

Article

Robust Watermarking Algorithm for Building Information Modeling Based on Element Perturbation and Invisible Characters

Qianwen Zhou ^{1,2}, Changqing Zhu ^{1,2,*} and Na Ren ^{1,2,3}

¹ Key Laboratory of Virtual Geographic Environment, Nanjing Normal University, Ministry of Education, Nanjing 210023, China; zqw@njnu.edu.cn (Q.Z.); 09359@njnu.edu.cn (N.R.)

² State Key Laboratory Cultivation Base of Geographical Environment Evolution, Nanjing 210023, China

³ Hunan Engineering Research Center of Geographic Information Security and Application, Changsha 410018, China

* Correspondence: zcq88@263.net

Abstract: With the increasing ease of building information modeling data usage, digital watermarking technology has become increasingly crucial for BIM data copyright protection. In response to the problem that existing robust watermarking methods mainly focus on BIM exchange formats and cannot adapt to BIM data, a novel watermarking algorithm specifically designed for BIM data, which combines element perturbation and invisible character embedding, is proposed. The proposed algorithm first calculates the centroid of the enclosing box to locate the elements, and establishes a synchronous relationship between the element coordinates and the watermarked bits using a mapping mechanism, by which the watermarking robustness is effectively enhanced. Taking into consideration both data availability and the need for watermark invisibility, the algorithm classifies the BIM elements based on their mobility, and perturbs the movable elements while embedding invisible characters within the attributes of the immovable elements. Then, the watermark information after dislocation is embedded into the data. We use building model and structural model BIM data to carry out the experiments, and the results demonstrate that the signal-to-noise ratio and peak signal-to-noise ratio before and after watermark embedding are both greater than 100 dB. In addition, the increased information redundancy accounts for less than 0.15% of the original data., which means watermark embedding has very little impact on the original data. Additionally, the NC coefficient of watermark extraction is higher than 0.85 when facing attacks such as translation, element addition, element deletion, and geometry–property separation. These findings indicate a high level of imperceptibility and robustness offered by the algorithm. In conclusion, the robust watermarking algorithm for BIM data fulfills the practical requirements and provides a feasible solution for protecting the copyright of BIM data.



Citation: Zhou, Q.; Zhu, C.; Ren, N. Robust Watermarking Algorithm for Building Information Modeling Based on Element Perturbation and Invisible Characters. *Appl. Sci.* **2023**, *13*, 12957. <https://doi.org/10.3390/app132312957>

Academic Editors: Frank Y. Shih and Xin Zhong

Received: 15 November 2023

Revised: 28 November 2023

Accepted: 1 December 2023

Published: 4 December 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: BIM watermarking; element perturbation; invisible character; data security

1. Introduction

The development of smart cities leads to the need for better urban 3D modeling technologies to digitally represent urban geographic entities, and building information modeling (BIM) has emerged as an essential tool for this purpose due to its ability to provide comprehensive information about a city and support micro-spatial analysis [1–3]. However, due to the development of computer multimedia technology, BIM data are becoming more accessible and are susceptible to infringement and leakage during distribution and use [4–6]. Therefore, the secure use of BIM data is becoming increasingly important. Digital watermarking technology has emerged as a cutting-edge solution for copyright protection by hiding copyright information in data to carry out copyright identification, infringement traceability, and content authentication [7–11]. The robust watermark tightly

integrates copyright information with the data to resist a wide range of attacks, making robust watermarking technology for BIM data important content in the field of data security [12–14].

The existing research on robust watermarking algorithms for BIM data is limited, but there are numerous studies on 3D mesh models and point cloud models, which are visualized 3D models like BIM data, that could provide a reference for watermarking research on BIM data. The current watermarking algorithms for 3D models can be grouped into three categories: transform domain-based watermarking algorithms, global feature-based watermarking algorithms, and local feature-based watermarking algorithms.

Robust watermarking algorithms based on the transform domain generally use frequency analysis methods such as discrete cosine transform, wavelet analysis, and streaming harmonic bases to embed the watermark by modifying coefficients obtained from model changes [15–19]. For example, Hamidi et al. proposed a robust 3D mesh watermarking algorithm based on mesh saliency and wavelet transform. The watermark is inserted by quantifying the wavelet coefficients using quantization index modulation (QIM) according to the mesh saliency of the 3D semiregular mesh [18]. While this approach has good imperceptibility, since such algorithms embed watermark information into data transformation coefficients, watermark embedding extraction relies on carrier data transformation, which greatly increases the computational cost of the algorithms. BIM data, on the other hand, comprise fine models with a large number of elements, thus requiring more computationally efficient watermarking algorithms.

Robust watermarking algorithms based on global features embed watermarking information into global feature quantities in the geometric domain, such as histogram offset and region error expansion [20–23]. For example, Jang et al. divided the mesh in a 3D mesh model into several segments based on the shape diameter function (SDF) and then used the histogram mapping function to embed the watermark into each segment [23]. This type of algorithms exhibits robustness against various forms of attacks owing to the inherent stability of global statistical characteristics. Furthermore, in contrast to transform domain algorithms, they demonstrate lower computational demands. However, they are vulnerable to attacks that destroy the overall structure of the model.

Robust watermarking algorithms based on local features use the relative positions of adjacent points, vertex order, vertex coordinate, etc. as data features to embed watermarking information [24–28]. For example, Narendra et al. cluster the vertices in a 3D mesh model by proposing an intelligent fuzzy based clustering scheme and embedded the watermark based on surface-based segmentation [27]. This type of algorithm can select the most suitable watermark embedding location after structural analysis of the 3D model, so they are more flexible than global feature-based algorithms and transform domain algorithms. In addition, since the algorithms do not rely on global features, attacks that destroy the overall shape of the model such as data clipping do not affect the correct detection of the watermark. However, since BIM data have strict topological constraints and a different data structure from 3D mesh models and 3D point cloud models, it is difficult to apply existing algorithms based on local features directly to BIM data.

In summary, transform domain-based watermarking algorithms have better imperceptibility but require higher computational costs. Global feature-based watermarking algorithms reduce the computational costs, but it is difficult to resist attacks that destroy the overall shape of the model. Local feature-based watermarking algorithms are more flexible and resistant to attacks that destroy the shape of the model. Therefore, this class of methods is more suitable for BIM data copyright protection. However, due to data structure differences, existing algorithms based on local features for 3D models are not yet available for BIM data.

Therefore, in order to solve the above problem and fill the gap in the study of robust watermarking of BIM data, this paper presents a novel approach to addressing the problem of securing BIM data via the use of a robust watermarking algorithm. Our method combines the spatial location and attribute information of elements to generate local features.

Considering the data structure characteristics and watermark robustness, we utilized the centroid of the enclosing box to locate elements and establish a mapping mechanism to relate elements to watermarked bits. Then, we perturbed the movable elements and embedded invisible characters into the attribute of the immovable elements. Finally, watermark information was embedded multiple times.

The rest of this paper is organized as follows: Section 2 provides an overview of the proposed algorithm, including its basic idea and preliminaries. In Section 3, we describe our proposed method in detail. Section 4 presents the experimental results, while Section 5 offers a discussion of our findings. Finally, in Section 6, we draw conclusions based on our research.

2. Preliminaries

2.1. Basic Idea

The key to a robust watermarking algorithm for BIM data is to improve the robustness of the watermarking to attacks while maintaining the model’s spatial topological relationship. This paper chose the local features of BIM data for watermark information embedding and extraction. To design the relevant watermarking scheme, the following two problems need to be solved: (1) which local features are selected and embedded into the watermark information; (2) how to synchronize the watermark information.

To solve the first problem, we perturb the BIM data’s moveable elements to embed the watermark information. In addition, since some elements of the BIM data are spatially constrained and cannot be moved to achieve watermark embedding, the authors of this paper designed watermark embedding algorithms for immovable elements based on element attributes to increase the number of watermark information carriers. In response to the second question, in order to enhance the robustness of watermarking, the mapping relationship between elements and watermarking bits is established by the high position of the elements’ coordinates so that the relationship between the elements of each water-marking carrier is independent. In summary, this paper uses the spatial location and at-tribute information of the elements as the local features to achieve watermark information embedding and establishes the watermark synchronization relationship based on the elements’ coordinates. Watermark information detection is the inverse process of watermark information embedding.

2.2. Element Perturbation

To increase the robustness of watermarking and to enable the watermarking of movable elements, this paper employs element perturbation based on the work of Qiu [29]. The spatial coordinates are extended from two-dimensional to three-dimensional, allowing for the movement of elements in any direction of the X, Y, and Z axes. With the coordinate accuracy of elements in the BIM data reaching 13 decimal places, precise changes in the coordinates can be made to carry watermark information using the parity of coordinate values. A schematic diagram of element perturbation is shown in Figure 1.

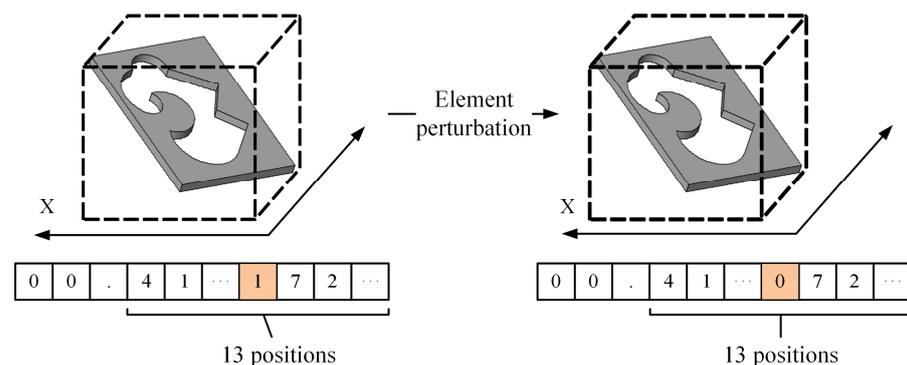


Figure 1. Schematic diagram of element perturbation.

