duplicate points. Inconsistent points, often outliers or anomalies, can distort the true geometry of the scene and introduce errors in later stages of compression and reconstruction (Rusu and Cousins, 2011). To minimize computational burden, duplicate points are also removed.User The cleaning process employs K-Nearest Neighbours (KNN) noise filtering, to produce a refined point cloud dataset (See Algorithm 1).

2.1.2 Coordinates Conversion The second step focuses on transitioning the point cloud data from a global to a local coordinate system. This adjustment is crucial for two main reasons: it simplifies computational tasks and makes data manipulation more straightforward. Affine transformations are employed for this conversion, ensuring that the geometric properties of the point cloud remain intact while preparing the data for subsequent processing stages (Hartley and Zisserman, 2004).

These pre-processing steps are not only essential for improving the efficiency of the compression algorithm but also crucial for maintaining the integrity of the data, ensuring an accurate representation in the compressed point cloud.
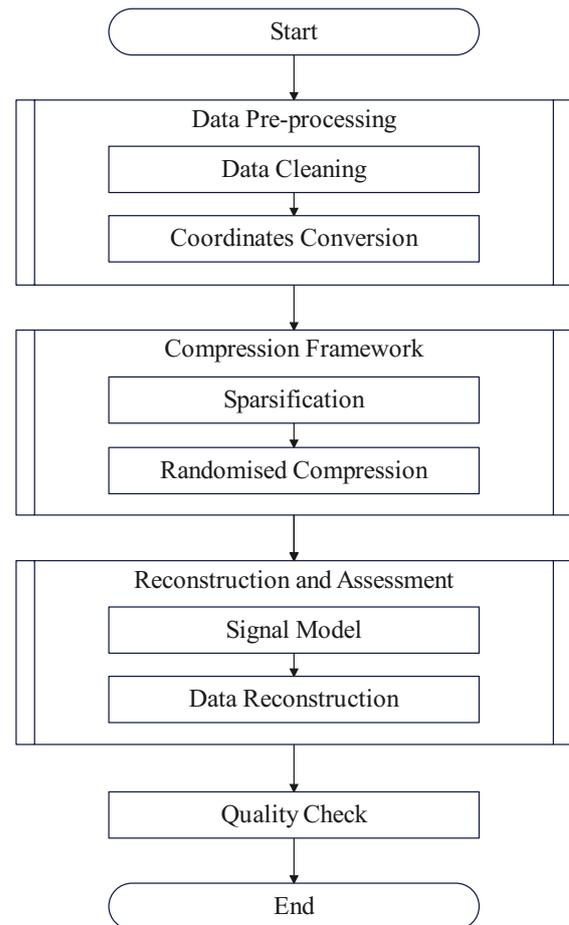
Figure 1. The pipeline of our methodology.

2.2 Compression Framework

The core of our method lies in the Compression Framework, which is designed to efficiently compress point cloud data while preserving its essential features.

For this we adapt the concepts of Sparse Randomised Projections (SRP) which stands as forefront of advanced techniques within the domain of CS, presenting notable advantages for efficient signal acquisition and restoration. In principle, SRP involves the stochastic projection of a high-dimensional signal onto a lower-dimensional subspace.

This framework integrates two primary stages: Sparsification and Randomised Compression. These techniques are sequentially applied to the point cloud data, ensuring both computational efficiency and data integrity.

2.2.1 Sparsification To obtain Sparse Data Representations suited for the CS frame we employ two intermediate stages:

- **Data Dimensionality Reduction via Octree Segmentation**: Octree segmentation is the first step in our compression framework. The point cloud data is partitioned into hierarchical octree structures, which allows for a more manageable and computationally efficient representation of the 3D space (Samet, 1990). This hierarchical segmentation enables us to apply different levels of compression to different regions, thereby optimising the trade-off between data size and feature preservation (Elseberg et al., 2013).