As shown in Figure 1, after the movable elements such as doors, windows, and floors are moved in the X-axis direction, the value of a position in the coordinates changes to correspond to the watermark information of the watermark bit. This approach has minimal impact on the elements and does not easily cause model errors.

However, positioning the points of different elements in BIM data may not be synchronized with the movement of the element after model operations, such as translation and rotation. Therefore, to unify the spatial positioning of the elements and facilitate the accurate embedding and detection of the watermark information, we calculated the centroid of the enclosing box based on the element and used it as the coordinates of the spatial position of the elements. The center point of the enclosing box was calculated as follows:

$$V_i = \left(\frac{Box_i \cdot x_{\max} - Box_i \cdot x_{\min}}{2}, \frac{Box_i \cdot y_{\max} - Box_i \cdot y_{\min}}{2}, \frac{Box_i \cdot z_{\max} - Box_i \cdot z_{\min}}{2} \right) \quad (1)$$

where $i(i = 1, 2, 3 \dots)$ represents the element index. V_i is the coordinate of the element, Box_i represents a collection of the enclosing box coordinate points of the element, and $()_{\max}$ and $()_{\min}$ represent the maximum and minimum values, respectively.

By establishing a unified method for locating the spatial positions of the elements, watermark information can be accurately embedded and detected using element perturbation.

2.3. Embedding Invisible Characters

The use of Unicode encoding as a universal character set encoding standard is widespread in modern computer operating systems. It includes special symbols, some of which are invisible characters that are difficult to detect and do not affect the reading process, making them a suitable candidate for watermarking information steganography [30]. Furthermore, as Unicode characters only occupy two bytes, embedding invisible characters increases the data redundancy by a negligible amount.

BIM data contain rich attribute information, making them suitable for embedding invisible characters into the attributes of immovable elements. This method does not affect the spatial topological relationship of the elements and does not lead to model errors. Moreover, this approach is more secure and has less perceptibility. Furthermore, this method utilizes immovable elements of BIM data as watermark information carriers, which can increase the number of watermark information carriers.

2.4. Watermark Synchronization Mechanism Based on a Mapping Mechanism

To ensure that the mapping relationships of watermark bits for each element remain independent, we propose a synchronization mechanism that maps the watermark positions using the coordinates of elements. Consider that BIM data exhibit agglomeration characteristics, whereby elements located on the same floor or connected to the same wall tend to have similar coordinate values. So, we utilized multiplication operations to extend the range of the figure element coordinate value, followed by a hash calculation to obtain the watermark bit. This approach spreads the element coordinates across a three-dimensional space so that the watermark information becomes more uniformly dispersed within the BIM data. Moreover, it enhances the randomness of the watermark information embedded into the location. A schematic diagram of the proposed watermark synchronization mechanism is shown in Figure 2.

As the accuracy of BIM data coordinates may diminish in practical applications, we constructed the mapping relationship by selecting the high decimal point of the coordinate of the figure element. The watermark index calculation method is shown as follows:

$$Index_i = \text{mod}(\sum \text{mod}(D \times 10^P, 10^S), Length), D \in \{V_i \cdot x, V_i \cdot y, V_i \cdot z\} \quad (2)$$

where $i(i = 1, 2, 3 \dots)$ represents the element index, $Index_i$ represents the index of the watermark corresponding to this element, and V_i is the coordinate. P denotes the lowest valid position after the coordinate decimal point, and S represents the number of required

coordinate positions. $\text{mod}()$ is the remainder calculation, and $Length$ represents the length of the watermark.

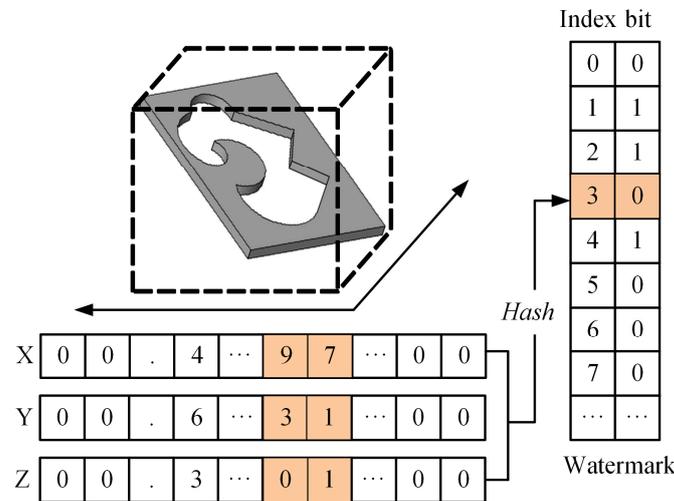


Figure 2. Watermark bit mapping mechanism.

This method ensures that each element’s watermark mapping relationship remains independent, and changes, such as element deletion, movement, rotation, etc., will not affect the mapping of other elements to the correct watermark bit. Thus, it effectively enhances the algorithm’s robustness.

3. Algorithm Implementation

In this paper, we propose a robust watermarking algorithm based on element perturbation and invisible characters, which is specifically designed for BIM data, overcoming the limitations of the existing algorithms for 3D models.

3.1. Watermark Generation

We employed a meaningful watermark that can provide more intuitive experimental results than meaningless ones. Considering that meaningful watermarks are typically easier to crack, we preprocessed the original watermark sequence W read by scanning using a disruption function L , defined by the following equation:

$$W' = L(W, Key) \tag{3}$$

where W' is the unordered watermark sequence, L is a logistic chaotic system, which is non-periodic and hard to decipher, and can be used for watermark decorrelation [31]. Key is the key determined by the watermark embedder. In a logistic chaotic system, the chaotic sequence generation is formulated as follows:

$$C_{n+1} = u \times C_n \times (1 - C_n) \tag{4}$$

where $n(n = 1, 2, 3 \dots)$ is the index of the chaotic sequence, $C_n(C_n \in [0, 1])$ represents the value of the chaotic sequence, and C_1 and u are set by Key . Specifically, when $3.57 \leq u \leq 4$, the logistic chaotic system is in a fully chaotic state.

Based on the above process, the original watermark image is shown in Figure 3a, and the scrambled image is depicted in Figure 3b.

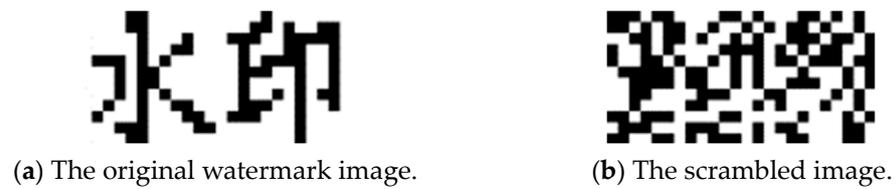


Figure 3. Watermarked images before and after disarray.

3.2. Watermark Embedding

The watermark information embedding process is shown in Figure 4.

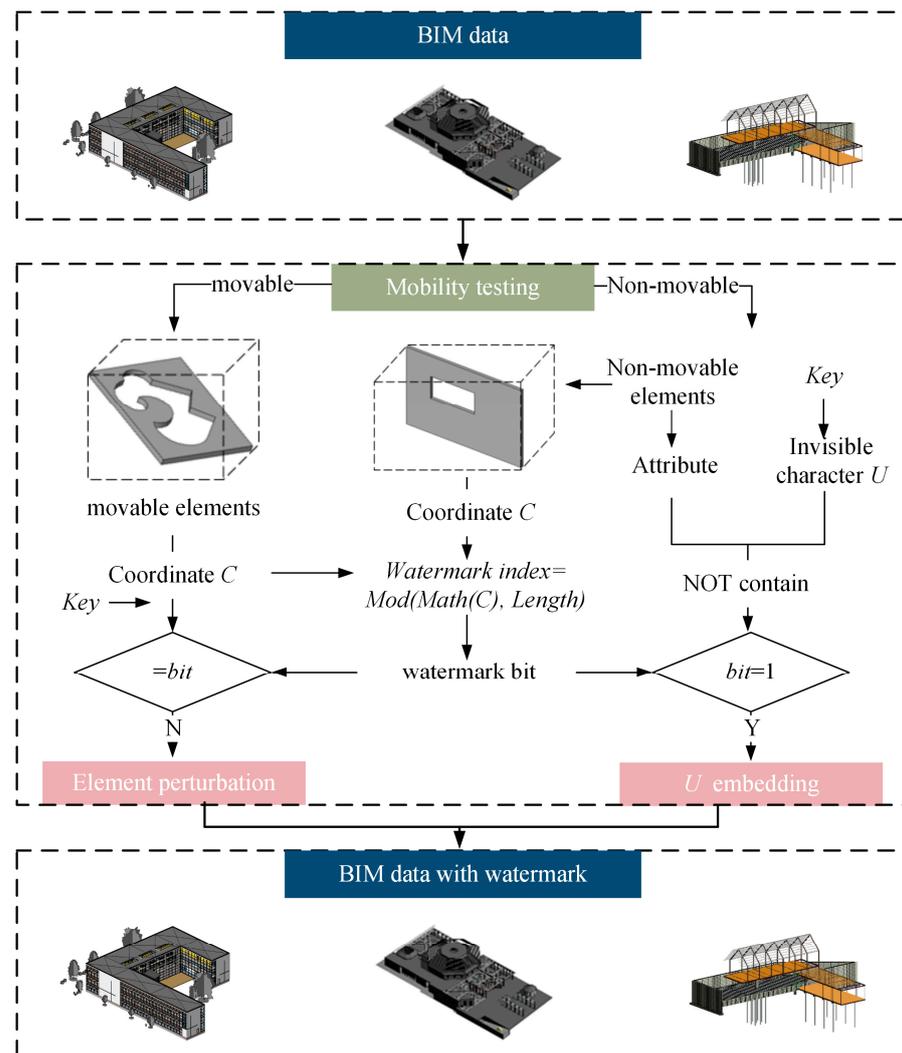


Figure 4. The watermark information embedding process.

The watermark embedding steps are as follows:

Step 1: Firstly, considering element addition and deletion in the BIM data, the stable structure elements, such as curtain walls, floor slabs, and walls, in the BIM data are identified for use as watermark carriers. The center point V_i of the enclosing box of each element is calculated using Equation (1) in Section 2.2, and added to the set V , expressed as $V = \{V_1, V_2, \dots, V_n\}$, $V_i = (X_i, Y_i, Z_i)$, where n is the total number of watermark carrier elements. Secondly, taking into account the loss of coordinate accuracy and the aggregation characteristics of the BIM data, to establish the mapping relationship of the watermark for each element, a high decimal point of the coordinates of elements is selected according to Equation (2) in Section 2.4.

Finally, the elements are tested for movement, and the movable elements are added to the set S_1 , denoted as $S_1 = \{V_1, V_2, \dots, V_N\}$, where N is the number of movable elements. The center points of the immovable elements are added to the set S_2 , denoted as $S_2 = \{V_1, V_2, \dots, V_M\}$, where M is the number of immovable elements. The sum of M and N is the total number of elements n .

Step 2: The movable elements are perturbed in the X-axis directions to represent the watermark information using the parity of the X coordinates of the elements. To maintain the integrity and usability of the BIM data, perturbing elements in the Z-axis direction is avoided, as this may cause model generation errors. Firstly, the coordinate position is determined according to *Key*, and secondly, all the elements in the set S_1 are traversed and perturbed according to Equation (5), which specifies the perturbation method. The elements are perturbed when the calculated value of the coordinate setting position does not match with the corresponding watermark information bit in order to embed the watermark information.

$$X'_i = \begin{cases} X_i, & \text{mod}(X_i \times 10^k, 2) = W_{Index_i} \\ X_i + 10^{-k}, & \text{mod}(X_i \times 10^k, 2) \neq W_{Index_i}, W_{Index_i} = 1 \\ X_i - 10^{-k}, & \text{mod}(X_i \times 10^k, 2) \neq W_{Index_i}, W_{Index_i} = 0 \end{cases} \quad (5)$$

where X'_i denotes the coordinates of the element in S_1 after embedding the watermark information, k is the watermark information embedding location, and $Index_i$ and W_{Index_i} denote the watermark index and watermark bit, respectively.

Step 3: The invisible characters are embedded into the immovable image elements of set S_2 . Before embedding the invisible characters, the invisible character code U is set to be embedded, and the attribute field of the element is set according to *Key*. Each immovable image element is iterated in the set S_2 , and the invisible character is embedded according to Equation (6) if the corresponding watermark bit is 1 and the attribute field of the image element does not already contain the invisible character U . If the attribute field of the image element already contains the invisible character U , no operation is performed.

$$C'_j = C_j + U \quad (6)$$

where C_j is the original attribute field of the element in the set S_2 , C'_j is the attribute field after embedding invisible characters, and U is the invisible character.

3.3. Watermark Extraction

The process of watermark information detection is illustrated in Figure 5.

Watermark extraction is the inverse process of watermark embedding. Firstly, the BIM data are preprocessed as described in Section 3.2 to obtain the sets of movable elements S_1 and immovable elements S_2 . The watermark information is then extracted based on the local features of the BIM data. The extraction steps are as follows:

Step 1: The elements are iterated in the set S_1 , and the coordinate position carrying the watermark information is obtained based on *Key*. The watermark information is extracted using Equation (7), I'_{Index_i} is updated with the detected watermark value, and the number of times the watermark value is written at that location is increased.

$$I'_{Index_i} = \begin{cases} 0 + I_{Index_i}, & \text{mod}(X_i \times 10^k, 2) = 0 \\ 1 + I_{Index_i}, & \text{mod}(X_i \times 10^k, 2) = 1 \end{cases} \quad (7)$$

where $Index_i$ is the index of the sequence corresponding to the element, and I_{Index_i} and I'_{Index_i} are the number of times this location maps a watermark value of 1 before and after the update, respectively. X_i denotes the coordinates with watermark information, and k is the watermark information embedding location.

Step 2: The invisible characters U and the attributes of the elements carrying watermark information are obtained based on *Key*. Then, the elements are iterated in the set S_2 , and the watermark information is extracted using Equation (8). The detected watermark

value updates I'_{Index_i} , and increases the number of times the watermark value is written at that location is increased.

$$I'_{Index_j} = \begin{cases} 0 + I_{Index_j}, & C_j \text{ contains } U \\ 1 + I_{Index_j}, & C_j \text{ contains } U \end{cases} \quad (8)$$

where $Index_i$ is the index of the sequence corresponding to the element, and I_{Index_i} and I'_{Index_i} are the number of times this location maps the invisible character U before and after the update, respectively.

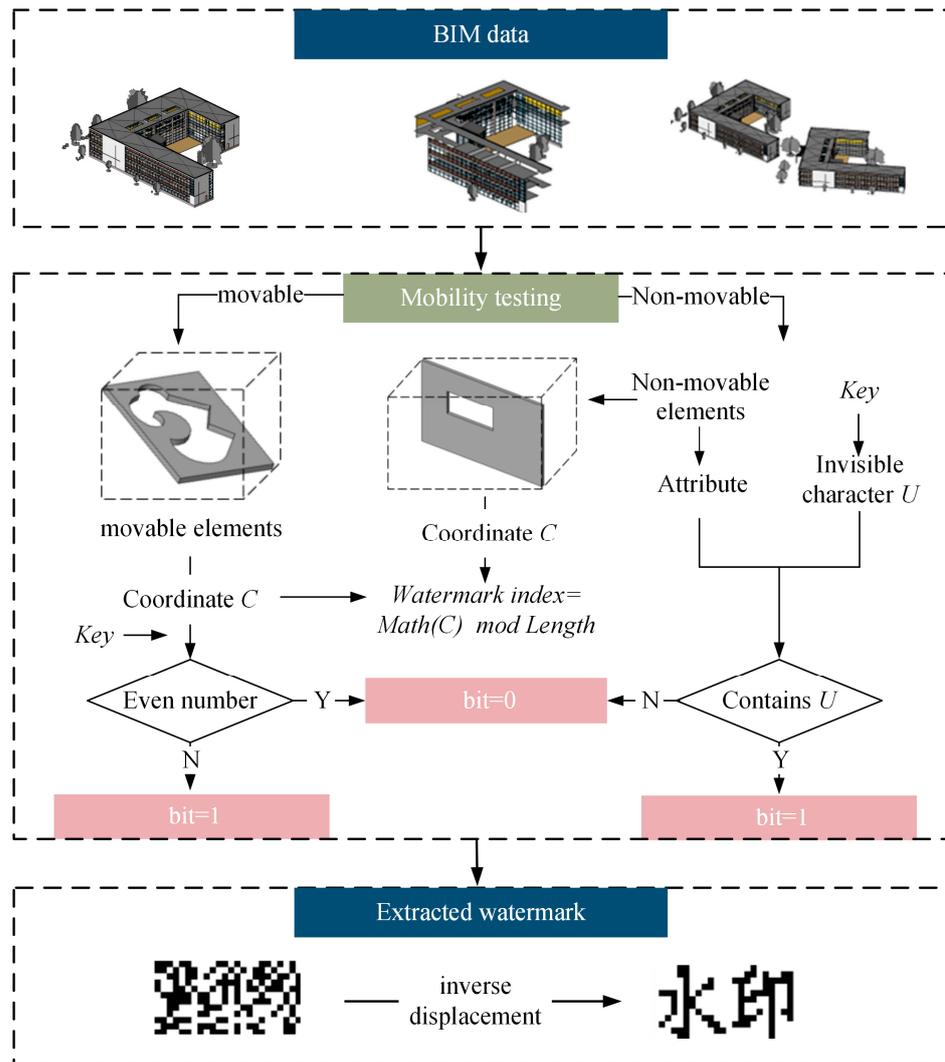


Figure 5. The watermark information extracting process.

Step 3: Based on the majority principle, all watermark bits are traversed and the watermark information is updated using Equation (9) to obtain the watermark sequence I . Then, the one-dimensional watermark sequence I is dimensioned and inverted to obtain the BIM data watermark information image.

$$I_i = \begin{cases} 0, & \frac{I_i}{Sum_i} > 0.5 \\ 1, & \frac{I_i}{Sum_i} \leq 0.5 \end{cases} \quad (9)$$

where $i(i = 1, 2, 3 \dots)$ represents the index of a watermark sequence and Sum_i denotes the number of times the watermark information is written to this index of sequence I .

Here, the majority principle is used to judge the watermarking bit and watermarking information, making it resistant to changes in the spatial location or attribute information of some elements.

4. Experiments and Results

To evaluate the effectiveness of our algorithm, we selected three models of RVT format, which is the mainstream BIM format, including two building models and a structural model, as shown in Figure 6. The basic information of the BIM experimental data is shown in Table 1.