Algorithm 1 K-Nearest Neighbours Noise Filtering for Point Clouds

---

1: **Input:** Point cloud $P = \{p_1, p_2, \ldots, p_n\}$, Number of neighbours K
2: **Output:** Refined point cloud $\hat{P}$
3: **procedure** KNNNOISEFILTERING(P, K)
4:     **for** each point p_i in P **do**
5:         Find K nearest neighbors $N_K(p_i) = \{n_1, n_2, \ldots, n_K\}$ of p_i
6:         Initialise $\hat{p}_i = 0$
7:         **for** each neighbor n_j in $N_K(p_i)$ **do**
8:             Calculate distance $d(p_i, n_j)$
9:             $\hat{p}_i = \hat{p}_i + n_j$
10:         **end for**
11:         $\hat{p}_i = \frac{1}{K}\hat{p}_i$ ▷ Averaging the positions
12:     **end for**
13:     $\hat{P} = \{\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_n\}$
14:     **return** $\hat{P}$
15: **end procedure**

In traditional octree segmentation, the criteria for establishing an octree are typically based on the level or the area an octant occupies. However, in our approach (Algorithm 2), we utilise "binCapability" as the limiting factor. This is due to the fact that subsequent operations target each leaf individually, necessitating such a restriction. The "binCapability" represents the number of points contained within each leaf.

Algorithm 2 Octree Segmentation for Point Cloud

1: **procedure** CREATEOCTREE($node, points, binCapacity$)
2:     **if** length($points$) $\leq binCapacity$ **then**
3:         $node.points \leftarrow points$
4:         $node.isLeaf \leftarrow$ True **return**
5:     **end if**
6:     $node.isLeaf \leftarrow$ False
7:     $node.children \leftarrow$ Subdivide($node$)
8:     **for** $child \in node.children$ **do**
9:         $childPoints \leftarrow$ FilterPointsToOctant($points, child$)
10:         CreateOctree($child, childPoints, binCapacity$)
11:     **end for**
12: **end procedure**
13:
14: **procedure** SUBDIVIDE($node$)
15:     $children \leftarrow$ Initialize 8 empty nodes for each octant
16:     Set the boundaries for each child based on $node$ **return** $children$
17: **end procedure**
18:
19: **procedure** FILTERPOINTSTOOCTANT($points, octant$)
20:     $filteredPoints \leftarrow$ Empty list
21:     **for** $point \in points$ **do**
22:         **if** $point$ is inside $octant$ **then**
23:             Append $point$ to $filteredPoints$
24:         **end if**
25:     **end for return** $filteredPoints$
26: **end procedure**

- *Sparse Representation via Basis Transformation:*
  Wavelets are mathematical functions that can transform data into different frequency components (Mallat, 1989). By employing wavelet basis transformation, we can represent the point cloud data in a form that is more amenable to compression. This transformation not only reduces the data size but also allows for the preservation of essential features in both low and high-frequency domains (Donoho, 1995). The wavelet coefficients generated in this phase are then quantised and encoded, forming the compressed data representation.

Typically, the outcomes of a wavelet transform are influenced by the selection of the wavelet basis. The choice of wavelet basis is commonly based on some general criteria (Luisa, 2012):

- Compact Support: The wavelet must possess compact support, implying it is zero beyond a specific finite range. This characteristic guarantees computational efficiency in the wavelet transform. A longer support length typically demands more computation and produces higher amplitude wavelet coefficients.

- Orthogonality: Orthogonal wavelets streamline both the analysis and the reconstruction phases. The orthogonality of wavelets guarantees the absence of superfluous information in the transformed data. However, typically, enhanced regularity is associated with extended support length, leading to increased computational time. Thus, a balance between regularity and support length is essential.

- Vanishing Moments: Vanishing moments constrain a wavelet's capacity to depict polynomial characteristics in a signal. Wavelets with increased vanishing moments can more efficiently represent higher-degree polynomials.

- Regularity: Intuitively, regularity can be considered a measure of smoothness, which can affect the ability to capture signal details (Han and Han, 2017).

- Adaptability: The ability to adjust or tune the wavelet according to the specific characteristics of the data can be beneficial.

In this study, we employed two distinct wavelet bases: the Haar wavelet (Haar, 1909) and the Biorthogonal wavelet (Mallat, 1999). The Haar wavelet, known for its simplicity and compact support, offers efficient computational properties. On the other hand, the Biorthogonal wavelet is renowned for its ability to maintain both reconstruction and decomposition filters, allowing for perfect reconstruction of signals and images, which is crucial in preserving the integrity of the original data. The choice of these wavelets was driven by their complementary characteristics, enabling a comprehensive analysis of the data from different wavelet perspectives.