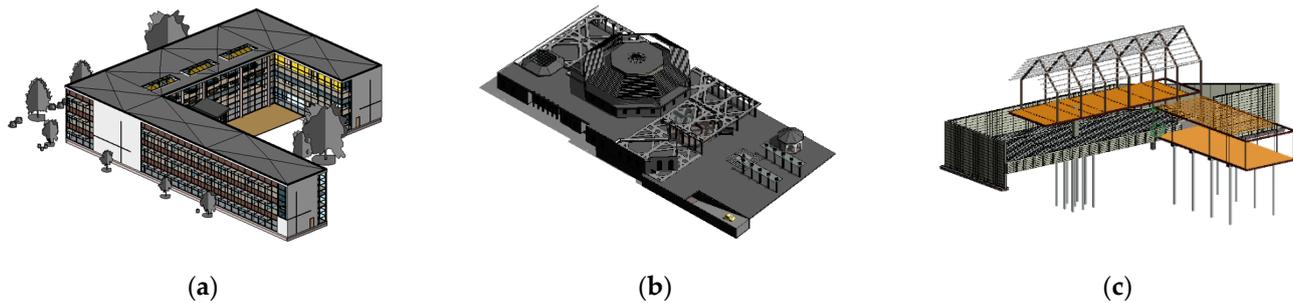


Figure 6. Experimental models. (a) Office building. (b) Gymnasium. (c) Structural model.

Table 1. The basic information of BIM experimental data.

| BIM Data | Size (MB) | Number of Elements |
|------------------|-----------|--------------------|
| Office building | 13.65 | 5841 |
| Gymnasium | 17.36 | 848 |
| Structural model | 5.54 | 844 |

Our experiments and analysis focused on imperceptibility, the original data file size change, and robustness.

4.1. Imperceptibility Analysis

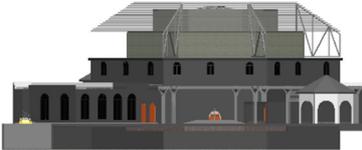
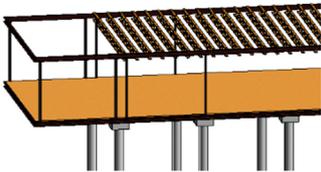
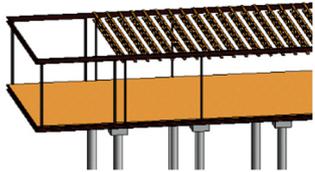
To achieve the objective of hiding watermark information in the BIM data, it is essential that the data remain visually intact before and after watermark embedding. Hence, we designed experiments to analyze the imperceptibility of watermark embedding qualitatively and quantitatively.

4.1.1. Qualitative Analysis

We present a comparison of the details of the BIM experimental data before and after watermark embedding in Table 2.

Subjective visual observation shows no significant difference between the architecture and structure before and after watermark embedding, indicating that the embedding of the digital watermark does not affect the visual quality of the original data. Additionally, the watermarking algorithm presented in this paper has good imperceptibility in terms of invisible character embedding, as they cannot be observed in the graphical element attribute fields after opening the software interface.

Table 2. Comparison of model details before and after watermark embedding.

| BIM Data | Before Embedding | After Embedding |
|------------------|---|---|
| Office building |  |  |
| Gymnasium |  |  |
| Structural model |  |  |

4.1.2. Quantitative Analysis

To quantitatively analyze the imperceptibility of the watermarked information, we selected the Hausdorff distance [32], signal-to-noise ratio (SNR), and peak signal-to-noise ratio (PSNR) as the evaluation metrics [33]. The Hausdorff distance is a method to calculate the degree of similarity between two sets. The smaller its value, the higher the degree of similarity before and after embedding watermarks into BIM data, indicating the better imperceptibility of the digital watermarking. SNR and PSNR are parameters to measure the degree of influence of watermarking on the original data, considering the original element position and the position after watermark embedding as the normal signal and noise. The higher the values of SNR and PSNR, the less the watermark information affects the original data, indicating better imperceptibility. In BIM data, SNR and PSNR are calculated as shown in Equations (10) and (11).

$$SNR(A, B) = 10 \times \lg \frac{\sum_{i=1}^n x_{Ai}^2}{\sum_{i=1}^n (x_{Ai} - x_{Bi})^2} \tag{10}$$

where $SNR(A, B)$ represents the SNR value of the BIM data after the watermark embedding, A is the original BIM data, and B is the data embedded with the watermark. x_{Ai} and x_{Bi} are the value of the element coordinates of the original BIM data and watermarked BIM data, respectively. n is the number of BIM data elements.

$$PSNR(A, B) = 10 \times \lg \frac{n \times \max(x_{Ai})^2}{\sum_{i=1}^n (x_{Ai} - x_{Bi})^2} \tag{11}$$

where $PSNR(A, B)$ represents the PSNR value of the BIM data after the watermark embedding, A is the original BIM data, and B is the data embedded with the watermark. n is the number of BIM data elements and $\max(x_{Ai})$ is the maximum value of the x -coordinate of the elements.

The results of the calculation of these three variables complement each other and can fully illustrate the imperceptibility of the algorithm. The experimental results of the above three indicators are shown in Table 3.

Table 3. Results of quantitative analysis of imperceptibility.

| BIM Data | Hausdorff (m) | SNR (dB) | PSNR (dB) |
|------------------|-----------------------|----------|-----------|
| Office building | 2.24×10^{-4} | 127.11 | 114.15 |
| Gymnasium | 2.56×10^{-5} | 118.44 | 111.29 |
| Structural model | 4.32×10^{-5} | 103.49 | 107.09 |

Table 3 shows the experimental results of the above three indicators. In general, the maximum error of the electronic map feature point plane is ± 1.2 m. After embedding the watermark, the maximum Hausdorff distance in the three BIM datasets was 0.000224 m, which is much lower than the feature point plane position error. Hence, the perturbation range of this algorithm map element is small. The SNR and PSNR of the experimental data are higher than 100 dB, indicating that the embedding of the watermark causes very little noise in the BIM data. In summary, the watermark had minimal impact on the original BIM data, and the imperceptibility was excellent.

4.2. Analysis of File Size Change

Although the invisible characters are not visually perceptible, embedding them as watermark information can increase the size of the BIM data due to the additional characters. This increase in size can result in information redundancy, which may reduce the usefulness of the model, especially in BIM data transfer, use, and copy scenarios. Thus, it is important to evaluate the file size change resulting from the watermark embedding. To quantify this, we calculated the ratio of the incremental size of the redundancy to the original data size as a quantitative index, which is shown in Equation (12).

$$Ratio(A, B) = \frac{Size(B) - Size(A)}{Size(A)} \times 100\% \quad (12)$$

where $Ratio(A, B)$ is the data size change ratio. A and B are the original BIM data and the BIM data after watermark embedding, respectively. $Size()$ is the data size. After fully analyzing the actual application scenarios of BIM data, we can usually consider that when the change in the size of the BIM data after certain operations is less than 0.5%, it can be assumed that these operations have almost no effect on the size of BIM data, i.e., they do not affect the use of the data.

The result of the file size transformation is shown in Table 4 and Figure 7.

As presented in Table 4, the redundancy increment ratios for the gymnasium and structural model were both less than 0.1%, owing to the high proportion of movable elements in the models, which resulted in a low embedding of invisible characters. In contrast, the majority of the elements in the office building were immovable due to restrictions on the elevation and neighboring elements, resulting in a high proportion of embedded invisible characters and a relatively high redundancy increment ratio of 0.14%. Overall, the redundancy increment ratio for all three watermarked BIM data files was less than 0.5%, suggesting that the increase in BIM data redundancy after watermark embedding was small, and that the imperceptibility of the watermarking algorithm was excellent. The embedding of watermarks has essentially no impact on the BIM data's size or application.

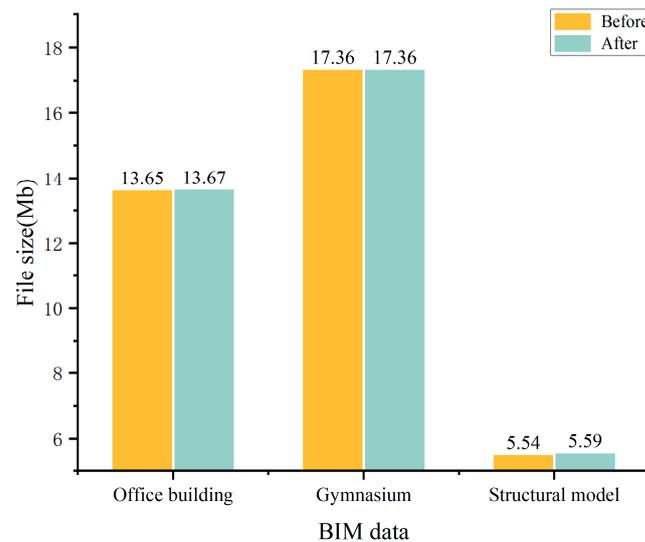


Figure 7. File increment comparison.

Table 4. Results of redundancy increment.

| BIM Data | Redundancy (Mb) | Ratio (%) |
|------------------|-----------------------|-----------|
| Office building | 0.02 | 0.14 |
| Gymnasium | 0.39×10^{-2} | 0.02 |
| Structural model | 0.05 | 0.09 |

4.3. Robustness Analysis

4.3.1. Evaluation Indicator of Experimental Results

In the distribution and use of BIM data, unintentional or malicious attacks may occur, necessitating the extraction of watermark information from the model for data copyright determination. To evaluate the similarity between the extracted and claimed watermark information, this study introduces normalized correlation (NC) as a quantitative index. The extracted watermark information is compared with the original watermark information, and the NC calculation can be summarized as the ratio of the number of watermark bits with correct information extracted to the total number of watermark bits. An NC value of 1 indicates complete extraction of the correct watermark, while a value of 0 indicates no extraction at all. The higher the coefficient, the higher the watermark extraction accuracy. Different usage scenarios have different requirements for the NC value. In this study, we set the threshold value at 0.75, and considered the watermark extraction successful when the NC coefficient was greater than the threshold value.

4.3.2. Watermark Extraction Results

After embedding the watermark information into the experimental BIM data, watermark extraction and detection were performed to check whether the watermark was successfully embedded. When the watermark data are not subjected to any attack, the watermark should be detected completely, i.e., the NC should be 1. The watermark detection was performed without attacking the BIM data, and the results are presented in Table 5.

As shown in Table 5, the NC values for all three BIM data watermark extractions were 1. This indicates that after the pre-element screening and embedding condition judgment, the watermark information was correctly embedded into the carrier and extracted completely via the blind detection process.

Table 5. Watermark information detection results.

| BIM Data | NC Value | Extracted Watermark |
|------------------|----------|---------------------|
| Office building | 1.00 | 水印 |
| Gymnasium | 1.00 | 水印 |
| Structural model | 1.00 | 水印 |

4.3.3. Robustness of Translation Attack

Translation is a common operation for BIM data. In actual use, the data often face two kinds of translation attacks: (1) In BIM + GIS application scenarios, BIM data follow the overall translation of the project base point in the map. This does not affect the project coordinate system, so the coordinates of the model within the project do not change, and the watermark extraction is not affected. (2) In the overall translation of the model within the project relative to the project base point, such a translation attack will not only lead to a change in the coordinates of each element within the model but will also cause the automatic deletion of some elements, leading to noise in the extracted watermark image. Therefore, to evaluate the robustness of the algorithm against this kind of translation attack, the authors in this paper designed a series of experiments with intervals of 100 m, and the results are shown in Table 6 and Figure 8.

Table 6 and Figure 8 show that the watermark information can still be extracted more completely after translation along the X, Y, and Z axes. The watermark sequence's NC was higher than 0.95, which is much higher than the threshold value of 0.75 set in this study. This is because the overall translation of the data led to a change in the part before the decimal point of the element coordinates, while the algorithm in this paper carried the watermark information with the part after the decimal point of the coordinates to establish the watermark bit mapping relationship. Furthermore, the algorithm employs the majority principle to determine the watermark value, so local changes, such as the deletion of elements caused by the translation, may not significantly affect the final watermark sequence. In conclusion, the algorithm proposed in this paper exhibited strong robustness against translation attacks on BIM data.

Table 6. Results of translation attacks.

| BIM Data | Direction | Attack Distance (m) | NC Value |
|------------------|-----------|---------------------|----------|
| Office building | X/Y/Z | 100 | 0.99 |
| | | 200 | 0.99 |
| | | 300 | 0.99 |
| Gymnasium | X/Y/Z | 100 | 0.98 |
| | | 200 | 0.98 |
| | | 300 | 0.98 |
| Structural model | X/Y/Z | 100 | 1 |
| | | 200 | 1 |
| | | 300 | 1 |

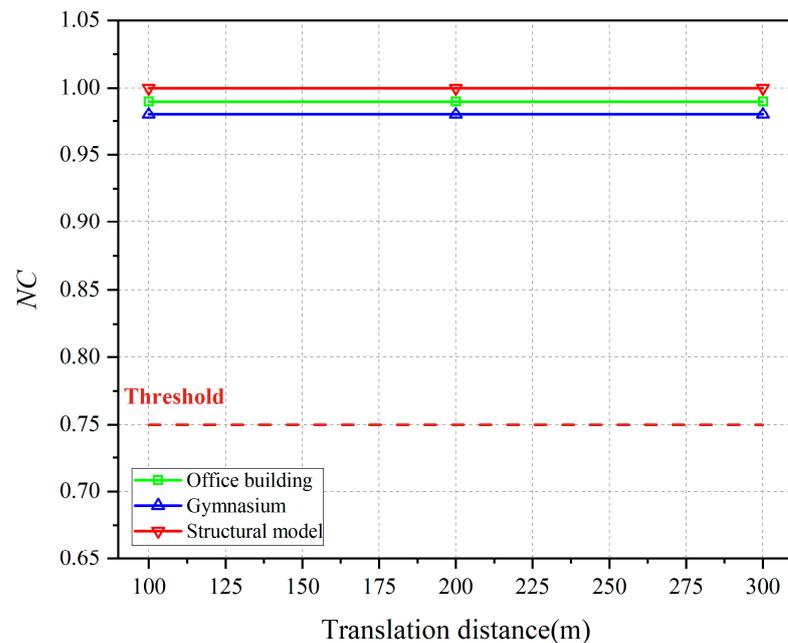


Figure 8. Results of translation attacks.

4.3.4. Robustness of Element Attacks

In the actual application operation of BIM data, the deletion and addition of elements are common types of attacks that BIM data may face. This paper establishes a mapping relationship between each element's coordinate value and the algorithm's watermark bit, allowing each element's watermark embedding process to be independent. Therefore, deleting or adding certain elements does not significantly affect the detection of watermark information of other watermark carriers. Moreover, the algorithm employs a voting mechanism to extract watermark information, enabling it to tolerate minor changes in figure elements that do not impact the final watermark sequence.

However, BIM data may automatically update, leading to changes in the spatial location of adjacent elements due to the deletion or addition of elements, which can result in incorrect watermark extraction. To evaluate the algorithm's resistance to element deletion attacks, we conducted experiments with a 10% element deletion and addition interval and generated the corresponding BIM models shown in Figure 9. Table 7 and Figure 10 present the experimental results for different pixel deletion rates.

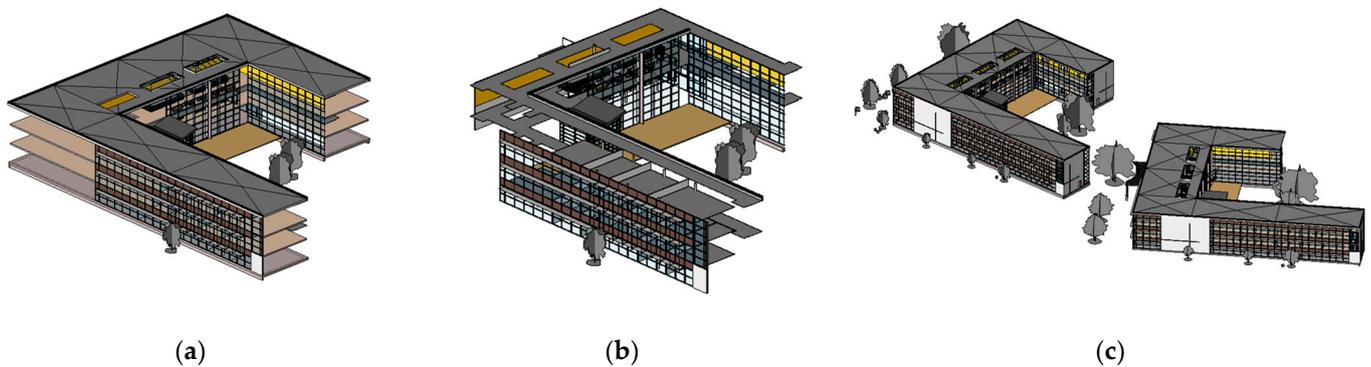


Figure 9. Partial model after element attack. (a) Deletion ratio: 40%. (b) Deletion ratio: 60%. (c) Addition ratio: 60%.

Table 7. Results of element attack.

| Attack Type | Attack Ratio (%) | NC Value | | |
|------------------|------------------|-----------------|-----------|------------------|
| | | Office Building | Gymnasium | Structural Model |
| Element Deletion | 10 | 1.00 | 1.00 | 1.00 |
| | 20 | 1.00 | 0.99 | 1.00 |
| | 30 | 1.00 | 0.97 | 0.98 |
| | 40 | 1.00 | 0.97 | 0.98 |
| | 50 | 1.00 | 0.96 | 0.95 |
| | 60 | 0.99 | 0.91 | 0.93 |
| Element Addition | 10 | 1.00 | 1.00 | 1.00 |
| | 20 | 1.00 | 0.98 | 0.97 |
| | 30 | 1.00 | 0.97 | 0.95 |
| | 40 | 0.98 | 0.96 | 0.95 |
| | 50 | 0.98 | 0.95 | 0.92 |
| | 60 | 0.96 | 0.92 | 0.89 |

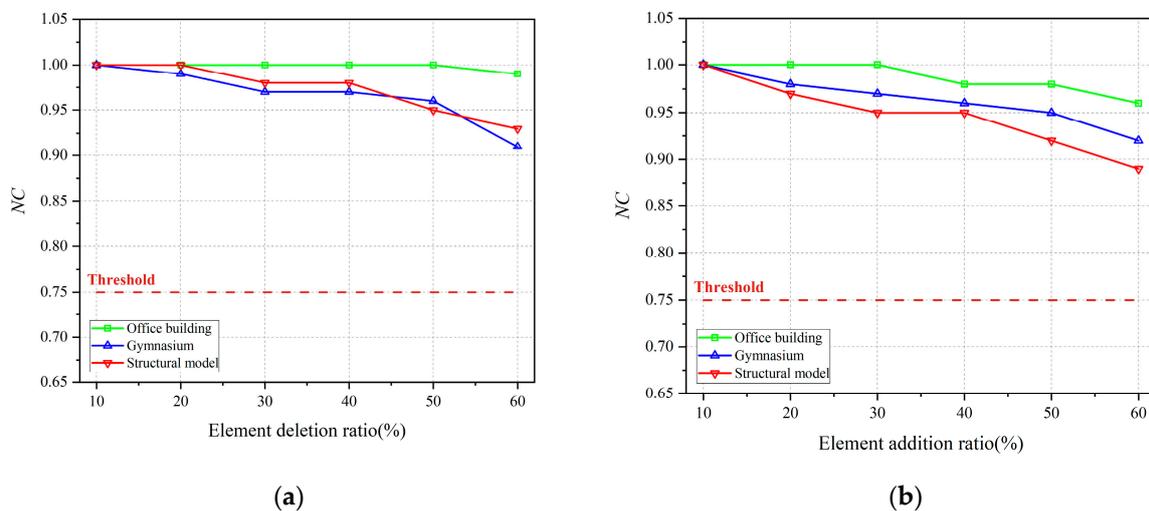


Figure 10. Results of element attack. (a) Element deletion attack. (b) Element addition attack.