Octree segmentation ensures a judicious handling of the measurement data, while the Basis transformation efficiently captures the sparsity patterns in the scene coefficients. Presenting the possibilities to approach the problem as an undetermined set of equations, best solved using CS based algorithms.

2.2.2 Randomised Compression To further enhance the fidelity of data recovery, the employment of an incoherent measurement matrix is preferred. Achieving this involves introducing a randomly generated component into the underlying signal model. This intentional introduction of randomness ensures that the resulting measurement matrix is well conditioned, thereby satisfying the restricted isometric property for a defined sparsity order, as outlined in the work by Donoho et al. (Donoho, 2006).

The introduction of the randomised component happens simultaneously with the former stage through randomised compression of the obtained Octree segmentation, by considering radom

Gaussian matrix of similar size multiplied in combination. This ensure the mutual coherence of the overall sensing matrix is lowered , making each column well discernible and thus making the matrix more well conditioned.

2.3 Signal Model

The signal model used in the study is presented in the following equation:

$$\mathbf{y} = \mathbf{PABx} + \mathbf{n} \tag{1}$$

where: $\mathbf{y}$ is the measurement vector containing the segmented point cloud of size $C^{M \times 1}$. $\mathbf{P}$ is the projection matrix which inducing the randomised projections of size $C^{M \times P}$. $\mathbf{S}$ is the measurement matrix of size $C^{P \times N}$ and $\mathbf{B}$ is the sparse basis transformation matrix of size $C^{N \times N}$. $\mathbf{x}$ is the vector of unknown coefficients to be estimated of size $C^{N \times 1}$. We consider, the combination of the matrices $\mathbf{P}$, $\mathbf{S}$ and $\mathbf{B}$ as $\mathbf{\Psi}$ of size $C^{M \times N}$, as the sensing matrix. The presented frame work is suited for CS algorithms, we consider the Basis Pursuit Denoising Algorithm, as given in Algorithm 3 as the solver of choice.

Algorithm 3 Basis Pursuit Denoising (BPDN)

1: **Input:** Measurement matrix $\mathbf{\Psi}$, observations $\mathbf{y}$, regularisation parameter λ
2: **Output:** Sparse solution vector $\mathbf{x}$
3:
4: **Initialise:** Choose an initial guess for $\mathbf{x}$
5:
6: **while** Not converged **do**
7: $\quad \mathbf{x} \leftarrow \min_{\mathbf{x}} \left(\frac{1}{2} \|\mathbf{y} - \mathbf{\Psi x}\|_2^2 + \lambda \|\mathbf{x}\|_1 \right)$ $\quad \triangleright$ Use a BPDN solver
8: $\quad$ **Convergence check:** Check if the solution has converged
9: **end while**
10:
11: **Output:** Sparse solution vector $\mathbf{x}$

Basis Pursuit (BP) is a crucial technique in CS for the sparse approximation of a vector $\mathbf{x}$ from observations $\mathbf{y}$ (Equation 2).

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \mathbf{y} = \mathbf{\Psi x} \tag{2}$$

Formulated as a convex optimization problem, BP effectively recovers $\mathbf{x}$ only under the condition of noise-free measurements in $\mathbf{y}$ (Chen et al., 2001). The presence of noise significantly affects its accuracy in signal reconstruction. When measurements are corrupted by noise, exact reconstruction of the original signal is not the primary goal. Instead, the focus shifts to denoising. This is achieved by relaxing the equality constraint in the optimization problem.

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \frac{1}{2} \|\mathbf{y} - \mathbf{\Psi x}\|_2^2 \leq \epsilon \tag{3}$$

In Equation 3, $\|\mathbf{y} - \mathbf{\Psi x}\|_2^2$ is accounting for the noise in the measurements, and $\|\mathbf{x}\|_1$ is promoting the sparsity in the solution. The constraint $\frac{1}{2} \|\mathbf{y} - \mathbf{\Psi x}\|_2^2 \leq \epsilon$, where $\| \cdot \|_2$, also known as the Euclidean norm, represents the length or size of a vector (Golub and Van Loan, 2013). This norm ensures that the solution remains within an acceptable error bound ϵ. The inclusion of the Euclidean norm is crucial for effectively handling

the noise present in the measurements, as it quantifies the deviation of the reconstructed signal from the noisy observations (Rani et al., 2018).

In Algorithm 3, we address the solution of the BPDN problem by adopting its Lagrangian form. This approach transforms BPDN into an unconstrained optimization problem, as detailed in Equation 4.

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \left(\frac{1}{2} \|\mathbf{y} - \mathbf{\Psi x}\|_2^2 + \lambda \|\mathbf{x}\|_1 \right) \tag{4}$$

where $\|\mathbf{y} - \mathbf{\Psi x}\|_2^2$ accounts for the noise in the measurements, and $\lambda \|\mathbf{x}\|_1$ promotes sparsity in the solution.

The BPDN solver applied in Algorithm 3 is Fixed-Point Continuation (FPC) (Hale et al., 2007).

2.4 Recovery Guarantees

When considering the reliability of recovery with CS-based methods, such as Basis Pursuit, satisfying the Restricted Isometric Property (RIP) for the underlying sensing matrix plays a crucial part in ensuring stable and uniform recovery for a considered level of sparsity. The RIP leverages the property of the Restricted Isometry Constant (RIC) to establish the most stringent performance guarantees presently known in the field. For a given sparsity level S, the Restricted Isometry Constant, denoted as δ_S, is defined as the smallest positive constant that satisfies certain mathematical criteria, which is shown in the following equation:

$$(1 - \delta_S)\|\mathbf{x}\|_2^2 \leq \|\mathbf{\Psi x}\|_2^2 \leq (1 + \delta_S)\|\mathbf{x}\|_2^2 \tag{5}$$

holds for all vectors with $\|\mathbf{x}\|_0 \leq S$, i.e., the RIC establishes bounds for the singular values of the sub-matrices obtained by selecting any S columns from the complete $\mathbf{\Psi}$.

The randomised projection matrix $\mathbf{P}$ considered in this study ensures, the columns the sensing matrix is highly incoherent, thereby satisfying the RIP metric (for a given sparsity value) and thus ensuring reliability in performance of reconstruction.

2.5 Quality Check

To ensure the reliability of the compressed data, we compared the reconstructed dataset against the original dataset. This comparison was primarily facilitated using the Peak Signal-to-Noise Ratio (PSNR), a widely accepted metric for evaluating the quality of compressed data, especially in the context of point clouds compression. The PSNR is computed using the following equation:

$$PSNR = 10 \times \log_{10} \left(\frac{MAX^2}{MSE} \right) \tag{6}$$

Where: MAX represents the maximum possible point intensity value in the point cloud. MSE denotes the Mean Squared Error between the original and the reconstructed point cloud data.

Higher PSNR values suggest a closer resemblance to the original, indicating minimal compression loss. In contrast, lower PSNR values may indicate more significant data loss or quality degradation. Detailed PSNR results and their implications will be elaborated in the next section.

3. RESULTS

3.1 Dataset

In our research, we predominantly rely on the Santiago Urban Dataset (SUD) (González Collazo et al., 2022). This dataset, originating from Santiago de Compostela, Spain, amalgamates data sourced from two distinct laser scanning techniques: Mobile Laser Scanning (MLS) and Handheld Mobile Laser Scanning (HMLS). Together, they cover an expansive area of approximately 2 km of urban thoroughfares.

3.2 Experiments and Results

To comprehensively evaluate the potential of our compression method in urban data analysis, we conducted a series of experiments. These experiments were designed to assess the interplay and impact of various factors on the compression outcomes. Specifically, we varied the wavelet basis type, the measurement matrix employed, the chosen sparsity basis, and the sampling ratio. This methodological diversity ensured a broad spectrum of results, allowing us to gauge the efficacy and versatility of our approach. By systematically altering these parameters, we were able to derive a multitude of compression outcomes, each accompanied by its unique set of performance metrics. This rigorous approach not only underscores the robustness of our methodology but also provides valuable insights into the optimal configurations for different urban data scenarios.