Our analysis of Table 7 indicates that deleting and adding elements within a 10% range had no effect on extracting the watermark information from the three models. The watermark extraction of the office building remained unaffected by deletion attacks with 20% and 30% ratios, while the watermark information of the gymnasium and structure model displayed a small amount of noise but could still represent the copyright information completely. As the office building model had significantly more elements, even when subjected to 40% and 50% element deletion, the watermark information could still be extracted entirely. However, the gymnasium model and the structure model had fewer total elements, resulting in a lower NC value and slightly noisier watermark information compared to the office building. When 60% of the elements were deleted, the NC values of the three BIM models were higher than 0.90, except for the structure model under an attack of element addition. The overall analysis shows that the quality of the extracted watermark information from the BIM data decreased gradually with the increasing intensity of the element deletion and addition. However, the NC value of the watermark was higher than 0.89, significantly surpassing the threshold value of 0.75 required to display watermark information adequately. Therefore, this paper’s algorithm exhibited strong robustness against element attacks, meeting the daily use needs.

4.3.5. Robustness of Change in Level of Detail

Elements in BIM data contain point, line, and surface information. From the same perspective, adjusting the level of detail settings can alter the display form of element

points, lines, and surfaces, resulting in different geometric representations of building structures. In real-world scenarios, such changes in the level of detail can occur, which may affect the accurate extraction of the watermark information. To evaluate the algorithm's robustness against this type of attack, we conducted three experiments using BIM data with detailed, medium, and coarse levels of detail. The experimental results are presented in Table 8 and Figure 11.

Table 8. Results of changes in the level of detail.

| BIM Data | Level of Detail | NC Value |
|------------------|-----------------|----------|
| Office building | Detailed | 1.00 |
| | Medium | 1.00 |
| | Rough | 1.00 |
| Gymnasium | Detailed | 1.00 |
| | Medium | 1.00 |
| | Rough | 1.00 |
| Structural model | Detailed | 1.00 |
| | Medium | 1.00 |
| | Rough | 1.00 |

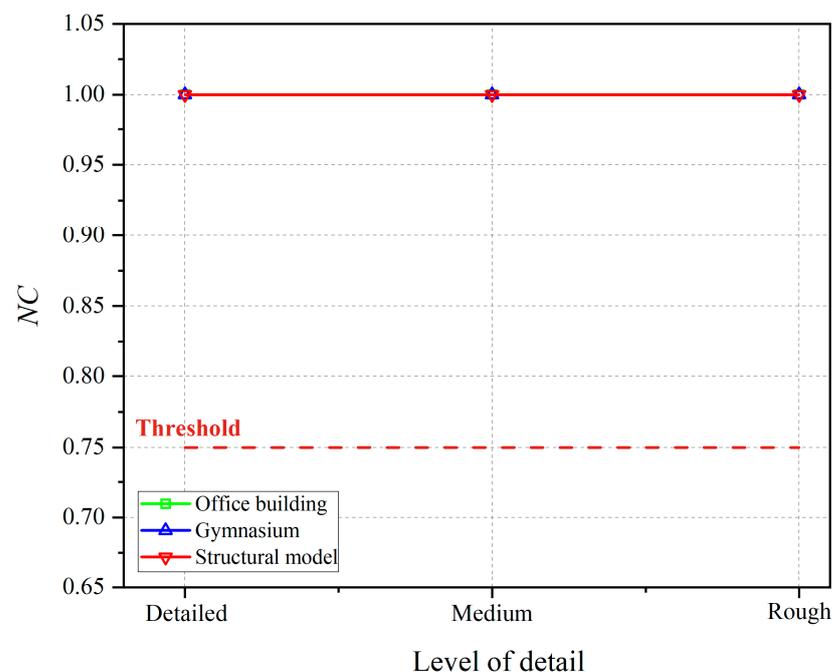


Figure 11. Results of changes in level of detail.

As shown in Table 8, changing the level of detail from detailed to coarse did not affect the algorithm's ability to extract watermark information from the BIM data. This is because the algorithm mapped the watermark bits to the center point of the bounding box of the image element and embedded watermark information using spatial location and attribute information. Changes in the level of detail only affected the morphological details of the element and did not impact the watermark bits or the watermark information embedded into the element. Therefore, the algorithm presented in this paper was resistant to changes in level of detail and could effectively extract watermark information from BIM data under different detail level settings.

4.3.6. Robustness against Geometry–Property Separation

In practice, BIM data are often parsed and converted into various formats to facilitate their use on different technical platforms. In BIM, the practice of storing geometric models

and attributes separately is referred to as geometry–property separation. This process typically involves geometry–property separation, where geometric and attribute information is classified and managed in different formats. However, this separation can potentially affect the watermark information carried by the data. To evaluate the robustness of our watermarking algorithm against geometry–property separation attacks, we conducted experiments in which BIM data in RVT format were converted into FBX geometry format and JSON text format, and watermark extraction was performed on the two data formats separately. The experimental results are presented in Table 9.

Table 9. Results of geometry–property separation.

| BIM Data | Format | Type | NC Value |
|------------------|--------|-----------------|----------|
| Office building | FBX | Geometry format | 1.00 |
| | JSON | Text format | 1.00 |
| Gymnasium | FBX | Geometry format | 0.99 |
| | JSON | Text format | 0.98 |
| Structural model | FBX | Geometry format | 0.98 |
| | JSON | Text format | 1.00 |

As shown in Table 9, the watermark extraction of the office building model was not affected by the number–mode separation, and the watermark extraction of the gymnasium and structure models showed only slight effects. Nonetheless, both the watermark NC values were higher than 0.98, which significantly exceeds the threshold value of 0.75, indicating that the watermark information was essentially complete. Therefore, our algorithm exhibited superior robustness against geometry–property separation attacks on BIM data and could ensure the integrity of the watermark information. Furthermore, the experimental results illustrate that the proposed algorithm is not only resistant to such attacks, but also applicable to data with only geometric information or only attribute information, such as building management models.

5. Discussion

The experimental results presented in Section 4 demonstrate that the proposed watermarking algorithm, which is based on element perturbation and invisible characters, can provide a feasible watermarking scheme for BIM data. The advantages of this method in this work are: (1) The combination of element perturbation and invisible character embedding methods enhances the robustness and imperceptibility of the watermark. (2) The robustness of the watermark is further enhanced by a mapping mechanism. This effectively solves the challenge of applying existing 3D robust watermarking techniques to BIM data. In order to provide a better understanding of the proposed method, we offer further discussion in the following three areas.

5.1. Analysis of the Watermark Synchronization Mechanism

The watermark algorithm proposed in this paper establishes a many-to-one synchronization mechanism based on the coordinates of the elements, allowing each bit of watermark information to be embedded into the BIM data multiple times. This redundancy enables the extraction of the watermark to be based on the majority principle for voting. In the event that elements in the BIM data undergo deletion or addition attacks, the possibility of errors caused by changes to these elements is reduced, and the probability of extracting complete and correct watermark information is significantly improved. For example, a watermark bit has N watermark carriers, and if n ($n < 0.5 \times N$) of the elements change, resulting in the loss of the watermark information, there are still more than 50% of the elements pointing to the correct information, ensuring that the extracted watermark sequence is correct. This concept is validated by the element deletion and element addition experiments in Section 4. Given that the elements of BIM data are relatively concentrated on each

floor, this paper proposes that the watermark synchronization mechanism is optimized by the multiplication principle, thereby embedding the watermark information into the data more uniformly.

Moreover, this paper employs the center point of the bounding box as the coordinates of the figure elements, and this calculation method is applicable to elements that are positioned according to points, lines, and surfaces. This approach standardizes the spatial positioning of BIM data, thereby enhancing the stability of the watermark bit mapping.

5.2. Analysis of Algorithm Applicability

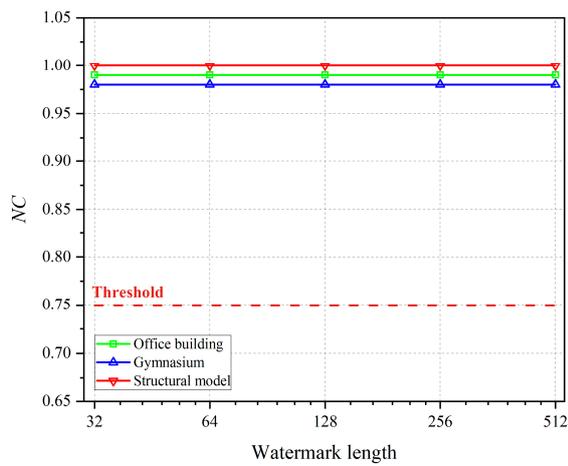
In the proposed watermarking algorithm, watermark information is embedded based on the mobility analysis, with movable elements subjected to small-scale perturbations, and immovable elements carry invisible characters. This method can be used for all the elements in the model, whether the main elements, such as walls, or elements such as doors and windows. As a result, the proposed method can be applied to any type of BIM data, as demonstrated in the building model, office building, and structural model presented in Section 4. In conclusion, the method has an extremely wide range of applications.