3.2.1 Haar and Biorthogonal Wavelets

Figure 2 portrays the results obtained using different wavelets. In our analysis, both the Biorthogonal and Haar wavelets demonstrated comparable Peak Signal-to-Noise Ratio (PSNR) values across a range of compression ratios. This similarity in performance suggests that both wavelets are equally effective in maintaining the quality of the reconstructed data relative to the original, despite their inherent differences in structure and design.
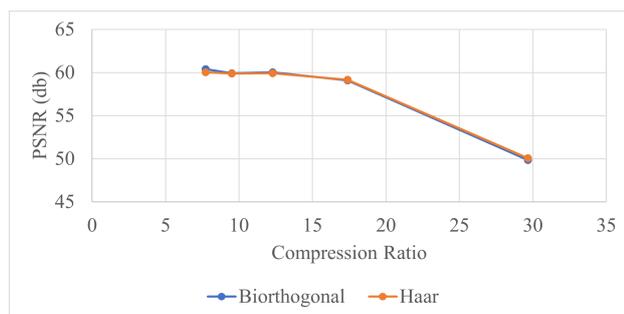
Figure 2. The Compression Ratios and PSNR for different wavelet types. Both Biorthogonal and Haar wavelets showcase similar PSNR values at varying compression ratios.

3.2.2 Measurement Matrices

From Figure 3 we can see that the primary variations in PSNR are due to the matrices themselves. In our study, we consistently used bior1.1 wavelets and FFT, which highlighted that the primary differences in Peak Signal-to-Noise Ratio (PSNR) were due to the measurement matrices used. The Bernoulli Matrix and Gaussian Random Matrix both showed high PSNR values across different compression ratios, indicating they maintain data integrity effectively during compression. Conversely, the Part Fourier Matrix and Toeplitz Matrix displayed lower

PSNR, particularly at higher compression ratios, suggesting they are less effective in preserving data quality under compression. This underscores the importance of matrix selection in data compression processes.
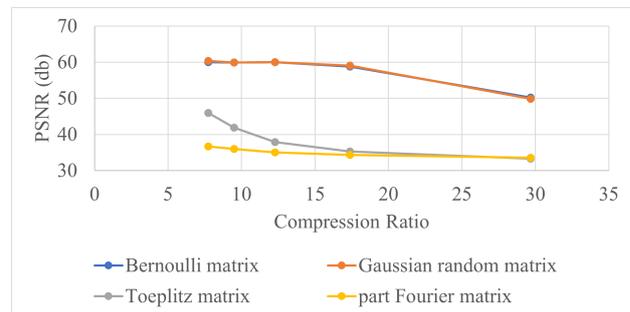
Figure 3. The graphical representation illustrates Compression Ratios and PSNR values for various measurement matrices. Bernoulli Matrix and Gaussian Random Matrix exhibit closely aligned, high PSNR values across compression ratios, reflecting superior data integrity in compression. In contrast, Part Fourier Matrix and Toeplitz Matrix consistently show lower PSNR, particularly at higher compression ratios, indicating reduced efficiency in maintaining data quality.

3.2.3 Sparsity Basis

Analysing Figure 4 which compares the results of Discrete Cosine Transform Basis (DCT) and Fast Fourier Transform Basis (FFT). In our study, the Fast Fourier Transform (FFT) consistently achieved higher Peak Signal-to-Noise Ratio (PSNR) than the Discrete Cosine Transform (DCT) across all compression ratios and with both Bernoulli and Gaussian Random Matrices. This advantage of FFT was more evident at higher compression ratios, though the difference diminished at lower ratios. Additionally, while the type of matrix (Bernoulli or Gaussian) had a minor impact, the choice between FFT and DCT was the more significant factor in determining PSNR performance.

4. DISCUSSION & CONCLUSION

4.1 Discussion

Our experimental design incorporated variations in wavelet selection, measurement matrix, sparsity basis, and sampling ratio, resulting in a comprehensive exploration of compression outcomes and performance metrics. In the conducted experiments, the precision of the points was set to millimeters, aligning with the accuracy level of the Santiago Urban Dataset (SUD), which is up to 5 mm.

Compression Ratio	BPP	Bernoulli Matrix	Gaussian Random Matrix	PCL
29.67	5.38	50.16	49.85	-
22.86	6.98	-	-	34.15
17.38	9.18	58.82	59.09	-
12.28	12.99	59.98	60.02	-
9.5	16.79	59.90	59.91	-
7.73	20.64	59.98	60.38	-
4.90	32.55	-	-	79.54

Table 1. The comparison of Compression Ratio, bpp and PSNR (dB) for different matrices and PCL library

Table 1 compares the Peak Signal-to-Noise Ratio (PSNR) in decibels (dB) of our methods using Bernoulli and Gaussian Random Matrices against the Point Cloud Library (PCL) (Rusu and Cousins, 2011) across different compression ratios. Additionally, it includes Bits Per Point (BPP) for each method. The
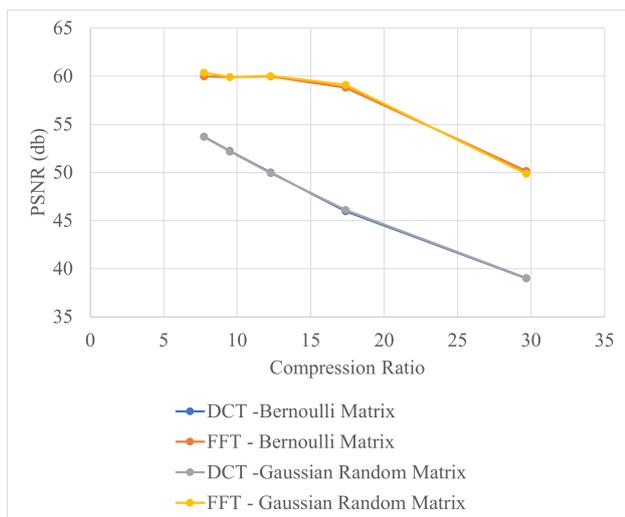
Figure 4. The Compression Ratios and PSNR for different measurement matrices and sparsity basis. FFT consistently outperforms DCT in terms of PSNR across all compression ratios and both matrix types. The superiority of FFT becomes more pronounced at higher compression ratios, though the difference narrows as the ratio decreases. While both Bernoulli and Gaussian Random Matrices exhibit similar trends, the choice of matrix slightly influences the resultant PSNR.

BPP value is calculated using Equation 7. It is a crucial metric as it provides insight into data efficiency, complementing the compression ratio by showing how effectively each point is represented in the compressed data. At higher compression ratios (e.g., 29.67 and 17.38), both the Bernoulli and Gaussian Random Matrix methods demonstrate superior PSNR values compared to the PCL, indicating a higher quality of data retention despite significant data size reduction. For instance, at a compression ratio of 29.67, the Bernoulli Matrix achieves a PSNR of 50.17 dB, and the Gaussian Random Matrix achieves 49.85 dB. In contrast, the PCL shows its strength at a much lower compression ratio of 4.9, where it achieves a notably high PSNR of 79.54 dB. This suggests that while the PCL may be more efficient at lower compression levels (i.e., less data reduction), while our method excel in scenarios requiring higher compression.

$$\text{BPP} = \frac{\text{Compressed Size (in bits)}}{\text{Number of Points}} \qquad (7)$$

Upon examining the visual representations presented in Figures 5(a) and 5(b), several observations arise regarding the efficacy of our compression and reconstruction techniques. Using the original dataset (Figure 5(a)) as a benchmark, the reconstructed dataset (Figure 5(b)), generated with a Gaussian Random Matrix and a compression ratio of 17.38, achieved a PSNR of 59.0937 dB. This high PSNR value attests to the fidelity of the reconstruction relative to the original data.

The cloud-to-cloud distance calculation, visualised in Figure 6, offers additional insights into spatial disparities between the original and reconstructed datasets. The gradient, transitioning from blue to green, signifies increasing distances, with blue areas indicating minimal deviation and green areas signifying greater discrepancies. The mean distance between the two datasets was determined to be 0.141 m, with a standard deviation of
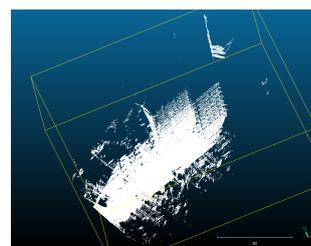
(a) The original dataset
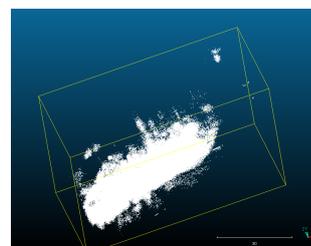
(b) The reconstructed dataset