It is worth noting that the correct mapping of watermark bits may be affected when BIM data are rotated and the X and Y coordinates of all the elements in the model are changed, particularly for BIM data with a high proportion of movable elements. Therefore, while this method is widely used in practice, there are still some directions that need further research in the future, including rotation robustness enhancement, redundancy reduction, and combination with other watermarking algorithms.

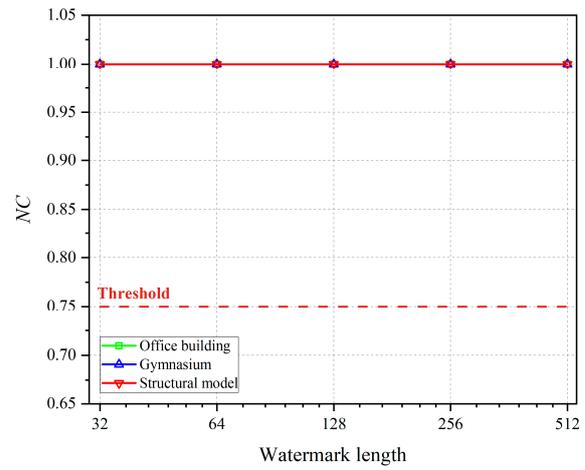
5.3. Analysis of Watermarking Capacity

The watermark capacity represents the maximum amount of a watermark that can be embedded into data. In this paper, we used the coordinates and attributes of elements as embedding targets. Movable elements can carry two bits of watermark information, whereas immovable elements can carry at least one bit of watermark information. Therefore, the watermark capacity of our proposed method is theoretically larger than the total number of elements in BIM data. However, some unstable elements are unsuitable as watermark carriers, and it is necessary to ensure that multiple elements correspond to one watermark bit. Thus, the watermarking capacity is generally smaller than the total number of elements. In this section, we examine the relationship between the watermark length and the robustness in our proposed watermarking algorithm. Using the BIM data from Section 4 as experimental data, we conducted watermarking experiments on model translation, detail level change, graphical element, and numerical model separation attacks at watermark lengths of 32, 64, 128, 256, and 512. Table 10 and Figure 12 show the experimental results.

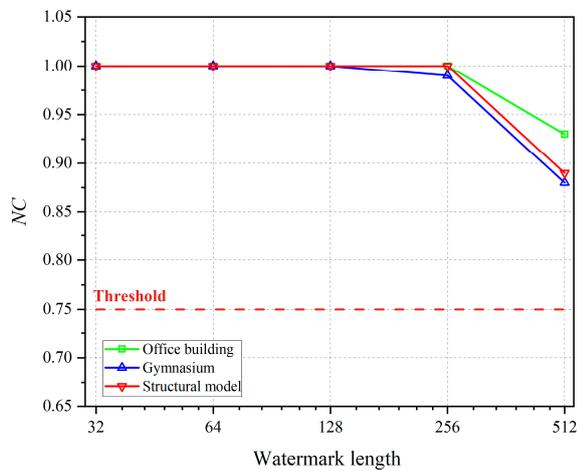
In Table 10 and Figure 12, it can be observed that the robustness of the watermark against element addition and deletion as well as pattern separation attacks decreased continuously as the watermark length increased from 32 to 512, except for translation and detail level change attacks, which did not affect the watermark extraction. Once the watermark length exceeded 50% of the total number of elements of the gymnasium and structure model, which occurred at a length of 512 bits, the ability of the watermark to resist various attacks was significantly reduced. Specifically, when encountering element deletion, the proportion of the extracted watermark was only 20%, and the NC value almost approached the threshold value set in this paper of 0.75. Thus, to ensure the robustness of the watermark algorithm, it is crucial to limit the length of the watermark sequence to be less than the total number of elements of the BIM data. In this study, a binary sequence of 288 bits was used as the watermark, indicating that to fully embed the BIM data and achieve high robustness, it should contain at least 576 elements. Therefore, the watermark capacity of the proposed method needs to be enhanced, and further investigation will be conducted in future research.



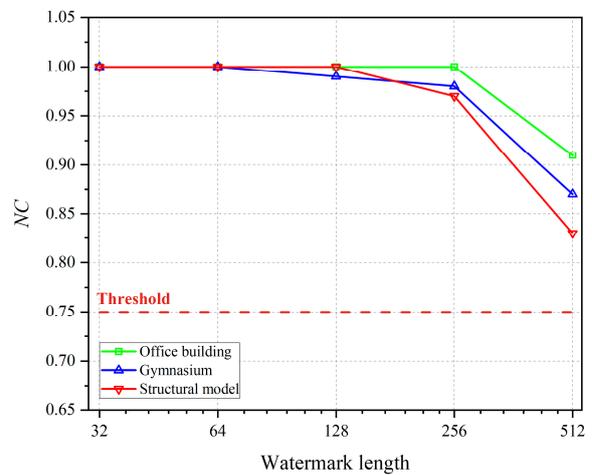
(a)



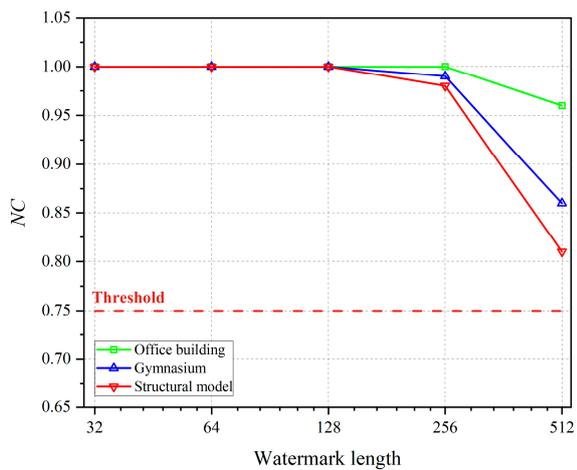
(b)



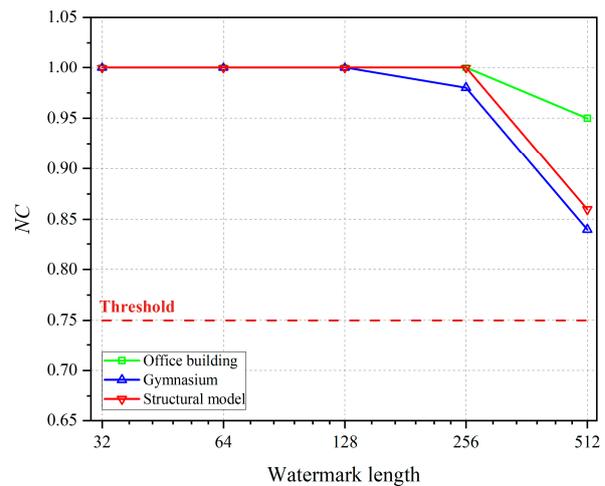
(c)



(d)



(e)



(f)

Figure 12. Results of attacks under different watermark lengths. (a) Results of 100-m translating attack. (b) Results of changing level of detail. (c) Results of deleting 20% of elements. (d) Results of adding 20% of elements. (e) Results of FBX format data. (f) Results of JSON format data.

Table 10. Results of attacks under different watermark lengths.

| BIM Data | Attacks | NC with Different Watermark Lengths | | | | |
|------------------|----------------------------|-------------------------------------|------|------|------|------|
| | | 32 | 64 | 128 | 256 | 512 |
| Office building | Translate 100 m | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 |
| | Change level of detail | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | Delete 20% of elements | 1.00 | 1.00 | 1.00 | 1.00 | 0.93 |
| | Add 20% of elements | 1.00 | 1.00 | 1.00 | 1.00 | 0.91 |
| | Exporting FBX format data | 1.00 | 1.00 | 1.00 | 1.00 | 0.96 |
| | Exporting JSON format data | 1.00 | 1.00 | 1.00 | 1.00 | 0.95 |
| Gymnasium | Translate 100 m | 0.98 | 0.98 | 0.98 | 0.98 | 0.98 |
| | Change level of detail | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | Delete 20% of elements | 1.00 | 1.00 | 1.00 | 0.99 | 0.88 |
| | Add 20% of elements | 1.00 | 1.00 | 0.99 | 0.98 | 0.87 |
| | Exporting FBX format data | 1.00 | 1.00 | 1.00 | 0.99 | 0.86 |
| | Exporting JSON format data | 1.00 | 1.00 | 1.00 | 0.98 | 0.84 |
| Structural model | Translate 100 m | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | Change level of detail | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | Delete 20% of elements | 1.00 | 1.00 | 1.00 | 1.00 | 0.89 |
| | Add 20% of elements | 1.00 | 1.00 | 1.00 | 0.97 | 0.83 |
| | Exporting FBX format data | 1.00 | 1.00 | 1.00 | 0.98 | 0.81 |
| | Exporting JSON format data | 1.00 | 1.00 | 1.00 | 1 | 0.86 |

6. Conclusions

This paper presents a digital watermarking algorithm for protecting the copyright of BIM data. The algorithm is based on the local features of the model, which is characterized by adding little redundancy, not changing the original topological relationships of the BIM data, and ensuring the availability of the data. In the proposed method, movable element perturbation and invisible character embedding of non-movable elements are combined to expand the number of watermark carriers, and achieve multiple embedding of digital watermark information. In addition, a mapping mechanism was used to further increase the watermark robustness. The experimental results show that the algorithm has good imperceptibility, and the embedding of invisible characters adds little redundancy. The proposed algorithm is robust to translation, element deletion, element addition, level-of-detail change attacks, as well as number–mode separation attacks. Furthermore, the watermark detection process is blind detection, which does not require the participation of the original BIM data, making the algorithm highly practical. In summary, the proposed algorithm solves the previous problem that 3D robust watermarking cannot be applied to BIM data, and provides a feasible solution for protecting the copyright of BIM data.

Author Contributions: Conceptualization, Q.Z., C.Z., and N.R.; methodology, Q.Z.; validation, Q.Z., C.Z., and N.R.; formal analysis, Q.Z. and N.R.; writing—original draft preparation, Q.Z.; writing—review and editing, Q.Z., C.Z., and N.R.; visualization, Q.Z.; supervision, C.Z. and N.R.; funding acquisition, C.Z. and N.R. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Science Foundation of China under grant 42071362, the National Key Research and Development Program of China under grant2022YFC3803600 and 2023YFB3907100.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: BIM data pertaining to this study can be obtained from Autodesk Revit 2017 (<https://help.autodesk.com/view/RVT/2017/CHS/?guid=GUID-61EF2F22-3A1F-4317-B925-1E85F138BE88>, accessed on 1 April 2023). The other data and the code that support the findings of this study are available on request from the corresponding author.

Acknowledgments: The authors would like to sincerely thank all the groups that provided data and support for this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kim, C.; Kim, J. Spatial spillovers of sport industry clusters and community resilience: Bridging a spatial lens to building a smart tourism city. *Inf. Process. Manag.* **2023**, *60*, 103266. [\[CrossRef\]](#)
2. Zhu, J.; Tan, Y.; Wang, X.; Wu, P. BIM/GIS integration for web GIS-based bridge management. *Ann. GIS* **2021**, *27*, 99–109. [\[CrossRef\]](#)
3. Chen, Y.; Huang, D.; Liu, Z.; Osmani, M.; Demian, P. Construction 4.0, Industry 4.0, and Building Information Modeling (BIM) for sustainable building development within the smart city. *Sustainability* **2022**, *14*, 10028. [\[CrossRef\]](#)
4. Lou, J.; Lu, W. Construction information authentication and integrity using blockchain-oriented watermarking techniques. *Autom. Constr.* **2022**, *143*, 104570. [\[CrossRef\]](#)
5. Das, M.; Tao, X.; Cheng, J.C.P. BIM security: A critical review and recommendations using encryption strategy and blockchain. *Autom. Constr.* **2021**, *126*, 103682. [\[CrossRef\]](#)
6. Ding, L.; Xu, X. Application of cloud storage on BIM life-cycle management. *Int. J. Adv. Robot Syst.* **2014**, *11*, 129. [\[CrossRef\]](#)
7. Gomez-Coronel, S.L.; Moya-Albor, E.; Brieva, J.; Romero-Arellano, A.A. Robust and Secure Watermarking Approach Based on Hermite Transform and SVD-DCT. *Appl. Sci.* **2023**, *13*, 8430. [\[CrossRef\]](#)
8. Megias, D.; Mazurczyk, W.; Kuribayashi, M. Data Hiding and Its Applications: Digital Watermarking and Steganography. *Appl. Sci.* **2021**, *11*, 10928. [\[CrossRef\]](#)
9. Qiao, T.; Ma, Y.; Zheng, N.; Wu, H.; Chen, Y.; Xu, M.; Luo, X. A Novel Model Watermarking for Protecting Generative Adversarial Network. *Comput. Secur.* **2023**, *127*, 103102. [\[CrossRef\]](#)
10. Liu, G.; Wang, H.; Miao, C. A three-dimensional text image watermarking model based on multilayer overlapping of extracted two-dimensional information. *Inf. Process. Manag.* **2023**, *60*, 103122. [\[CrossRef\]](#)
11. Wang, H.; Zhang, X.; Chen, S.; Su, Q. A blind watermarking algorithm based on specific component quantization in DHT domain. *Optik* **2023**, *281*, 170792. [\[CrossRef\]](#)
12. Faheem, Z.B.; Ali, M.; Raza, M.A.; Arslan, F.; Ali, J.; Masud, M.; Shorfuzzaman, M. Image Watermarking Scheme Using LSB and Image Gradient. *Appl. Sci.* **2022**, *12*, 4202. [\[CrossRef\]](#)
13. Wang, J.; Wu, D.; Li, L.; Zhao, J.; Wu, H.; Tang, Y. Robust periodic blind watermarking based on sub-block mapping and block encryption. *Expert Syst. Appl.* **2023**, *224*, 119981. [\[CrossRef\]](#)
14. Cao, H.; Hu, F.; Sun, Y.; Chen, S.; Su, Q. Robust and reversible color image watermarking based on DFT in the spatial domain. *Optik* **2022**, *262*, 169319. [\[CrossRef\]](#)
15. Sayahi, I.; Jallouli, M.; Mabrouk, A.B.; Mahjoub, M.A.; Amar, C.B. Robust hybrid watermarking approach for 3D multiresolution meshes based on spherical harmonics and wavelet transform. *Multimed. Tools Appl.* **2023**, *82*, 39841–39866. [\[CrossRef\]](#)
16. Abulkasim, H.; Jamjoom, M.; Abbas, S. Securing Copyright Using 3D Objects Blind Watermarking Scheme. *Comput. Mater. Contin.* **2022**, *72*, 5969–5983. [\[CrossRef\]](#)
17. Li, Z.; Bors, A.G. Steganalysis of meshes based on 3D wavelet multiresolution analysis. *Inf. Sci.* **2020**, *522*, 164–179. [\[CrossRef\]](#)
18. Hamidi, M.; Chetouani, A.; El Haziti, M.; El Hassouni, M.; Cherifi, H. Blind robust 3D mesh watermarking based on mesh saliency and wavelet transform for copyright protection. *Information* **2019**, *10*, 67. [\[CrossRef\]](#)
19. Al-Saadi, H.S.; Ghareeb, A.; Elhadad, A.A. Blind Watermarking Model of the 3D Object and the Polygonal Mesh Objects for Securing Copyright. *Comput. Intell. Neurosci.* **2021**, *2021*, 1392903. [\[CrossRef\]](#)
20. Delmotte, A.; Tanaka, K.; Kubo, H.; Funatomi, T.; Mukaigawa, Y. Blind 3D-printing watermarking using moment alignment and surface norm distribution. *IEEE Trans. Multimed.* **2020**, *23*, 3467–3482. [\[CrossRef\]](#)
21. Luo, T.; Li, L.; Zhang, S.; Wang, S.; Gu, W. A Novel Reversible Data Hiding Method for 3D Model in Homomorphic Encryption Domain. *Symmetry* **2021**, *13*, 1090. [\[CrossRef\]](#)
22. Li, L.; Wang, S.; Luo, T.; Chang, C.C.; Zhou, Q.; Li, H. Reversible data hiding for encrypted 3D model based on prediction error expansion. *J. Sens.* **2020**, *2020*, 8851999. [\[CrossRef\]](#)
23. Jang, H.U.; Choi, H.Y.; Son, J.; Kim, D.; Hou, J.U.; Choi, S.; Lee, H.K. Cropping-resilient 3D mesh watermarking based on consistent segmentation and mesh steganalysis. *Multimed. Tools Appl.* **2018**, *77*, 5685–5712. [\[CrossRef\]](#)
24. Wang, F.; Zhou, H.; Fang, H.; Zhang, W.; Yu, N. Deep 3D mesh watermarking with self-adaptive robustness. *Cybersecurity* **2022**, *5*, 24. [\[CrossRef\]](#)
25. Ren, S.; Cheng, H.; Fan, A. Dual information hiding algorithm based on the regularity of 3D mesh model. *Optoelectron. Lett.* **2022**, *18*, 559–565. [\[CrossRef\]](#)
26. Choi, H.Y.; Jang, H.U.; Son, J.; Lee, H.K. Blind 3D mesh watermarking based on cropping-resilient synchronization. *Multimed. Tools Appl.* **2017**, *76*, 26695–26721. [\[CrossRef\]](#)
27. Narendra, M.; Valarmathi, M.L.; Anbarasi, L.J.; Sarobin, M.V.R.; Al-Turjman, F. High embedding capacity in 3D model using intelligent Fuzzy based clustering. *Neural Comput. Appl.* **2022**, *34*, 17783–17792. [\[CrossRef\]](#)

28. Khalil, O.H.; Elhadad, A.; Ghareeb, A. A blind proposed 3D mesh watermarking technique for copyright protection. *Imaging Sci. J.* **2020**, *68*, 90–99. [[CrossRef](#)]
29. Qiu, Y.; Duan, H.; Sun, J.; Gu, H. Rich-information reversible watermarking scheme of vector maps. *Multimed. Tools Appl.* **2019**, *78*, 24955–24977. [[CrossRef](#)]
30. Singh, B.; Sharma, M.K. Efficient watermarking technique for protection and authentication of document images. *Multimed. Tools Appl.* **2022**, *81*, 22985–23005. [[CrossRef](#)]
31. Sun, Y.; Li, H.; Li, N. A novel cancelable fingerprint scheme based on random security sampling mechanism and relocation bloom filter. *Comput. Secur.* **2023**, *125*, 103021. [[CrossRef](#)]
32. van Kreveld, M.; Miltzow, T.; Ophelders, T.; Sonke, W.; Vermeulen, J.L. Between shapes, using the Hausdorff distance. *Comput. Geom.* **2022**, *100*, 101817. [[CrossRef](#)]
33. Kim, W.-H.; Hou, J.-U.; Jang, H.-U.; Lee, H.-K. Robust Template-Based Watermarking for DIBR 3D Images. *Appl. Sci.* **2018**, *8*, 911. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.