Figure 5. A comparison of the original and reconstructed point cloud dataset (using Gaussian Random Matrix with compression Ratio 17.38, PSNR is 59.0937 dB)
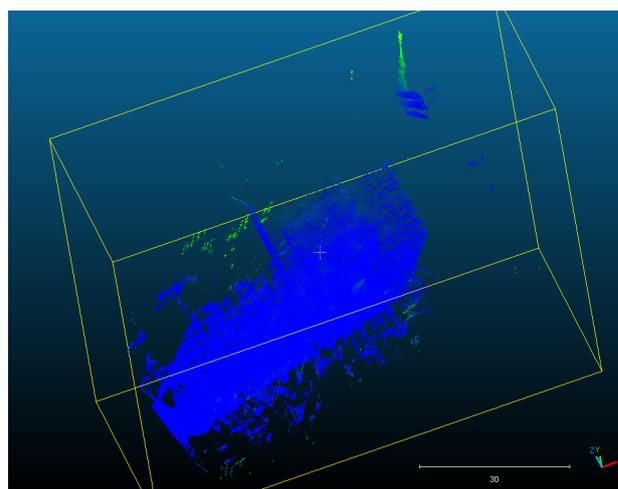
Figure 6. Cloud to cloud distance calculation. A gradient from blue to green represents increasing distances.

0.522 m. The relatively low mean distance, coupled with the high PSNR value, underscores the efficacy of our methodology in preserving the integrity of the original dataset while achieving significant compression.

The reconstruction, as evidenced by the PSNR and mean distance metrics, demonstrates commendable accuracy. However, deviations are evident in specific regions, particularly those marked in green in Figure 6. A closer examination reveals that many of these green-highlighted areas are located towards the peripheries, away from the central region, and include isolated segments detached from the primary structure. These observations suggest that while our compression algorithm effectively retains the integrity of the main regions, there is room for refinement in preserving details in the peripheral and isolated areas. This insight provides valuable guidance for future algorithmic iterations to enhance overall performance and fidelity.

4.2 Conclusion

The efficient management of large point cloud datasets, especially in urban environments, remains a formidable challenge in the realm of 3D data acquisition and processing. This research introduced a novel methodology that integrates principles of geometric data manipulation, wavelet transformations, and sparse signal processing to address this challenge. The demonstrated compression of extensive point cloud datasets, as exemplified by experiments on the Santiago Urban Dataset (SUD), yielded promising results.

The achieved Peak Signal-to-Noise Ratio (PSNR) values, hovering around 60 dB at high compression ratios, signify the method's capability to maintain high data fidelity. Notably, this performance surpasses traditional methods, such as those implemented in the Point Cloud Library (PCL). Visual and quantitative comparisons between the original and reconstructed datasets further affirm the efficacy of our methodology. While core dataset regions were preserved with high fidelity, the findings also pinpoint areas for potential improvement, particularly in peripheral and isolated regions.

The integration of techniques like Octree Segmentation, Wavelet Basis Transformation, and Compressive Sensing into a unified framework has proven to be both innovative and effective. However, as with any pioneering approach, refinement and optimization opportunities persist. Subsequent work can delve deeper into enhancing the compression algorithm, ensuring even greater fidelity across all dataset regions, and exploring its applicability across diverse urban scenarios.

In conclusion, our research establishes a robust and efficient alternative in the compression of extensive large 3D point cloud datasets. The implications of this work are expansive, with potential applications ranging from urban planning to disaster response, emphasising the significance of continued advancements in this scientific domain.

ACKNOWLEDGEMENTS

REFERENCES

Biljecki, F., Ledoux, H., Stoter, J., Zhao, J., 2014. Formalisation of the level of detail in 3D city modelling. *Computers, environment and urban systems*, 48, 1–15.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., Çöltekin, A., 2015. Applications of 3D city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4(4), 2842-2889.

Candès, E. J., Romberg, J., Tao, T., 2006. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2), 489–509.

Cao, C., Preda, M., Zaharia, T., 2019. 3d point cloud compression: A survey. *The 24th International Conference on 3D Web Technology*, 1–9.

Chen, S. S., Donoho, D. L., Saunders, M. A., 2001. Atomic decomposition by basis pursuit. *SIAM review*, 43(1), 129–159.

Donoho, D. L., 1995. De-noising by soft-thresholding. *IEEE transactions on information theory*, 41(3), 613–627.

Donoho, D. L., 2006. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4), 1289-1306.

Elseberg, J., Borrmann, D., Nüchter, A., 2013. One billion points in the cloud–an octree for efficient processing of 3D laser scans. *ISPRS Journal of Photogrammetry and Remote Sensing*, 76, 76–88.

Golomb, S. W., 1966. Run-Length Encodings. *IEEE Transactions on Information Theory*, 12(3), 399–401.

Golub, G. H., Van Loan, C. F., 2013. *Matrix computations*. JHU press.

González Collazo, S. M., Balado, J., Garrido, I., Grandío, J., Rashdi, R., Tsiranidou, E., del Río-Barral, P., Rúa, E., Puente, I., Lorenzo, H., 2022. Santiago Urban Dataset Sud: Combination of Handled and Mobile Laser Scanning Point Clouds. *Available at SSRN 4227615*.

Graziosi, D., Nakagami, O., Kuma, S., Zaghetto, A., Suzuki, T., Tabatabai, A., 2020. An overview of ongoing point cloud compression standardization activities: Video-based (V-PCC) and geometry-based (G-PCC). *APSIPA Transactions on Signal and Information Processing*, 9, e13.

Guarda, A. F., Rodrigues, N. M., Pereira, F., 2020. Deep learning-based point cloud geometry coding: Rd control through implicit and explicit quantization. *2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, IEEE, 1–6.

Haar, A., 1909. *Zur theorie der orthogonalen funktionensysteme*. Georg-August-Universitat, Gottingen.

Hale, E. T., Yin, W., Zhang, Y., 2007. A fixed-point continuation method for l1-regularized minimization with applications to compressed sensing. *CAAM TR07-07, Rice University*, 43, 44.

Han, B., Han, B., 2017. Wavelet Filter Banks. *Framelets and Wavelets: Algorithms, Analysis, and Applications*, 67–151.

Hartley, R., Zisserman, A., 2004. *Multiple View Geometry in Computer Vision*. 2 edn, Cambridge University Press.

Hooda, R., Pan, W. D., Syed, T. M., 2022. A survey on 3D point cloud compression using machine learning approaches. *SoutheastCon 2022*, 522–529.

Knuth, D. E., 1985. Dynamic Huffman Coding. *Journal of Algorithms*, 6(2), 163–180.

Luisa, B., 2012. *Wavelets: A Tutorial in Theory and Applications*. ISSN, Elsevier Science.

Mallat, S., 1999. *A Wavelet Tour of Signal Processing*. Electronics & Electrical, Elsevier Science.

Mallat, S. G., 1989. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence*, 11(7), 674–693.

Nagesh, S., Ender, J., González-Huici, M. A., 2022. Array position optimisation for compressed sensing mimo radar based on mutual coherence minimisation. *2022 23rd International Radar Symposium (IRS)*, 98–103.

Otepka, J., Ghuffar, S., Waldhauser, C., Hochreiter, R., Pfeifer, N., 2013. Georeferenced point clouds: A survey of features and point cloud management. *ISPRS Journal of Photogrammetry and Remote Sensing*, 76, 408-427.

Peng, J., Kuo, C. C. J., 2005. Technologies for 3D mesh compression: A survey. *Journal of Visual Communication and Image Representation*, 16(6), 688-733.

Pu, S., Vosselman, G., 2009. Knowledge based reconstruction of building models from terrestrial laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(6), 575–584.

Qiu, Z., Martínez-Sánchez, J., Arias-Sánchez, P., Rashdi, R., 2023. External multi-modal imaging sensor calibration for sensor fusion: A review. *Information Fusion*, 101806.

Rani, M., Dhok, S. B., Deshmukh, R. B., 2018. A systematic review of compressive sensing: Concepts, implementations and applications. *IEEE access*, 6, 4875–4894.

Rusu, R. B., Cousins, S., 2011. 3D is here: Point Cloud Library (PCL). *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Shanghai, China.

Samet, H., 1990. *The Design and Analysis of Spatial Data Structures*. Addison-Wesley Longman Publishing Co., Inc., USA.

Schnabel, R., Klein, R., 2006. Octree-based Point-Cloud Compression. *PBG@ SIGGRAPH*, 3.

Tu, C., Takeuchi, E., Carballo, A., Takeda, K., 2019. Point cloud compression for 3d lidar sensor using recurrent neural network with residual blocks. *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, 3274–3280.

Yan, W., Liu, S., Li, T. H., Li, Z., Li, G. et al., 2019. Deep autoencoder-based lossy geometry compression for point clouds. *arXiv preprint arXiv:1905.03691*.